

UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE CIÊNCIAS APLICADAS A EDUCAÇÃO
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

Reusabilidade em SOA:
Um Mapeamento Sistemático da Literatura

Filipe Guimarães Ramos
Orientador: Prof. Msc. José Jorge Lima Dias Junior

RIO TINTO - PB
2014

Filipe Guimarães Ramos

Reusabilidade em SOA: Um Mapeamento Sistemático da Literatura

Trabalho de Conclusão de Curso apresentado para obtenção do título de Bacharel à banca examinadora no Curso de Bacharelado em Sistemas de Informação do Centro de Ciências Aplicadas e Educação (CCAEE), Campus IV da Universidade Federal da Paraíba.

Orientador: Prof. Msc. José Jorge L. Dias Junior.

RIO TINTO - PB
2014

R175r Ramos, Filipe Guimarães.
Reusabilidade em SOA: um mapeamento sistemático da literatura. / Filipe
Guimarães Ramos. – Rio Tinto: [s.n.], 2014.
75 f. : il. –

Orientador: Prof. Msc. José Jorge Lima Dias Junior.
Monografia (Graduação) – UFPB/CCAIE.

1. Software. 2. Arquitetura orientada a serviços - computação. 3.
Sistemas de informação.

UFPB/BS-CCAIE

CDU: 004.05(043.2)

Filipe Guimarães Ramos

**Reusabilidade em SOA:
Um Mapeamento Sistemático da Literatura**

Trabalho de Conclusão de Curso submetido ao Curso de Bacharelado em Sistemas de Informação da Universidade Federal da Paraíba, Campus IV, como parte dos requisitos necessários para obtenção do grau de BACHAREL EM SISTEMAS DE INFORMAÇÃO.

Assinatura do autor: _____

APROVADO POR:

Orientador: Prof. Msc. José Jorge Lima Dias Junior
Universidade Federal da Paraíba – Campus IV

Profa. Msc. Renata Viegas de Figueiredo
Universidade Federal da Paraíba – Campus IV

Profa. Dra. Yuska Paola Costa Aguiar
Universidade Federal da Paraíba – Campus IV

RIO TINTO - PB
2014

Aos amigos, colegas e professores, minha eterna gratidão por compartilhar comigo seus conhecimentos.

AGRADECIMENTOS

Agradeço primeiramente a Deus por me dar a dádiva de viver.

Ao meu orientador José Jorge Lima D. Junior, por acreditar no meu potencial, pela paciência, pelos ensinamentos e total assistência. Uma cara, que para mim, foi e continuará sendo um referencial.

A minha família pelo apoio moral e financeiro para que eu conseguisse chegar até aqui. Em especial a minha mãe por ser uma guerreira e sempre ser doar por inteira para que seus filhos tenham sucesso na vida. E ao meu falecido irmão Daniel Guimarães Ramos pela herança de ensinamentos sobre a vida.

A colega Joyce Aline Oliveira pela disponibilidade e colaboração indispensável na execução deste trabalho.

Aos meus amigos do grupo Cachaças por estarem sempre presentes tanto nos momentos de alegria como de tristeza. Amigos que espero ter comigo pelo resto da minha vida.

A todos os professores que fizeram parte da minha formação, por compartilharem seus conhecimentos.

E a todos os amigos que fiz na universidade, por viver comigo momentos inesquecíveis.

RESUMO

Considerando os desafios e os benefícios relacionados ao reuso em SOA (Arquitetura Orientada a Serviço), este trabalho apresenta um Mapeamento Sistemático da Literatura sobre este atributo com o propósito de identificar fatores que influenciam a sua reusabilidade, estratégias adotadas para potencializar o reuso, fases do ciclo de vida em que as estratégias normalmente são utilizadas e métricas associadas ao reuso. Os 24 artigos mapeados neste trabalho evidenciaram que o maior motivador do reuso é a redução de custo, a estratégia mais adotada é Linha de Produtos de *Software* e as fases de análise e de projeto de serviços são amplamente abordadas. Além das evidências citadas anteriormente, este trabalho também identifica lacunas de pesquisa na área de reusabilidade em SOA. A pesquisa é relevante para pesquisadores e organizações que busquem um estudo estruturado sobre o tema.

Palavras chave: Arquitetura Orientada a Serviços, SOA, Reuso, Reusabilidade, Mapeamento Sistemático da Literatura

ABSTRACT

Regarding the challenges and benefits related to the SOA (Service-Oriented Architecture) reuse, this paper presents a Literature Systematic Mapping about this attribute. It aims at identifying factors that influences the reusability, strategies adopted and metrics related to reuse in a SOA context. The 24 mapped papers have evidenced that cost is the most driver to reuse adoption. The strategy most adopted is Software Product Line and the service analysis and design activities are used widely. Beyond the evidence cited above, this study also identifies gaps in the area of reusability in SOA. This paper is relevant to the researchers and practitioners that need a structured work about this area.

Keywords: Service-Oriented Architecture, Reuse, Reusability, Systematic Literature Mapping

LISTA DE FIGURAS

Figura 1. Principais elementos de uma SOA (Dias, 2010).....	11
Figura 2. Abordagem <i>Top-down</i> e <i>Bottom-up</i> (Dias, 2010).....	14
Figura 3. Processo do mapeamento sistemático da literatura.....	16

LISTA DE TABELAS

Tabela 1. Benefícios do reuso de <i>software</i> (Almeida et al. 2007).....	7
Tabela 2. Características de Serviços (Furtado et al., 2009).....	10
Tabela 3. Padrões fundamentais de <i>Web Services</i> (Jossutis, 2008; Sommerville, 2011)	12
Tabela 4. Padrões de suporte de <i>Web Services</i> (Sommerville, 2011).....	13
Tabela 5. Questões de Pesquisa.....	17
Tabela 6. Bibliotecas digitais	17
Tabela 7. Strings de busca utilizadas	18
Tabela 8. Critérios de exclusão.....	18
Tabela 9. Resultado gerais da busca nas bibliotecas digitais	19
Tabela 10. Resultado após aplicação dos critérios de inclusão.	19
Tabela 11. Resultado final da seleção de artigos.....	19
Tabela 12. Artigos que citam os respectivos fatores como influenciadores do reuso em SOA.	24
Tabela 13. Conceitos utilizados nas estratégias apresentadas.	26
Tabela 14. Fase do ciclo de vida em que a estratégia é utilizada.	30

LISTA DE GRÁFICOS

Gráfico 1. Quantidade de artigos publicados por ano.....	21
Gráfico 2. Porcentagem de artigos publicados por ano.	22
Gráfico 3. Métodos de pesquisa realizados.	22
Gráfico 4. Representatividade por país.	23

LISTA DE SIGLAS

BPEL	<i>Business Process Execution Language</i>
BPM	<i>Bussiness Process Management</i>
BPMN	<i>Business Process Modeling Notation</i>
BPMF	Família de Modelo de Processos de Negócio, tradução de <i>Family Model Business Processes</i>
CBD	Desenvolvimento baseado em componentes, tradução de <i>Component-Based Development</i>
DSL	Linguagens Específicas do Domínio, tradução de <i>Domain Specific Languages</i>
EJB	<i>Enterprise JavaBeans</i>
ERP	<i>Enterprise Resource Planning</i>
FMS	Serviços de Modelos Recursos, tradução de <i>Features Models Services</i>
FODA	<i>Feature-Oriented Domain Analysis</i>
HTTP	<i>Hypertext Transfer Protocol</i>
JCA	<i>Java Connector Architecture</i>
LPS	Linhas de Produto de <i>Software</i>
MDE	<i>Model-Driven Engineering</i>
MSL	Mapeamento Sistemático da Literatura
OOAD	<i>Object-Oriented Analysis and Design</i>
ROI	Retorno Sobre Investimento, tradução de <i>Return on investment</i>
SCA	Arquitetura de Componente de Serviços, tradução de <i>Service Component Architecture</i>
SDCA	<i>Service Deployment and Configuration Architecture</i>
SOA	Arquitetura Orientada a Serviço, tradução de <i>Service-Oriented Architecture</i>
SOAML	<i>Service-Oriented Architecture Modeling Language</i>

SOAP	<i>Simple Object Access Protocol</i>
SOPL	Linha de Produtos Orientada a Serviços, tradução de <i>Services Oriented Products Line</i>
SOPLA	Arquitetura de Linha de Produtos Orientada a Serviços, tradução de <i>Services Oriented Products Line Architecture</i>
SOSE	Service Oriented <i>Software</i> Engineering
SPL	<i>Software Product Lines</i>
UDDI	<i>Universal Description, Discovery, and Integration</i>
UML	<i>Unified Modeling Language</i>
VOSD	Design de Serviço Orientado à Variação, tradução de <i>Variation Oriented Service Design</i>
WSDL	<i>Web Services Description Language</i>
XML	<i>eXtensible Markup Language</i>

SUMÁRIO

RESUMO	VIII
ABSTRACT	IX
LISTA DE FIGURAS	X
LISTA DE TABELAS.....	XI
LISTA DE GRÁFICOS	XII
LISTA DE SIGLAS.....	XIII
1 INTRODUÇÃO	1
1.1 OBJETIVOS.....	2
1.2 ESTRUTURA	3
2 FUNDAMENTAÇÃO TEÓRICA.....	4
2.1 REUSO DE <i>SOFTWARE</i>	4
2.2 ARQUITETURA ORIENTADA A SERVIÇOS	9
3 MÉTODO DE PESQUISA	16
3.1 QUESTÕES DE PESQUISA	16
3.2 PROCESSO DE BUSCA.....	17
3.3 CRITÉRIOS DE INCLUSÃO E EXCLUSÃO	18
3.4 CONDUÇÃO DO MAPEAMENTO SISTEMÁTICO DA LITERATURA.....	18
4 RESULTADOS.....	20
4.1 Q1: FATORES QUE INFLUENCIAM A REUSABILIDADE EM SOA	23
4.2 Q2: ESTRATÉGIAS ADOTADAS PARA POTENCIALIZAR A REUSABILIDADE EM SOA.....	25
4.3 Q3: EM QUE FASE DO CICLO DE DESENVOLVIMENTO AS ESTRATÉGIAS PARA POTENCIALIZAR A REUSABILIDADE EM SOA SÃO APLICADAS.....	28
4.4 Q4: MÉTRICAS UTILIZADAS PARA MEDIR REUSABILIDADE EM SOA.	30
5 CONSIDERAÇÕES FINAIS.....	31
5.1 LIMITAÇÕES E AMEAÇAS A VALIDADE	31
5.2 TRABALHOS RELACIONADOS.....	31
5.3 CONCLUSÃO	32
5.4 TRABALHOS FUTUROS	32
5.5 PRODUÇÃO BIBLIOGRÁFICA	32
REFERÊNCIAS BIBLIOGRÁFICAS	34
APÊNDICE A – ARTIGOS INCLUÍDOS	37
APÊNDICE B – ARTIGOS EXCLUÍDOS	43
APÊNDICE C – ARTIGOS INACESSÍVEIS.....	49
APÊNDICE D – ARTIGOS EXCLUÍDOS QUE ABORDAM MÉTRICAS DE REUSO EM SOA	50
APÊNDICE E – PROTOCOLO DO MAPEAMENTO SISTEMÁTICO.....	51

1 INTRODUÇÃO

A Arquitetura Orientada a Serviços (SOA) tem sido amplamente utilizada pelas organizações como uma estratégia no alcance da flexibilidade, da integração e da diminuição de custos inerentes ao desenvolvimento de *softwares*.

A adoção de SOA possibilita que funcionalidades sejam utilizadas como serviços na construção de novas aplicações sem a necessidade de reimplementá-las (Josutis, 2007) (Choi e Kim, 2008). A estratégia da reusabilidade em SOA consiste em utilizar os serviços não apenas em um único sistema, mas em um grupo deles para aumentar a agilidade de desenvolvimento (Alferez e Pelechano, 2011). Portanto, a reutilização é um dos grandes benefícios prometidos ao se adotar SOA, por ter essa capacidade de utilizar um mesmo serviço em mais de um sistema. De acordo com Geetha e Karthikeyan (2013), os serviços reutilizáveis aumentam o retorno do investimento (ROI) e também reduzem custos relacionados a projeto, desenvolvimento, teste e manutenção de *softwares*. Segundo Choi e Kim (2008) a reusabilidade também é tida como um dos principais critérios para a avaliação da qualidade de serviços. Outro benefício citado por Karthikeyan e Geetha (2013) é a capacidade de criar novos processos de negócio a partir de serviços existentes para atender as necessidades dos clientes e as exigências do mercado.

Modelos de maturidade propostos para este contexto consistem em abordagens para o planejamento sistemático de adoção, implantação e continuidade de uma Arquitetura Orientada a Serviços, apresentam reusabilidade como um atributo que deve ser internalizado na organização para alcançar a maturidade de SOA. Os modelos de maturidade possibilitam à empresa analisar o nível de alinhamento com os conceitos relacionados a SOA, tornando possível avaliar o grau de amadurecimento de um projeto SOA e consequentemente quanto é necessário evoluir para a obtenção na íntegra desta abordagem. (Dias et al., 2012).

Devido às vantagens mencionadas, a busca pelo reuso tem sido um dos motivos para a ampla adoção de SOA, visto ser este um dos princípios de projetos orientados a serviços (Shanmugasundaram et al., 2012) (Erl, 2005). No entanto, apesar dos benefícios apresentados, a dificuldade em definir quais serviços reutilizar e qual processo adotar para construir serviços reutilizáveis tem sido reportada em pesquisas empíricas no âmbito de SOA (Brahe, 2007).

De acordo com Shanmugasundaram et al. (2012), muitas pesquisas focam atributos de qualidade de SOA, mas não exploram especificamente o atributo de reusabilidade. A investigação da validade desta premissa, os benefícios propostos pela reusabilidade e as

dificuldades relatadas empiricamente inerentes ao processo de reuso motivaram a realização de um Mapeamento Sistemático da Literatura (MSL), com o propósito de identificar o estado da arte neste contexto, os fatores que influenciam positivamente e negativamente a reusabilidade, as estratégias comumente adotadas para a melhoria contínua do reuso, as métricas associadas a mensuração desse atributo, entre outros aspectos.

Mapeamento Sistemático da Literatura consiste em um método baseado em evidências e é classificado como estudo secundário, pois depende dos estudos primários para revelar evidências e construir conhecimento (Petersen et al., 2007). De acordo com Kitchenham (2007), o MSL é uma revisão ampla dos estudos primários de uma determinada área que tem como objetivo identificar as evidências disponíveis na área investigada. Budgen (2008), afirma que MSL se destina a identificar lacunas no conjunto de estudos primários, onde são necessárias novos ou melhores estudos primários, como também para identificar necessidade de realização de um mapeamento com um escopo maior.

A pesquisa é academicamente relevante por não ter sido encontrado um trabalho semelhante que agregue diferentes aspectos da reusabilidade em SOA através de um Mapeamento Sistemático. Em nível prático, a pesquisa pode ser útil para profissionais e interessados em SOA que visam compreender as características associadas à reusabilidade e não contam com uma revisão estruturada sobre o tema.

1.1 Objetivos

Este trabalho tem como objetivo geral realizar um mapeamento que permita agregar e categorizar as evidências produzidas na área de reusabilidade em SOA, que poderão ser utilizadas de formas complementares:

1. Na indústria, por profissionais que buscam conhecer formas de potencializar o reuso em SOA.
2. Na academia, por pesquisadores que tem interesse na área e buscam por uma visão do cenário de reuso em SOA.

O objetivo geral pode ser decomposto nos seguintes **objetivos específicos**:

- Realizar um estudo de mapeamento sistemático sobre reusabilidade em SOA;
- Identificar evidências sobre os fatores que influenciam o reuso em SOA;
- Identificar evidências sobre as estratégias adotadas para potencializar o reuso em SOA;
- Identificar evidências sobre as fases do ciclo de desenvolvimento em que as estratégias para potencializar o reuso em SOA são aplicadas;

- Identificar evidências sobre as métricas utilizadas para mensurar o reuso em SOA; e

1.2 Estrutura

O trabalho está organizado da seguinte forma: o Capítulo 2 apresenta os conceitos de Reuso de *Software* e Arquitetura Orientada a Serviços que são necessários pra compreensão do trabalho. No Capítulo 3 é apresentado o método de pesquisa, descrevendo todo o processo utilizado. No Capítulo 4 são apresentados os resultados alcançados através das respostas das questões de pesquisa. Finalmente, no Capítulo 5 discute-se alguns trabalhos futuros e considerações finais.

2 FUNDAMENTAÇÃO TEÓRICA

Para uma melhor compreensão deste trabalho, este capítulo apresentará os conceitos sobre a área de Reuso de *Software* assim como Arquitetura Orientada a Serviços como uma estratégia de se obter reuso.

2.1 Reuso de *Software*

Através da crescente demanda por sistemas de *software* complexos e completos e a necessidade de desenvolver novos produtos de forma mais rápida e com ciclos de produto cada vez menores surge a necessidade de meios para o aumento da produtividade. Programação orientada a objetos, desenvolvimento baseado em componentes, engenharia de domínio e as linhas de produtos de *software*, entre outros, são alguns dos meios que surgiram para contribuir com esta necessidade (Almeida et al., 2007), sendo conhecidas como forma de alcançar a reutilização de *software*. De acordo com Ezran et al. (2002), estudos sobre reutilização mostram que: *i)* 40% a 60% do código é reutilizável a partir de uma aplicação para outra; *ii)* 60% do projeto e código são reutilizáveis em aplicações de negócios; *iii)* 75% das funções do programa são comuns a mais de um programa, e que, *iv)* apenas 15% do código na maioria dos sistemas é exclusivo e novo para uma aplicação específica. De acordo com Mili et al. (1995), através da maximização da reutilização de ativos testados, certificados e organizados, as organizações podem obter melhorias no custo, tempo e qualidade.

Reutilização de *software* é definida por muitos pontos de vista diferentes. Peter Freeman (apud Ezran et al. 2002) define reutilização como o uso de qualquer informação que um desenvolvedor pode precisar no processo de criação de *software*. Tracz (apud Ezran et al. 2002) considera reutilizar como o uso de *software* que foi projetado para ser reutilizado. Basili e Rombach (1991) definem reuso de *software* como o uso de tudo associado a um projeto de *software*, incluindo o conhecimento. Para Frakes & Isoda (1994) reuso de *software* é definida como o uso do conhecimento de engenharia ou de artefatos de sistemas existentes para construir novos. De acordo com Ezran et al. (2002), a reutilização de *software* é a prática sistemática de desenvolvimento de *software* a partir de um estoque de blocos de construção, de modo que semelhanças em requisitos e / ou arquitetura entre as aplicações podem ser exploradas para obter benefícios substanciais de produtividade, qualidade e desempenho de negócios.

Este trabalho considera a visão geral de reuso de *software* de Krueger (1992), em que define Reutilização de *software* como um processo de criação de sistemas de *software* a partir

de *software* existente ao invés de construí-los do zero. A visão de Krueger (1992) foi considerada por sua amplitude.

2.1.1 As características básicas de Reuso de Software

Detalhes, apresentados por (Almeida et al., 2007), das três características principais listadas por (Ezran et al., 2002) na área de reutilização de *software*, são apresentados a seguir.

A reutilização é uma prática sistemática de desenvolvimento de *software*. Reutilização não-sistemática, ou seja, realizada de forma *ad hoc*, depende do conhecimento e da iniciativa individual, não é implantada de forma consistente em toda a organização, e está sujeito a pouco planejamento e controle de gestão. Se a organização é madura e bem gerida, não é impossível para reutilização não sistemática alcançar bons resultados. No entanto, mais provável é que a reutilização não sistemática seja caótica em seus efeitos, alimentando a cultura de alto risco de sucessos individuais e combate a problemas, e amplificando os problemas e defeitos, em vez de amortecê-los. Já na reutilização de *software* sistemática é necessário realizar uma compreensão de como a reutilização pode contribuir para os objetivos de todo o negócio, definir uma estratégia técnica e gerencial para atingir o valor máximo de reutilização, integrar a reutilização no processo total de *software*, e para o programa de melhoria de processo de *software*, garantir a todo o pessoal de *software* têm a competência e a motivação necessária, estabelecer um suporte apropriado organizacional, técnico orçamentário e usar medidas apropriadas para controlar o desempenho reutilização.

Reutilização explora semelhanças em requisitos e/ou arquitetura entre aplicações. Oportunidades de reutilização de um aplicativo para originar outro que tem necessidades semelhantes, ou arquiteturas semelhantes, ou ambos. A busca de semelhanças deve começar o mais próximo possível do início do projeto, isto é, quando os requisitos são identificados e decisões de arquitetura são feitas. As possibilidades de explorar semelhanças devem ser maximizadas por ter um processo de desenvolvimento que é concebido e gerido de modo a dar visibilidade total para o fluxo de requisitos e arquitetura e para todos os produtos de trabalho posteriores.

Reutilização oferece benefícios substanciais em produtividade, qualidade e desempenho empresarial. Reutilização sistemática de *software* é uma técnica que é utilizada para atender à necessidade de melhoria da qualidade de desenvolvimento de *software* e de eficiência (Krueger, 1992). Qualidade e produtividade podem ser melhoradas através da reutilização de todas as formas de experiência comprovada, incluindo produtos e processos,

bem como modelos de qualidade e produtividade. Produtividade poderia ser aumentada usando a experiência existente, em vez de criar tudo, desde o início (Basili et al., 1996). Melhorias de desempenho de negócios incluem redução de custos, de tempo mais curto para o mercado, e maior satisfação do cliente, que já foram observadas nas rubricas de produtividade e melhoria da qualidade. Esses benefícios podem iniciar um círculo virtuoso de maior rentabilidade, o crescimento, a competitividade, o aumento da quota de mercado e entrada em novos mercados.

2.1.2 Os benefícios do Reuso

Como mencionando anteriormente, a reutilização de *software* tem um impacto positivo na qualidade de *software*, assim como em custos e produtividade (Sametinger, 1997). Almeida et. al. (2007) dividem os benefícios em dois grupos: **melhoria da qualidade** que é formado pela qualidade, produtividade e confiabilidade e pelo grupo **redução de esforço** formado por trabalho redundante e tempo de desenvolvimento, tempo de mercado, documentação, custo de manutenção e tamanho da equipe. Os benefícios descritos por Almeida et. al. (2007) são apresentados na Tabela 1.

Grupo	Benefício	Descrição
Melhoria da qualidade	Qualidade	Correções de erros se acumulam a partir da reutilização do reuso. Isto origina uma maior qualidade de um componente reutilizado que seria o caso de um componente que é desenvolvido e utilizado apenas uma vez.
	Produtividade	Um ganho de produtividade é conseguido devido a diminuição da quantidade de código que é necessário ser desenvolvido. Isso resulta em menos esforços de teste e também economiza análise e projeto de trabalho, gerando economia global em custos.
	Confiabilidade	Usando componentes bem testados aumenta a confiabilidade de um sistema de <i>software</i> . Além disso, o uso de um componente de diversos sistemas aumenta a possibilidade de erros serem detectados e aumenta a confiança do componente.
Redução de esforço	Trabalho redundante e tempo de desenvolvimento	Desenvolver todos os sistemas a partir do zero significa desenvolvimento redundante de muitas partes, como especificações de requisitos, casos de uso, arquitetura, etc. Isto pode ser evitado quando estas peças estão disponíveis como componentes reutilizáveis e podem ser compartilhados, resultando em menos desenvolvimento,

		menos tempo e menor custos associado.
	Tempo de mercado	O sucesso ou fracasso de um produto de <i>software</i> é muitas vezes determinado por seu tempo de mercado. A utilização de componentes reutilizáveis pode resultar em uma redução de tempo de mercado, ou seja, o retorno do investimento começará mais cedo.
	Documentação	Embora a documentação seja muito importante para a manutenção de um sistema, esta é frequentemente negligenciada. Reutilizar componentes de <i>software</i> reduz a quantidade de documentação a ser escrita, mas agrava a importância do que está escrito. Assim, somente a estrutura geral do sistema e ativos recentemente desenvolvidos tem de ser documentado.
	Custo de manutenção	Menos defeitos e menos necessidade de manutenção do sistema podem ser esperados quando os componentes de qualidade comprovada já tenham sido utilizados.
	Tamanho da equipe	Algumas grandes equipes de desenvolvimento sofrem com a sobrecarga de comunicação. A duplicação do tamanho de uma equipe de desenvolvimento não resulta em duplicação da produtividade. Se muitos componentes podem ser reutilizados, em seguida, os sistemas de <i>software</i> podem ser desenvolvidos com equipes menores, levando a uma melhor comunicação, maior produtividade e menores custos.

Tabela 1. Benefícios do reuso de *software* (Almeida et al. 2007).

2.1.3 Obstáculos para se obter reuso

Existem alguns fatores que influenciam direta ou indiretamente na adoção de reutilização de *software*. Esses fatores podem ser divididos em obstáculos gerenciais, organizacionais, econômicos, conceituais ou técnicos (Sametinger, 1997). A seguir são apresentados estes obstáculos.

Obstáculos gerenciais e organizacionais: Reutilização não é apenas um problema técnico que deve ser resolvido por engenheiros de *software*, mas também é um problema que deve ser apoiado pela gestão e estruturas organizacionais adequadas. Dentre os obstáculos mais comuns se destacam a: (i) *Falta de apoio à gestão*, para realizar reuso de *software* é necessário custos iniciais, ele não pode ser amplamente alcançado em uma organização sem o apoio da gestão de nível superior. Os gestores têm de ser informado sobre os custos iniciais e tem que ser convencido sobre economia esperada; (ii) *O gerenciamento de projetos*, gerenciar projetos tradicionais não é uma tarefa fácil, principalmente, projetos relacionados à reutilização de *software*. Atingir reutilização de *software* em larga escala tem um impacto

sobre todo o ciclo de vida do *software*; (iii) *Estruturas organizacionais inadequadas*, as estruturas organizacionais devem considerar as diferentes necessidades que surgem quando, reutilização em larga escala explícita está sendo adotada. Por exemplo, uma equipe separada pode ser definida para desenvolver, manter e certificar componentes de *software*; (iv) *incentivos de gestão*, a falta de incentivos proíbe gestores de deixar seus desenvolvedores passar o tempo na fabricação de componentes de um sistema reutilizável. O sucesso de um desenvolvedor é muitas vezes medido apenas no tempo necessário para completar um projeto. Fazer qualquer trabalho além, embora seja benéfico para a empresa como um todo, diminui o seu sucesso. Mesmo quando os componentes são reutilizados, acessando repositórios de *software*, os benefícios obtidos são apenas uma fração do que poderia ser alcançado através da reutilização explícita, planejada e organizada.

Obstáculos econômicos: Reutilização pode economizar dinheiro no longo prazo, mas requer um custo para ser atingido. Custo associado à reutilização pode ser (Sametinger, 1997): custos de fazer algo reutilizável, custos de reutilizá-lo, e os custos de definição e implementação de um processo de reutilização. Reutilização requer investimentos iniciais em infra-estrutura, metodologia, treinamento, ferramentas e arquivos, com retornos sendo realizado somente anos mais tarde. Desenvolvendo ativos para reutilização é mais caro do que desenvolvê-los a uma única utilização. Níveis mais elevados de qualidade, confiabilidade, portabilidade, facilidade de manutenção, generalidade e documentação mais extensa é necessária. Estes custos não se justificam quando um componente é usado apenas uma vez.

Os obstáculos técnicos: Os obstáculos técnicos para a reutilização de *software* incluem questões relacionadas com a busca e recuperação de componentes, componentes legados e aspectos que envolvem a adaptação (Sametinger, 1997). Dentre estas questões é importante destacar a: (i) **Dificuldade de encontrar *software* reutilizáveis**, para reutilizar componentes de *software* deve existir formas eficientes de busca e recuperação deles. Além disso, é importante ter um repositório bem organizado contendo componentes com alguns meios de acesso a ele; (ii) ***Software* encontrado não reutilizável**, fácil acesso ao *software* existente não significa necessariamente aumentar a reutilização de *software*. Ativos reutilizáveis devem ser cuidadosamente especificados, projetados, implementados e documentados, assim, às vezes, modificar e adaptar o *software* pode ser mais caro do que programar a funcionalidade necessária a partir do zero; (iii) **Componentes legados não podem ser reutilizados**, uma abordagem conhecida pela reutilização de *software* é a utilização de *software* legado. No entanto, apenas recuperar os ativos existentes de sistema legado e tentar reutilizá-los para novos desenvolvimentos, não é suficiente para reutilização sistemática. Reengenharia pode ajudar na extração de componentes reutilizáveis do sistema legado, mas os esforços

necessários para a compreensão e extração deve ser considerado; (iv) **modificação**, é muito difícil encontrar um componente que funciona exatamente da mesma maneira que queremos. Desta forma, são necessárias modificações e deve existir maneiras para determinar os seus efeitos sobre o componente e os seus resultados de verificação anteriores.

2.2 Arquitetura Orientada a Serviços

Facilitar a integração dos sistemas, melhorar o fluxo de dados e atendimento ao cliente, reduzir os custos de TI e reutilização de funcionalidade já implementada são alguns dos benefícios que fazem as organizações terem interesse na adoção de SOA (Yoon e Carter, 2007).

A definição exata do termo SOA é difícil de ser encontrada, pois existem muitas diferentes (Josuttis, 2008). Nove definições formais diferentes de SOA foram identificadas por Erickson e Siau (2008), que vão desde “modularizando atividades comerciais de uma empresa para os serviços funcionais” até “usando *Web Services* para a computação distribuída.”. Neste trabalho é adotada a definição de Endrei et al. (2004), onde SOA é uma abordagem para a construção de sistemas distribuídos que oferece a funcionalidade de aplicativos como serviços tanto para aplicativos de usuário final como também para outros serviços.

2.2.1 Serviços

Serviços em SOA são recursos de *software* empacotados e bem definidos, auto-contidos, que proveem funcionalidades de negócio, independentes do estado ou contexto de outros serviços. Os serviços são identificados através de suas interfaces e os clientes do serviço não tem acesso à implementação do serviço, mas apenas à sua interface que informa como acessar o serviço. A Tabela 2 descreve algumas características de serviços (Furtado et al., 2009).

Característica de Serviços	
Granularidade adequada	Serviços devem representar funções, processos ou transações do negócio e encapsular outros componentes ou serviços de granularidade mais fina.
Contratos de serviços	Contrato de serviço especifica o que o serviço faz e como proceder para

bem definidos	utilizá-lo. Ele separa a funcionalidade das características tecnológicas.
Acoplamento fraco	Conceito tipicamente utilizado para tratar requisitos de escalabilidade, flexibilidade e tolerância a falha. O principal objetivo é minimizar dependências a fim de que modificações ou falhas em um serviço causem poucos impactos em outros serviços.
Capacidade de serem localizados	Serviços devem ser bem projetados e seus contratos devem ser publicados e estarem visíveis para os possíveis clientes poderem acessá-los. Serviços podem ser publicados através de: registros de serviços, repositórios de metadados, subdiretórios, ou uma localização qualquer conhecida.
Durabilidade	Os serviços devem existir ao longo do tempo, sendo mapeados a temas de processos duradouros. Os serviços podem ser alterados, mas o negócio ou tema do processo que o serviço implementa deve persistir.
Composição	Serviços devem ser projetados a fim de permitir que possam ser utilizados por outros serviços.
Alinhamento ao negócio	Serviços devem representar conceitos do negócio e estar de acordo com as necessidades do negócio como definido na estratégia e no plano do negócio, além de estarem associados aos processos de negócio identificados.
Reuso	Serviços devem ser implementados com reuso claro em processos de negócio. Deve-se estar ciente de que o reuso de um serviço pode levar à utilização do serviço por um novo cliente ou consumidor (não determinado previamente). Deve-se considerar a infra-estrutura a fim de garantir desempenho e tamanho apropriado de hardware e de largura da banda de rede.
Interoperabilidade	A interoperabilidade deve ser garantida pela aplicação de políticas, padrões e outros critérios de projeto durante o ciclo de vida do serviço: identificação de serviço, análise de serviços, projeto, implementação, teste de integração e implantação.

Tabela 2. Características de Serviços (Furtado et al., 2009)

2.2.2 Principais Elementos de SOA

A Figura 1 ilustra os principais elementos de SOA composta pelo fornecedor do serviço que é responsável por implementar e quem tem o domínio sobre o serviço; pelo registro de serviço que é o repositório onde os serviços são registrados para que possam ser localizados pelos consumidores; pelo consumidor do serviço que é quem encontra o serviço e o executa e

pelo contrato que consiste na especificação onde contém todas as informações necessárias para o consumidor utilizar o serviço (Dias, 2010b).

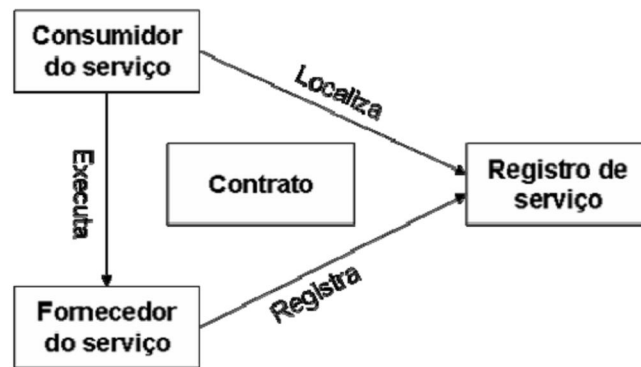


Figura 1. Principais elementos de uma SOA (Dias, 2010).

2.2.3 Reusabilidade em SOA

Antes de iniciar um projeto de uma SOA é preciso ter conhecimento de como os atributos de qualidade são afetados por este tipo de arquitetura, onde, os principais atributos de qualidade no cenário da SOA está a reusabilidade. (Dias Jr., 2010).

Dias et al. (2012) afirma que a reusabilidade evita a redundância de código, diminui esforços, custos e tempo associados a implementação e ainda aumenta a qualidade das aplicações. Dias (2010a) afirma também que, interoperabilidade, modularidade, registro central, e redução de dependências de tempo de compilação são alguns dos fatores que melhoram a probabilidade de reutilização como também formas eficientes de busca e recuperação de serviços e repositório bem organizado.

2.2.4 Web Services

Existem diferentes tecnologias associadas a SOA. Um das principais é a tecnologia *Web Services*. Sendo esta a maneira que a maioria dos analistas, fabricantes e autores recomendam como apropriada para a realização de SOA (Josuttis, 2008).

Web Services é definido por Josuttis (2008), como um conjunto de padrões que cobrem a interoperabilidade de sistemas e define tanto protocolos que são usados na comunicação quanto o formato das interfaces que são usadas para especificar os serviços e contratos de serviços.

Josuttis (2008), lista 5 padrões fundamentais de *Web Services* (Tabela 3), onde dois deles (XML e HTTP) são padrões gerais que já existiam e foram utilizados para realizar a abordagem de *Web Service* e os outros três (WSDL, SOAP e UDDI) são específicos para *Web Services*, sendo os primeiros padrões específicos para *Web Services*. Sommerville (2011) classifica os padrões UDDI e WSDL como padrões de definição de serviço e o padrão SOAP como padrão de serviço de mensagem o autor também apresenta como padrão fundamental o de processo WS-BPEL e também lista os padrões de suporte, WS-Security, WS-Addressing, WS-Transactions e WS-Reliable Messaging, que se destinam a apoiar SOA em diferentes aplicações empresariais. Os padrões fundamentais listados por Josuttis (2008) e Sommerville (2011) são descritos na tabela 3 e os padrões de suporte listado por Sommerville (2011) são descritos na tabela 4.

Padrões fundamentais de <i>Web Services</i>	
XML	É usado como formato geral para descrever modelos, formatos e tipos de dados. Todos os padrões de Web Service são baseados em XML 1.0, XSD (<i>XML Schema Definition</i>) e <i>namespaces</i> XML.
HTTP(S)	Protocolo de baixo nível usado pela internet. É o protocolo que pode ser usado para enviar <i>Web Services</i> pelas redes, usando tecnologia de internet.
WSDL	É usado para definir as interfaces dos serviços. Pode descrever dois aspectos diferentes de um serviço: sua assinatura (nome e parâmetros) e seus detalhes de ligação e <i>deploy</i> (protocolo e localização)
SOAP	É um padrão de troca de mensagens que oferece suporte à comunicação entre os serviços. Ele define os componentes essenciais e opcionais das mensagens passadas entre serviços.
UDDI	É um padrão para gerenciamento, de registro e localização, de <i>Web Services</i> .
WS-BPEL	É um padrão para uma linguagem de <i>workflow</i> , que é usada para definir programas de processo que envolve vários serviços diferentes.

Tabela 3. Padrões fundamentais de *Web Services* (Josuttis, 2008; Sommerville, 2011)

Padrões de suporte de <i>Web Services</i>	
WS-Security	Um conjunto de padrões que apóiam a proteção de <i>Web Services</i> ,

	incluindo padrões que especificam a definição de políticas de proteção e padrões que cobrem o uso de assinaturas digitais.
WS-Addressing	Define como as informações de endereço devem ser representadas em uma mensagem SOAP.
WS-Transactions	Define como as transações através de serviços distribuídos devem ser coordenadas.
WS-Reliable Messaging	Padrão para troca de mensagens que garante que elas serão entregues uma vez e apenas uma vez.

Tabela 4. Padrões de suporte de *Web Services* (Sommerville, 2011).

2.2.5 Business Process Management

SOA possui relação com algumas outras áreas, dentre elas, *Business Process Management* (BPM) – Gestão por processos de negócio. Esta relação acontece pelo fato de serviços fazerem parte de processos de negócio (Furtado et al., 2009). Serviços são as atividades dos níveis mais baixos de um processo decomposto e são eles os responsáveis por executarem as funções de negócio. A identificação destes serviços pode ser feita, em geral, através de duas abordagens, a *top-down* e a *bottom-up*. Na abordagem *top-down* um problema, sistema ou processo é decomposto em partes menores até atingir o nível de serviços, já na abordagem *bottom-up* um processo de negócio é construído por partes mais generalizadas através da composição de serviços (Josuttis, 2008). Entretanto, Bloomberg and Schmelzer (2006), declara que abordagens SOA devem ser tanto *top-down* quanto *bottom-up*, ou seja, tanto através da decomposição dos processos quanto da exposição de funcionalidades existentes como serviços e compondo-os dentro de processos. Isso se deve ao fato de que se for considerada apenas a abordagem *top-down*, provavelmente terá uma tendência de construir serviços tecnicamente difíceis ou complexos de implementar e se considerar apenas a abordagem *bottom-up* terá a tendência de construir serviços desnecessários ou que não atendam aos requisitos de negócio. A esta abordagem com a mistura das abordagens *top-down* e *bottom-up* dar-se o nome de *middle-out* ou *meet in the middle* ou *middle-ground* ou apenas *ágil* (Josuttis, 2008). A Figura 2 ilustra as abordagens *Top-down* e *Bottom-up*.

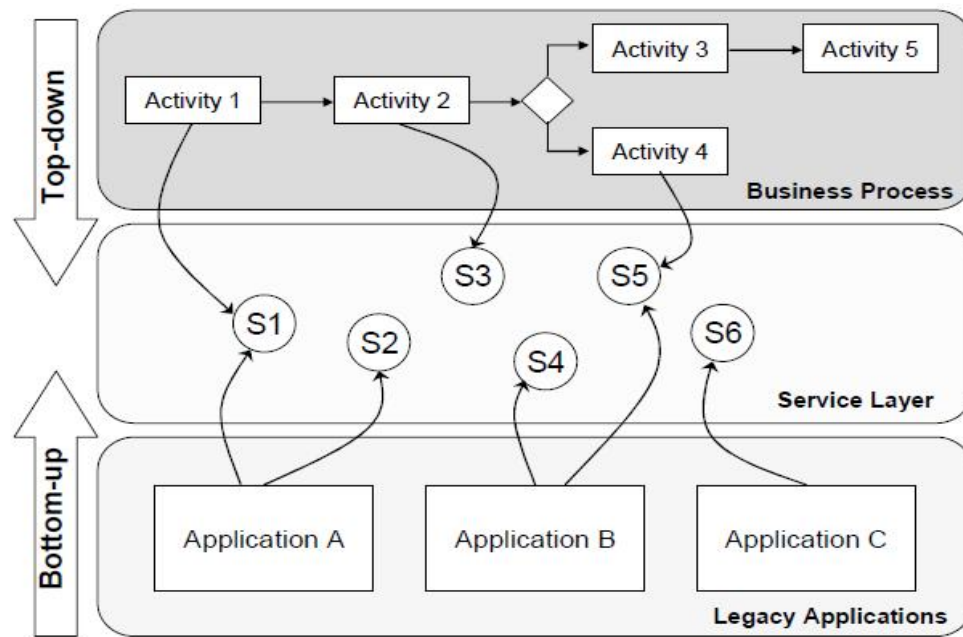


Figura 2. Abordagem *Top-down* e *Bottom-up* (Dias, 2010).

2.2.6 Metodologias SOA

Como SOA tem suas particularidades, vários autores tem propostos ciclos de vida específicos para sistemas baseados neste tipo de arquitetura. Svanidzaite (2012) afirma que a maioria das propostas de metodologia SOA dividem seu ciclo de desenvolvimento em seis fases: análise orientada a serviço, design orientado a serviço, desenvolvimento/construção de serviço, testes de serviço, implantação/transição de serviço e administração/gestão de serviço. Dentre as seis fases, Svanidzaite (2012) define a análise orientada a serviço e o design orientado a serviço como as fases mais importantes para o sucesso do desenvolvimento. Em (Svanidzaite, 2012) foi realizada uma análise das mais populares e conhecidas metodologias de desenvolvimento SOA que são: IBM RUP / SOMA, SOAF, a metodologia proposta por Thomas Erl, metodologia proposta por Papazoglou e SOPA. O resultado desta análise mostrou que as metodologias de análise SOA avaliadas variam das mais detalhadas e ordenadas para as menos detalhadas e desordenadas, possibilitando que o projetista adeque e adapte a metodologia ao escopo do seu projeto. Além disso, a maioria das metodologias SOA analisadas é construída e incorporada sobre técnicas comprovadas, tais como anotações OOAD, CBD, BPM, WSDL, BPEL, UML, o que significa que as abordagens utilizadas anteriormente ainda são aplicáveis e novas abordagens são oferecidas para o desenvolvimento de SOA, mas um novo método para organizar o processo de desenvolvimento SOA está faltando.

Dias et al. (2012), identifica os seguintes pontos chaves que as empresas devem considerar quando pensar em adotar SOA:

- Adoção gradativa;
- Consideração de diferentes dimensões na evolução da adoção, como arquitetura, pessoas, negócios, organização governança, infraestrutura, entre outras;
- Governança SOA como estratégia para garantir as políticas estabelecidas;
- BPM como estratégia de alinhamento entre negócio e TI;
- Envolvimento e educação de pessoas, como peças fundamentais para que as mudanças organizacionais aconteçam;
- Investimento de recursos e criação de um centro de excelência SOA para apoiar a iniciativa de adoção; e
- Utilização de ferramentas, por exemplo, um registro para catalogação dos serviços para que possam ser reutilizados.

Tanto SOA como reuso de *software*, são meios que surgiram para atender as novas necessidades de desenvolver *softwares* cada vez mais complexos e de forma cada vez mais rápida. A relação entre SOA e reuso de *software* se deve ao fato de que, através de SOA é possível aplicar o reuso de *software*, sendo, SOA é mecanismo para atingir o reuso de *software*. Esta relação se confirma na característica de SOA em desenvolver serviços para mais de um consumidor.

Após introduzir os conceitos necessários para entendimento deste trabalho será apresentado, em detalhes, no próxima capítulo o método de pesquisa utilizado.

3 Método de pesquisa

Uma metodologia baseada em evidências através do processo de mapeamento sistemático da literatura foi a adotada para realização deste trabalho. Mapeamento sistemático da literatura é um método que fornece uma visão geral de uma área de pesquisa e permite identificar, quantificar e analisar os tipos de pesquisas e os resultados disponíveis (Arksey e O'malley, 2005; Petersen et al., 2007).

Para a execução do mapeamento sistemático foi desenvolvido um protocolo, baseado nos parâmetros definidos em Kitchenham (2007), onde é apresentado todo o processo e os requisitos necessários para a execução do mapeamento. O processo do mapeamento sistemático da literatura realizado é ilustrado na Figura 3.

Nas próximas subseções serão apresentadas cada uma das atividades do planejamento e como foi a execução do processo adotado no mapeamento sistemático da literatura.

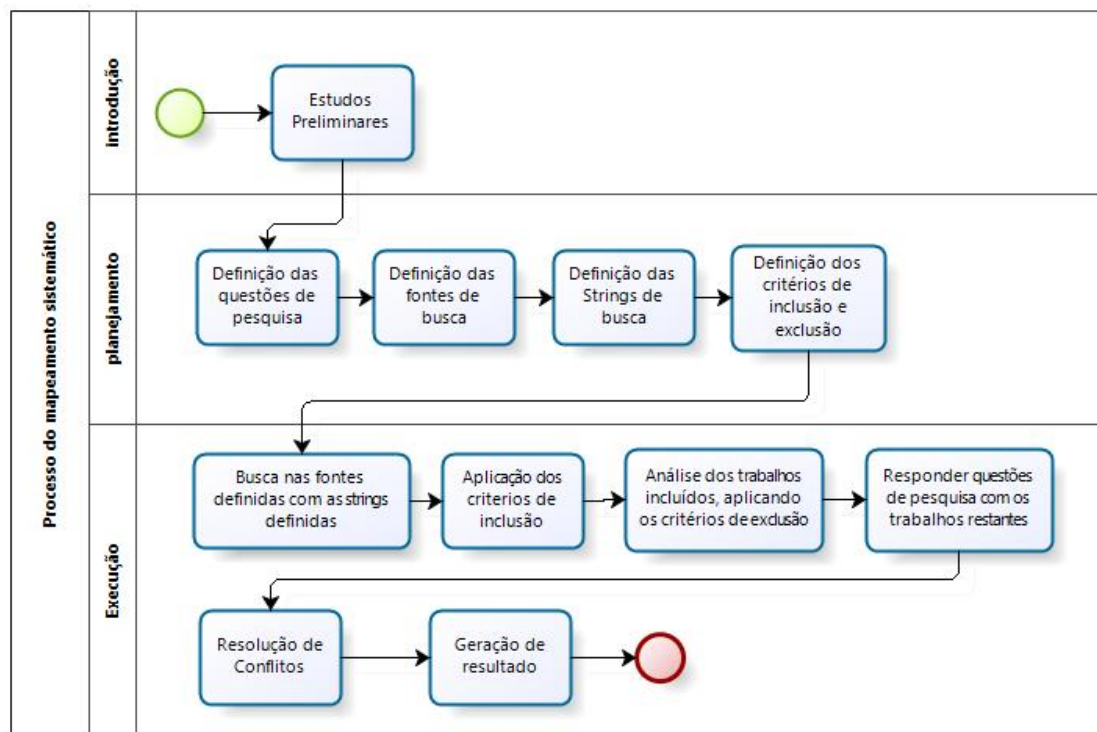


Figura 3. Processo do mapeamento sistemático da literatura

3.1 Questões de Pesquisa

Para guiar a execução do mapeamento, foram definidas algumas questões de pesquisa (Tabela 5). O objetivo, ao final da pesquisa, é responder estas questões.

Questões de Pesquisa
- Quais são os <i>fatores</i> que influenciam a reusabilidade em SOA?
- Quais são as <i>estratégias</i> adotadas para melhorar a reusabilidade em SOA?
- Em qual <i>fase do ciclo de vida</i> a estratégia é adotada?
- Quais são as <i>métricas</i> utilizadas para medir reusabilidade em SOA?

Tabela 5. Questões de Pesquisa

3.2 Processo de Busca

A estratégia usada para construir os termos de busca seguiu a mesma abordagem utilizada em (Kitchenham et al., 2006). Esta estratégia é sistematizada e define passos para derivar *strings* de busca a partir de questões de pesquisa, pontos de vista, *experts* e *papers* relevantes. Foram utilizadas os operadores booleanos OR e AND para incorporar sinônimos e integrar termos das *strings de busca* que consistiram em *SOA* ("SOA" / "service oriented architecture" / "SOSE" / "service oriented *software* engineering") e *Reuse* ("reuse" / "reusability" / "reusable"). Na Tabela 6 são apresentadas as *bibliotecas digitais* nas quais as buscas automáticas foram realizadas, as mesmas foram escolhidas por serem internacionalmente conhecidas no âmbito acadêmico pela qualidade dos estudos indexados. Além disso, estas bibliotecas permitem fácil acesso. Na tabela 7 são listadas as *strings* de busca correspondentes. Estas strings utilizadas são diferentes porque cada biblioteca digital possui sua própria sintaxe para aplicar operadores booleanos na pesquisa.

Bibliotecas Digitais	
IEEE Xplore	(http://ieeexplore.ieee.org/)
ACM Digital Library	(http://portal.acm.org)
Scopus	(http://www.scopus.com)

Tabela 6. Bibliotecas digitais

Base de busca	String
ACM Digital Library	(Abstract:(("SOA" or "service oriented" or "service-oriented") and ("reuse" or "reusability" or "reusable"))) or (Title:(("soa" or "service oriented" or "service-oriented") and ("reuse" or "reusability" or "reusable")))
IEEE Xplore	("SOA" OR "service-oriented" OR "service oriented") AND ("reuse" OR "reusability")
Scopus	TITLE-ABS-KEY(("service oriented" OR "service-oriented" OR "SOA") AND ("reuse" OR "reusability" OR "reusable"))

Tabela 7. Strings de busca utilizadas

3.3 Critérios de Inclusão e Exclusão

Com o objetivo de identificar estudos primários relevantes entre as pesquisas de reusabilidade em SOA, foi definido como **critério de inclusão** estudos primários publicados em conferências e periódicos que discutem como melhorar a reusabilidade em SOA ou em serviços. Já os **critérios de exclusão** definidos foram os listados na tabela 8:

Critérios de Exclusão
1. Trabalhos que apenas citam reuso como benefício de SOA, mas não possuem uma proposta para melhorar ou avaliar a reusabilidade de Arquitetura Orientada à Serviços.
2. Trabalhos que não tratam diretamente sobre como potencializar o reuso em SOA.
3. Trabalhos que tratam a reusabilidade apenas de um <i>software</i> específico.
4. Estudos duplicados (foi considerada a versão mais completa de artigos existentes em mais de um meio de publicação).
5. <i>Short papers</i> com menos de três páginas.
6. <i>White papers</i> , relatórios técnicos e livros.
7. Estudos secundários: revisões da literatura, mapeamentos, entre outros.

Tabela 8. Critérios de exclusão

3.4 Condução do Mapeamento Sistemático da Literatura

O mapeamento foi conduzido entre junho de 2013 e janeiro de 2014. Inicialmente foram executadas buscas automáticas nas bibliotecas digitais utilizando as *strings* definidas (Tabela 7). Destas buscas resultaram 96 registros na biblioteca da ACM Digital Library, 779 registros na biblioteca da IEEE Xplore e 1298 registro na biblioteca da Scopus (Tabela 9). Depois foi aplicado o critério de inclusão no título e no resumo dos artigos que gerou o resultado de 23 registros na biblioteca da ACM Digital Library, 34 registros na biblioteca da IEEE Xplore e 73 registros na biblioteca da Scopus (Tabela 10).

Base de busca	Quantidade de registros
ACM Digital Library	96
IEEE Xplore	779
Scopus	1298

Tabela 9. Resultado gerais da busca nas bibliotecas digitais

Base de busca	Quantidade de registros
ACM Digital Library	23
IEEE Xplore	34
Scopus	73

Tabela 10. Resultado após aplicação dos critérios de inclusão.

Dos artigos inclusos foi realizada a eliminação dos resultados duplicados e tiveram o título, a conferência/jornal e o ano de publicação registrados em uma planilha. Nesta etapa foram incluídos 117 artigos nos quais foram realizadas a leitura do resumo, da introdução, do núcleo da proposta e da conclusão do artigo. Após isso foram aplicados os critérios de exclusão e respondidas as questões de pesquisa definidas. Foram excluídos 87 artigos (Apêndice B) e incluídos 24 (Apêndice A). Foram solicitados aos autores via email 6 artigos inacessíveis (Apêndice C) gratuitamente nas bases de busca pesquisadas, no entanto, até o momento não foram obtidas respostas (Tabela 11). Todas as etapas do mapeamento foram realizadas separadamente por dois pesquisadores. Um terceiro pesquisador analisou e resolveu casos de conflitos.

Tipo da quantidade	Quantidade de registros
Total	117
Inacessíveis	6
Excluídos	87
Incluídos	24

Tabela 11. Resultado final da seleção de artigos.

Para responder as questões de pesquisa, utilizou-se o método *keywording*, que visou a captura de palavras-chaves para responderem as questões de pesquisa. Em seguida foi realizada a resolução de conflitos entre as respostas das questões de pesquisas obtendo com resultado a lista final dos artigos selecionados para a escrita do relatório de mapeamento sistemático e geração dos resultados (Apêndice A).

A fim de calibrar as análises, foi realizado um piloto no qual todos os pesquisadores analisaram independentemente cinco artigos selecionados aleatoriamente e discutiram os resultados por meio de reuniões virtuais nas quais um terceiro pesquisador resolveu os conflitos. A análise dos artigos por pesquisadores distintos, a resolução de conflitos por um

terceiro pesquisador e a execução de um piloto, visou a diminuição de possíveis vieses da pesquisa.

No próximo capítulo será apresentado os resultados gerados pelo análise dos 24 artigos incluídos (Apêndice A).

4 RESULTADOS

Este capítulo apresenta os resultados da análise realizada nos 24 artigos incluídos (apêndice A), respondendo as questões de pesquisa definidas na Seção 3.1. Embora o ano de publicação não tenha sido um critério do mapeamento, os artigos obtidos nas buscas às

bibliotecas digitais tiveram sua publicação realizada entre o ano de 2006 e 2013. Isto gera uma evidência de que o interesse por meios de potencializar o reuso em SOA teve início no ano de 2006. Nos Gráficos 1 e 2 é possível visualizar a quantidade de artigos publicados em cada ano. No período de 2006 a 2008 foram publicados 3 artigos sendo 1 no ano de 2006 e 2 no ano de 2008 o que representa 12,4% da quantidade total de artigos incluídos. No ano de 2009 foi realizada a publicação de 4 artigos, um aumento de mais de 100% de publicações em relação ao ano anterior e com uma representatividade de 16,67% da quantidade total de artigos incluídos. Em 2010 o número de publicações continuou aumentando em relação ao ano anterior, atingindo 6 artigos publicados que representa 25% do total de artigos incluídos. Em 2011 houve uma queda no número de publicações apresentando apenas 2 artigos publicados, o que representa 8,33% do total de artigos incluídos. Já em 2012 o número de artigos voltou a crescer, apresentado 6 publicações que representa 25% do total de artigos incluídos. No ano de 2013 foram encontrados 3 artigos publicados, o que representa 12,25% do total de artigos incluídos. Como o mapeamento foi realizado entre os anos de 2013 e 2014, é possível que algum artigo publicado neste período não tenha sido contabilizado.

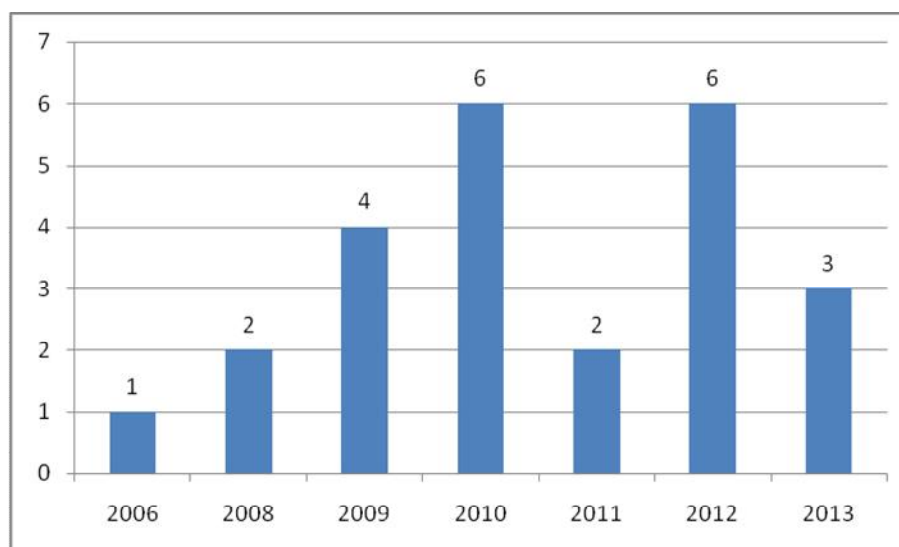


Gráfico 1. Quantidade de artigos publicados por ano.

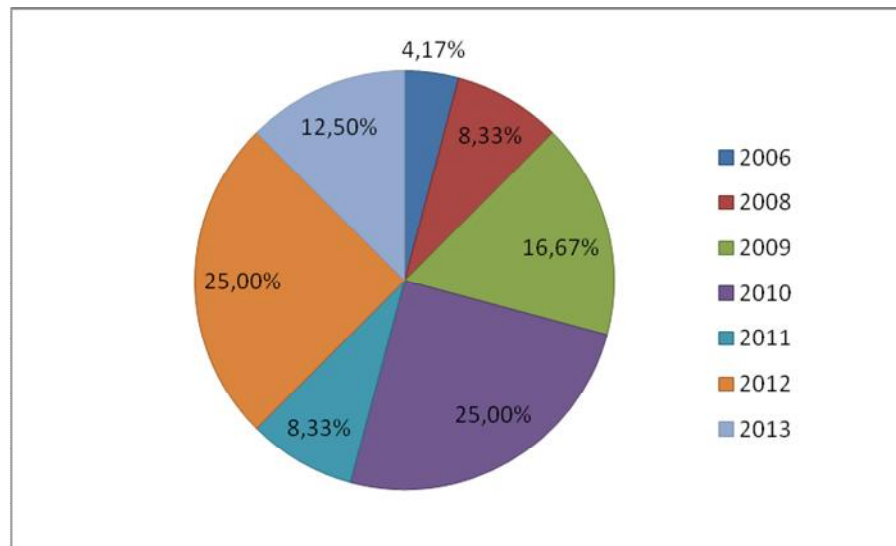


Gráfico 2. Porcentagem de artigos publicados por ano.

No Gráfico 3 são exibidos os métodos de pesquisa realizados no artigos incluídos. Dentre os 24 artigos incluídos, 15 utilizaram o método de *proposta de solução* de metodologias, modelos e abordagens com o objetivo de melhorar a reusabilidade em SOA. A grande quantidade de artigos evidencia **um interesse dos pesquisadores em sugerir meios para potencializar o reuso na Arquitetura Orientada a Serviços**. Outros 5 trabalhos, dentre os 24 incluídos, utilizaram *métodos mistos*, onde 2 utilizaram método de proposta de solução e validação de solução por meio de *estudo de caso*, 1 realizou a mixagem de *Pesquisa-Ação* com estudo de caso e 1 a mixagem de Pesquisa-Ação com *survey*. Outros 3 artigos utilizaram as metodologias de *survey*, *Pesquisa-Ação* e *estudo de caso* respectivamente. Apenas um artigo não explicitou o método utilizado.

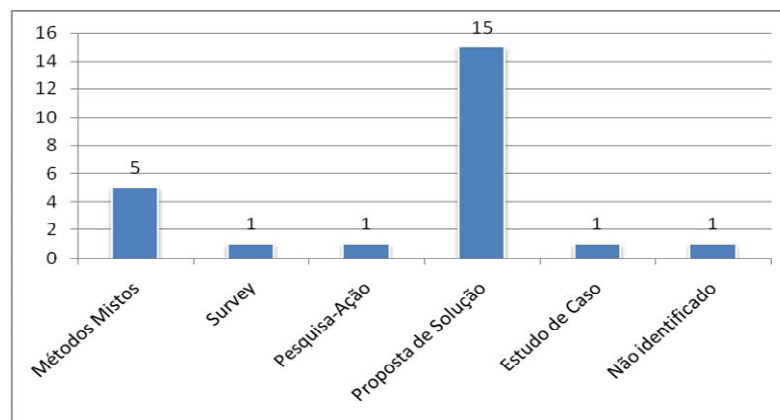


Gráfico 3. Métodos de pesquisa realizados.

O Gráfico 4 apresenta quais países estão pesquisando sobre o assunto. A Coreia é o país que tem o maior envolvimento nos artigos incluídos, tendo contribuído com 6 artigos de um total de 24. A China e a Índia estiveram envolvidas em 3 artigos, cada uma. A Austrália, Brasil,

Espanha, Reino Unido e Holanda estiveram, cada um, envolvidos em 2 artigos. E a Colômbia, México, Estados Unidos, Marrocos, Alemanha, Nigéria, África do Sul, Turquia e Irã apresentaram, cada um, envolvimento com apenas 1 artigo. A grande maioria dos artigos incluídos tiveram envolvimento da Coreia, Índia e China, isto evidencia que o continente Asiático tem grande interesse no assunto.

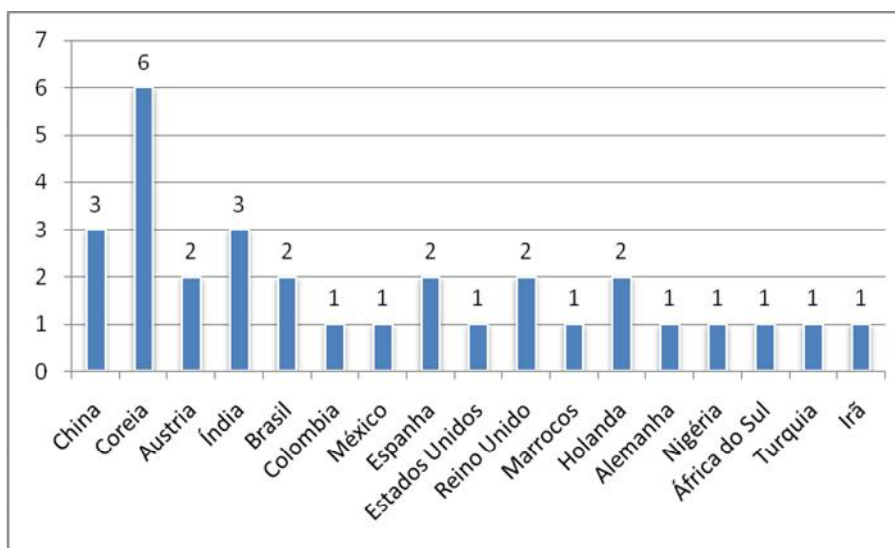


Gráfico 4. Representatividade por país.

Após serem apresentados dados gerais da análise serão apresentadas nas subseções abaixo as respostas das questões de pesquisa baseada na análise que foi realizada no mapeamento.

4.1 Q1: Fatores que influenciam a reusabilidade em SOA

Foram considerados como fatores qualquer elemento que influencia reusabilidade em SOA. Na análise foram identificados fatores em 66,67% (16;24) artigos. 8,3% (2;16) dos artigos [P2 e P64] mostraram que um dos fatores da reusabilidade em SOA é o *custo* envolvido no processo. A necessidade de reusar *sistemas legados*, de integrar *ambientes heterogêneos* e de maximizar o *Retorno do Investimento* (ROI) também influencia o reuso em SOA [P37]. A carência de um *processo sistemático* para identificar e conceber serviços reutilizáveis mostrou ser um fator que influencia o reuso [P86]. Outro fator que influencia a reusabilidade é a busca por processos que possibilitem o *desenho de linhas de produtos orientadas a serviços* [P58] como também a utilização de um *modelo de design de serviço* [P74] ou um *modelo de ciclo de vida de serviços para SOA* [P67]. O reuso também é influenciado pela necessidade de identificar, modelar e gerir a *variabilidade dos serviços* [P14

e P99] e também pela *identificação do serviço* [P48] e *reputação do serviço* [P15]. Outros fatores que influenciam o reuso são a busca pela *redução de tempo de desenvolvimento de software* [P2] e a *Governança SOA* [P5]. *Recursos de análise de domínio* [P18], *metadados utilizados em repositórios de serviços* [P19], *abordagem de desenvolvimento baseada em recursos* [PP44] e *uma combinação de BPEL4WS e Arquitetura de Componente de Serviços (SCA)* [P110] também são fatores que influencia a reusabilidade em SOA. 33.4% (8; 16) dos artigos não explicitaram claramente fatores que influenciam o reuso em SOA. É perceptível a variedade entre os fatores encontrados nos trabalhos mapeados (Tabela 12). Isto comprova que os fatores são subjetivos e variam de acordo com a percepção dos autores sobre o termo.

Fatores	Artigos
Custo	P2; P64
Tempo de Desenvolvimento	P2
Governança SOA	P5
Variabilidade de serviços	P14; P99
Reputação do Serviço	p15
Recursos de Analise de Dominio	P18
Metadados Utilizados no Repositorio	P19
Principios da Engenharia de SPL e MDD	P40
Desenvolvimento Baseada em Recursos	P44
Identificação de Serviços	P48
Processo para Desenhar SPL Orientado a Serviços	P58
Modelo de Ciclo de Vida de Serviços para SOA	P67
Projeto e Desen. de Serviços Baseado em SPL	P69
Modelo de Design de Serviço	P74
Processo Sistemático para Reuso de Serviços	P86
Combinação de BPEL4WS e SCA	P110

Tabela 12. Artigos que citam os respectivos fatores como influenciadores do reuso em SOA.

Podemos observar também que 3 fatores, apresentados em [P40, P58 e P69] respectivamente, estão relacionados com Linhas Produtos de *Software* (SPL), isto evidencia que SPL é um aliado em potencial para potencializar o reuso em SOA. Em SPL, os produtos de software são derivados da escolha de recursos que fazem parte de um produto já desenvolvido e a retirada de recursos que não são parte do produto de software que esta em

desenvolvimento, ou seja, SPL significar desenvolver um software através do reuso de funcionalidades já implementas em um software anterior (Alferez e Pelechano, 2011).

4.2 Q2: Estratégias adotadas para potencializar a reusabilidade em SOA

Nesta seção são apresentadas as *estratégias* adotadas pelos trabalhos mapeados para potencializar o reuso em SOA. Em 29,2% (7; 24) dos artigos [P2, P37, P40, P44, P48, P58, P69] foram apresentadas como estratégia a utilização de conceitos de *Linhas de Produtos de Software* para o alcance da reusabilidade. Sendo esta a estratégia mais representativa. Em 12,5% (3;24) dos estudos foi proposta a *gestão da variabilidade* de serviços [P14, P60, P99]. O uso de *repositórios* foi adotado em 8,4% (2;24) dos artigos, [P11,P19]. Outra estratégia adotada foi *abordagens de análise e de projeto* de serviços, abordada por 16,7% (4;24) dos artigos [P6, P67, P86, P74]. Os artigos restantes (33,4%, 8;24) optaram por estratégias distintas que consistiram em *Arquitetura 3D* [P1], *Governança SOA* [P5], *gestão da incompatibilidade de serviços* [P43], *gestão de sistemas legados* [P54], *Análise de Domínio Orientada a Características* (FODA) [P18], *Linguagem de Execução de Processos de Negócio* (BPEL) com *Arquitetura de Componentes de Serviços* (SCA) [P110], *Abordagem baseada em probabilidade de descrição e incompatibilidade lógica* [P64] e *Reputação de Serviços* [P15]. A Tabela 13 apresenta a quantidade de artigos que cada estratégia apresentou.

Conceito	Artigos
Linhas de Produtos de Software	P2; P37; P40; P44; P48; P58; P69
Gestão da Variabilidade	P14; P60; P99
Repositórios	P11; P19
Abordagens de Análise e Projeto	P6; P67; P86; P74
Arquiterura 3D	P1

Governança SOA	P5
Gestão da Incompatibilidade de Serviços	P43
Gestão de Sistemas Legados	P54
Análise de Domínio Orientada a Características	P18
Linguagem de Execução de Processos de Negócio (BPEL) com Arquitetura de Componentes de Serviços (SCA)	P110
Abordagem Baseada em Probabilidade de Descrição e Incompatibilidade Lógica	P64
Reputação de Serviços	P15

Tabela 13. Conceitos utilizados nas estratégias apresentadas.

No [P2] foi proposto o reuso de serviços por meio da **Hibridização de Linha de Produtos de Software com SOA (SOPL)**. Deste modo, os autores propuseram a análise e o projeto de serviços com base nos conceitos de Arquitetura de Linha de Produtos Orientada a Serviços (SOPLA), criando uma iteração entre as duas áreas para promover a reusabilidade em SOA. Em [P37] foram propostos os conceitos de Linhas de Produtos de Software (SPL) associados aos conceitos de variabilidade para apoiar a **descoberta e o registro de serviços com base em 2-LEVEL UDDI**. SPL também foi abordada no [P40] através de uma proposta de um **Método de Desenvolvimento Orientado à Modelos com a notação BPMN** para promover a composição e, conseqüentemente, o reuso de serviços.

SPL ainda foi utilizada em [P44] para tratar a descoberta, o desenvolvimento e a variabilidade de serviços por meio de **Família de Modelo de Processos de Negócio (BPMF)**. Este modelo visa auxiliar o design e a implementação de serviços reutilizáveis. No [P48] foi proposta uma junção entre SPL e Modelos de Processos de Negócio em SOA para criar um **método de identificação que une serviços de modelos recursos (FMS) de SPLs e Gestão de Processos de Negócios (BPM) em SOA** com vista no reuso de serviços. SPL também foi utilizada em [P58] na criação de **uma abordagem que visa especificar artefatos, definir atividades e fornecer orientações para a equipe de SOA da organização**. Esta abordagem promove a utilização destes artefatos em outros projetos, possibilitando o reuso em SOA. Outro registro de SPL foi identificado em [P69] onde foi proposto um **modelo de projeto e de desenvolvimento de serviços de domínio**.

Em [P19] foi desenvolvido e utilizado um **repositório integrado a ERPs e a sistemas legados** para promover o reuso em SOA. A ferramenta visou facilitar a gestão, a persistência, o compartilhamento e a busca de serviços e de sistemas disponíveis em um catálogo existente no repositório. No [P11] foi proposto um **modelo de referência de análise composto por ativos reutilizáveis existentes em repositórios múltiplos e integráveis nas diferentes**

camadas de abstração da aplicação. A integração destes repositórios visa promover a gestão e o reuso de artefatos gerados durante a fase de análise de serviços.

O [P6] propõe uma **metodologia para orientar o processo de análise e de projeto de serviços**. Para a fase de análise sugeriu-se: i) a identificação do serviço de negócio, ii) a identificação do contexto e iii) a decomposição do serviço de negócio. Para a fase de projeto propôs-se: i) a definição do serviço candidato, ii) a identificação do registro do serviço, iii) a definição do contrato do serviço, iv) a modelagem da rede de serviços e v) a modelagem da coreografia dos serviços. No trabalho citado é proposto que tanto a análise e o projeto de serviços sejam apoiados UML e por SOAML. Em [P86] foi proposto um **processo sistemático com base em completude, aplicabilidade, rastreabilidade e precisão para identificar e desenhar serviços reutilizáveis**. Este processo é composto por cinco atividades: i) Identificar processos de negócio; ii) Especificar processos de negócio; iii) Identificar serviços reusáveis; iv) Especificar serviços reusáveis e v) Projetar serviços reusáveis. Em [P74] propôs-se uma abordagem de projeto de serviços orientada a padrões que consiste em três fases: i) análise de negócios; ii) modelagem de serviços e iv) desenvolvimento de componentes de *software*.

No [P14] o conceito da variabilidade de serviços foi utilizado na construção de uma **metodologia para identificar e modelar possíveis variâncias em SOA**. Para permitir a representação da variabilidade foi proposto o uso da extensão varWSDL na geração de documentos WSDL, XML e na especificação de artefatos de serviços. A variabilidade também foi utilizada em [P60] que propôs uma metodologia **baseada em Engenharia Orientada a Variação para identificar e controlar variâncias de serviços**. Para tal foi utilizada também uma técnica denominada Design de Serviço Orientado à Variação (VOSD) para descobrir e derivar automaticamente variâncias de serviços a partir de um repositório reutilizável. No [P99] foi proposto um **conjunto de mecanismos, chamado de Engenharia Orientada a Variação (VOE)**, para aperfeiçoar de forma incremental os serviços de *software* existentes. VOE consiste em uma abordagem formal abrangente para a modelagem de variabilidade fim-a-fim.

No [P18] foi utilizado o conceito de Análise de Domínio Orientada à Características (FODA) na proposição de um **método para guiar a análise e o projeto de serviços reutilizáveis**. A proposta sugere a análise do contexto, dos requisitos do domínio e das aplicações para extrair características de serviços que poderão ser reusados em uma SOA. Em [P54] foi proposta a **análise da reutilização de sistemas legados em arquiteturas SOA com**

base em Engenharia Orientada a Modelo (MDE) e Linguagens Específicas do Domínio (DSL) para representar e analisar serviços candidatos em SOA integráveis a sistemas legados.

Em [P110] utilizou-se a combinação de Linguagem de Execução de Processos de Negócio (BPEL4WS) e Arquitetura de Componentes de Serviços (SCA) para a criação **de padrões de aplicações reutilizáveis com o uso de EJB e JCA**. No [P43] **padrões adaptadores com base em gestão da incompatibilidade** foram propostos para identificar e evitar casos de incompatibilidade na descoberta de serviços.

No [P64] foi proposta uma **abordagem que define uma métrica quantitativa para mensurar a reusabilidade de serviços compostos, com base na medição e análise lógica e probabilística do nível de reuso**.

Em [P1] os autores propuseram o uso de Arquitetura 3D (Decisão - *Decison*, Níveis - *Degree* e Dados - *Data*) para a Automação de Serviços. Com base nesta arquitetura foi desenvolvido um **sistema denominado SDCA para apoiar o projeto e a implementação de serviços reutilizáveis**.

Em [P5] a Governança em SOA inspirou a criação de um **modelo de reuso com modularidade e sem modularidade** baseada nesta abordagem.

A eficiência de cada estratégia pode ser influenciada por diversos fatores como ambiente organizacional, conhecimento da equipe em SOA, tecnologias, *softwares*, método de aplicação, objetivos, entre outros. Portanto, só é possível avaliar qual estratégia melhor potencializa o reuso mediante a real adoção da proposta. A variedade de estratégias encontradas nos artigos mapeados possibilita a escolha daquela que seja mais adequada aos propósitos de quem busca a reusabilidade em seu projeto SOA. Porém, a ampla utilização de Linhas de Produtos de *Software* (33%, 8;24) indica que os conceitos desta área podem prover uma ponte para alcançar o reuso em SOA. Nesse contexto os artigos [P37] e [P44] se destacaram por validar a proposta baseada em LPS através da condução de um estudo de caso e por reportar os resultados.

4.3 Q3: Em que fase do ciclo de desenvolvimento as estratégias para potencializar a reusabilidade em SOA são aplicadas

Esta questão teve como objetivo identificar em que fases as estratégias para potencializar a reusabilidade em SOA foram utilizadas. A Tabela 14 apresenta a quantidade de artigos que apresenta estratégias para cada fase, alguns artigos apresentam estratégias para mais de uma fase. Em 45,8% (11; 24) dos artigos a estratégia foi utilizada na fase de análise [P2, P6, P11, P18, P37, P40, P43, P44, P69, P86, P99]. Esta fase engloba atividades de elicitação e de especificação de requisitos de serviços. Em 50% (13; 24) dos trabalhos

mapeados a estratégia foi utilizada na fase de projeto, que envolve a modelagem dos serviços [P2, P6, P15, P19, P40, P43, P44, P54, P58, P60, P74, P86, P99]. A fase de desenvolvimento foi abordada em 25% (7; 24) dos artigos [P2, P14, P37, P54, P58, P99, P110]. No P37 a estratégia foi utilizada também na fase de implantação. Em 4,2% (1; 24) dos artigos a estratégia foi tratada na fase de automação [P1]. E em 4,2% (1; 24) dos artigos foi utilizado a estratégia na fase de monitoramento [P64]. Em 8,4% (2; 24) dos artigos não explicitaram a fase em que a estratégia acontece.

Foi observado que apenas 3 artigos do total de 24 focaram respectivamente a fase de implantação, de automação e de monitoramento. Estas constatações indicam que a área de SOA é academicamente carente de trabalhos que potencializem a reusabilidade nestas fases. Esta lacuna favorece a realização de pesquisas que proponham meios para possibilitar que componentes de Arquitetura Orientada a Serviços sejam reutilizados nas fases citadas anteriormente. Podemos observar também que nas fases de Análise e de Projeto se concentra a maior parte das estratégias para potencializar o reuso em SOA, evidenciando a importância destas fases para o reuso em SOA.

Na Tabela 14 é possível ver que alguns artigos apresentam estratégias que são aplicadas em mais de uma fase do ciclo de desenvolvimento. Em (2; 24) artigos [P2, P99] foram citados estratégias que são utilizadas na fase de Análise, na fase de Projeto e na fase de Desenvolvimento. Em outros (2; 24) artigos [P54, P58] foram apresentadas estratégias que são utilizadas tanto na fase de Projeto como na de Desenvolvimento. Em (5; 24) artigos [P6, P40, P43, P44, P86] foram apresentas estratégias que são aplicadas na fases de Análise e na fase de Projeto. Em (1; 24) artigo [P37] foi apresentada uma estratégia que é utilizada tanto na fase de Análise como na fase de Desenvolvimento e na fase de Implantação. Estratégias que foram utilizadas apenas na fase de Análise foram citadas em (3; 24) artigos [P11, P18, P69]. Em outros (2; 24) artigos [P14, P110] foram apresentadas estratégias que são utilizadas apenas na fase de desenvolvimento. E em (4; 24) artigos [P15, P19, P60, P74] foram apresentadas estratégias que foram utilizadas apenas na fase de Projeto. Isto confirma que as fases de Análise e Projeto são as fases mais importantes para impulsionar a reusabilidade em SOA.

Artigos/Fases	Análise	Projeto	Desenvolvimento	Implantação	Automação	Monitoramento	TOTAL
P1					X		1
P2	X	X	X				3
P5		X					1
P6	X						1

P11	X						1
P14			X				1
P15		X					1
P18	X						1
P19		X					1
P37	X		X	X			3
P40	X	X					2
P43	X	X					2
P44	X	X					2
P48							0
P54		X	X				2
P58		X	X				2
P60		X					1
P64						X	1
P67							0
P69	X						1
P74		X					1
P86	X	X					2
P99	X	X	X				3
P110			X				1
TOTAL	11	13	7	1	1	1	

Tabela 14. Fase do ciclo de vida em que a estratégia é utilizada.

4.4 Q4: Métricas utilizadas para medir reusabilidade em SOA.

Nesta questão buscou-se identificar possíveis métricas utilizadas para medir a reusabilidade em SOA. Apenas (1; 24) artigo [P64] dos artigos incluídos no mapeamento explicitou indicadores de medição. A baixa quantidade de artigos que explicitaram métricas para medir a reusabilidade em SOA se deve ao fato de que apenas (1; 24) artigos incluídos propuseram melhorias para a fase de monitoramento, na qual normalmente os serviços são mensurados. Além do artigo citado acima, houve uma exceção em (2; 24) dos artigos excluídos [P33, P39] (Apêndice D), que juntamente com o artigo incluído [P64] propuseram métricas para mensurar o **nível de dependência de serviços**, o **nível de acoplamento dos serviços** e o **nível de incompatibilidade de reuso de um serviço**. No entanto, estes artigos não tratavam diretamente como potencializar o reuso em SOA, sendo excluídos com base no segundo critério de exclusão definido na seção 2.3.

No próximo capítulo serão apresentadas as considerações finais do mapeamento.

5 CONSIDERAÇÕES FINAIS

Esta seção apresenta as considerações finais deste estudo, divididas em limitações do estudo, trabalhos relacionados, conclusões obtidas com o trabalho e propostas de trabalhos futuros.

5.1 Limitações e Ameaças a Validade

O estudo possui como principal limitação o viés decorrente da interpretação dos autores na análise dos artigos, como por exemplo a exclusão de trabalhos relevantes. Buscou-se mitigar esta limitação por meio da condução separada do mapeamento por dois pesquisadores e da avaliação das análises realizadas por um terceiro pesquisador. Os termos utilizados nas *strings* de busca também podem ter afetado a quantidade de artigos retornados, consequentemente, artigos primários importantes podem não ter sido analisados. Entretanto procurou-se mitigar este risco através da calibração das *strings* de busca utilizadas. Além disso, como estratégia para diminuir o viés dos pesquisadores foi aplicado um protocolo (Apêndice E) contendo todos os procedimentos e critérios utilizados para a pesquisa.

5.2 Trabalhos Relacionados

Uma investigação semelhante foi realizada por Daund (2012) que realizou um Mapeamento Sistemático da Literatura com o propósito de avaliar os atributos de qualidade de SOA. No entanto Daund (2012) não foca no aspecto de reusabilidade em especial. Shanmugasundaram et al. (2012) também apresentou um trabalho similar no qual avaliou oportunidades de pesquisa relacionadas a reuso em SOA por meio de uma revisão da literatura. No entanto, os autores não citam fatores, estratégias, fases do ciclo de vida e métricas relacionadas ao reuso. Portanto, estes pontos são considerados o diferencial do presente estudo, sendo relevante para pesquisadores e organizações que visam se aprofundar sobre os aspectos da reusabilidade em SOA.

5.3 Conclusão

Este trabalho apresentou um Mapeamento Sistemático da Literatura realizado entre junho de 2013 e janeiro de 2014 com o propósito de identificar: o *estado da arte* neste contexto; os *fatores* que influenciam positivamente e negativamente a reusabilidade, as *estratégias* comumente adotadas para potencializar o reuso em SOA, as *fases do ciclo de vida* em que as estratégias adotadas foram utilizadas e as *métricas* associadas a mensuração desse atributo. A pesquisa foi motivada pela ausência, até o estudado, de um estudo sistemático que agregue aspectos especificamente relacionados a reuso em SOA.

Apesar da quantidade relevante de trabalhos que propõem melhorias para potencializar o reuso e da diversidade de estratégias utilizadas, é perceptível a falta de aplicação real das propostas. Neste sentido, dos 24 artigos mapeados, dois [P37, P44] se destacaram pela completude, consistência e por apresentarem a validação das metodologias criadas por meio de estudos de caso. Porém, até o estudado, os trabalhos restantes não indicam que a proposta foi validada. Há carência também quanto a identificação de fatores que influenciam o reuso em SOA, visto que 33% (8; 24) dos artigos não explicitam este elemento. Referente às fases do ciclo de vida, foi identificado que as fases de implementação, automação e monitoramento de SOA são escassas de métodos que potencializem o reuso. O baixo número de trabalhos que definem métricas para mensurar e melhorar o reuso também consiste em uma lacuna na reusabilidade em SOA. Neste sentido, a área apresenta uma rica variedade de assuntos a serem investigados por pesquisadores que possuem interesse no tema.

5.4 Trabalhos Futuros

Como a investigação sobre reuso em SOA no presente trabalho englobou apenas artigos acadêmicos, é sugerido que as questões de pesquisa definidas na Seção 2.1 sejam respondidas por meio de um estudo empírico que investigue fatores, estratégias e métricas utilizadas para potencializar o reuso em SOA na prática. O estudo poderá ser realizado por meio de estudo de caso ou *survey* em organizações reais. Os resultados possibilitarão uma análise comparativa da reusabilidade em SOA na academia e na indústria.

5.5 Produção Bibliográfica

Este trabalho também resultou em um artigo que foi aceito para publicação no Simpósio Brasileiro de Sistemas de Informação (SBSI) no ano de 2014. O artigo foi escrito por mim, pelo Prof. Msc. José Jorge Lima Dias Junior da Universidade Federal da Paraíba e pela Msc. Joyce Aline Oliveira da Universidade Federal de Pernambuco.

REFERÊNCIAS BIBLIOGRÁFICAS

- Alferez, G. H. e Pelechano, V. (2011). "Systematic reuse of *Web Services* through *software product line engineering*". 9th IEEE European Conference on *Web Services* (ECOWS).
- Almeida, E. S., Alvaro, A., Garcia, V. C., Mascena, J. C. C. P., Burégio, V. A., Nascimento, L. M., Lucrédio, D., Meira, S. L. (2007) C.R.U.I.S.E: Component Reuse in *Software Engineering*. C.E.S.A.R e-book, Recife, 1st edition.
- Arksey, H.; O'malley, L. (2005) Scoping studies: towards a methodological framework. *International Journal of Social Research Methodology*, v. 8, n. 1, p. 19-32. doi: 10.1080/1364557032000119616.
- Basili, V. R.; Briand, L. C.; Melo, W. L. (1996) How reuse influences productivity in object-oriented systems. In: *Communications of the ACM*, Vol. 39, No. 10, pp. 104-116.
- Basili, V. R.; Rombach, H. D. (1991). Support for Comprehensive Reuse, In: *Software Engineering Journal, Special issue on software process and its support*, Vol. 06, No. 05, pp. 306-316.
- Bloomberg, Jason and Ronald Schmelzer. (2006). *Service Orient or Be Doomed! How Service Orientation Will Change Your Business*. Hoboken, NJ: John Wiley & Sons.
- Brahe, S. BPM on top of SOA: Experiences from the financial industry (2007) *Lecture Notes in Computer Science*, 4714 LNCS, pp. 96-111.
- Budgen, David. (2008). Turner, Mark; Brereton, Pearl; and Kitchenham, Barbara. Using Mapping Studies in *Software Engineering*, *Proceedings of Psychology of Programming Interest Group (PPIG)*, Lancaster University, pp. 195-204.
- Dias, J., Oliveira, J., Meira, S. (2012) "Pontos Chaves para Adoção de Uma Arquitetura Orientada a Serviços: Uma Análise Comparativa de Modelos de Maturidade SOA da Indústria". *Simpósio Brasileiro de Sistemas de Informação*, São Paulo.
- Dias, J., Oliveira, J., Meira, S. (2013) "Estudo Empírico sobre Adoção de SOA: Um Mapeamento Sistemático da Literatura". *Simpósio Brasileiro de Qualidade de Software*, Bahia.
- DIAS JR., J. SOA e seus Atributos de Qualidade - Entenda os atributos de qualidade em uma Arquitetura Orientada a Serviços. *Engenharia de Software Magazine*, 06 abr. 2010.
- DIAS, J. A *Software Architecture Process for SOA Definition - Designing Service-Oriented Architectures in an Enterprise Context*. LAP Lambert Academic Publishing. February, 2010.
- DIAS, J. *Arquitetura Orientada a Serviços - Sobre o que você precisa refletir para adotá-la em um contexto empresarial*. *Engenharia de Software Magazine*. Edição 22. Março, 2010.
- Endrei, M., Ang, J., Arsanjani, A., Chua, S., Comte, P., Krogdahl, P., Luo, M., Newling, T. (2004). *Patterns: Service-Oriented Architecture and Web Services*, IBM International Technical Support Organization. Redbooks.
- Erickson, J., Siau, K. (2008). *Web Services*, service-oriented computing, and service-oriented architecture: separating hype from reality. *Journal of Database Management* 19 (3), 42-54.

- Erl, T. (2005) *Service Oriented Architecture: Concepts, Technology and Design*. Prentice-Hall, Englewood Cliffs.
- Ezran M.; Morisio, M.; Tully, C. (2002) *Practical Software Reuse*, *Springer*.
- Furtado, C., Pereira, V., Azevedo, L., Baião, F., Santoro, F. (2009). *Arquitetura Orientada a Serviço – Conceituação*. Relatórios Técnicos do DIA/UNIRIO, No. 00012/2009.
- Frakes, W.B.; Isoda, S. (1994) Success Factors of Systematic *Software Reuse*, In: *IEEE Software*, Vol. 12, No. 01, pp. 15-19.
- Geetha J. and Karthikeyan T. (2013). A Tool for Measuring SOA Service Granularity. *IJCA Special Issue on International Conference on Communication, Computing and Information Technology ICCCMIT(3):17-21*.
- Josuttis, N. (2007). “SOA in Practice - The Art of Distributed System Design.” O'Reilly Media.
- Karthikeyan, T., Geetha, J. (2012). A Study and Critical Survey on Service Reusability Metrics. *IJITCS*, vol.4, no.5, pp.25-31.
- Kitchenham, B. A. (2007) Guidelines for performing Systematic Literature Reviews in Software Engineering, Version 2.3, EBSE Technical Report, EBSE-2007-01, 2007.
- Kitchenham, Barbara; Mendes, Emilia and Travassos, Guilherme. (2006). “A systematic review of cross vs within company cost estimation studies”, *Proceedings EASE*, 2006, pp. 89-98.
- Krueger, C.W. (1992) *Software Reuse*, In: *ACM Computing Surveys*, Vol. 24, No. 02, pp. 131-183.
- Mili, H.; Mili, F.; Mili, A. (1995). Reusing Software: Issues and Research Directions, In: *IEEE Transactions on Software Engineering*, Vol. 21, No. 06, pp. 528-562.
- Petersen, Kai; Feldt, Robert; Mujtaba, Shahid e Mattsson, Michael. (2008). “Systematic Mapping Studies in Software Engineering”, 12th International Conference on Evaluation and Assessment in Software Engineering (EASE), University of Bari, Italy, 26 - 27.
- Petersen, K.; Feldt, R.; Mujtaba, S.; Mattsson, M. (2007). Systematic Mapping Studies in Software Engineering. , p. 1-10.
- Prieto-Diaz, R.; Freeman, P. Classifying *Software* for Reusability, In: *IEEE Software*, Vol. 04, No. 01, January, 1987, pp. 06-16.
- Sametinger, J. (1997). *Software Engineering with Reusable Components*, *Springer- Verlag*, pp.275.
- Shanmugasundaram, G. Prasanna Venktesan, V. Punitha Devi, C. (2013). “Modeling Measures for Service Interpretation in Discoverability of Service Oriented Architecture” *Procedia - Social and Behavioral Sciences*, Volume 73, 27 February, Pages 128-135, ISSN 1877-0428
- Sommerville, Ian. (2011) *Engenharia de Software* 9. Ed. - São Paulo. Pearson Pretice Hall.

Svanidzaite, Sandra. (2012) A Comparison of SOA Methodologies Analysis & Design Phases. DB&Local Proceedings, volume 924 of CEUR Workshop Proceedings, page 202-207. CEUR-WS.org,

Yoon, T., Carter, P.E. (2007). Investigating the antecedents and benefits of SOA implementation: a multi-case study approach. In: Proceedings of the 13th Americas Conference on Information Systems (AMCIS). Keystone, CO, USA, pp. 1–11.

Apêndice A – Artigos incluídos

ID	Artigo	Conferência	Ano	Origem	Q1 - Fatores que influenciam	Q2a - Macro estratégia	Q2b - Micro estratégia	Q3 - Fase do ciclo de vida	Q4 - Métricas	Q5 - Método de pesquisa
p1	3D architecture viewpoints on service automation	Journal of Systems and Software 86	2013	Holanda; Reino Unido; Espanha	Não identificado	Arquitetura 3D para a Automação de Serviços (Decision, Degree e Data)	Desenvolvimento e utilização do sistema SDQA (Desenvolvimento de Serviços e Configuração da Arquitetura) com base na Arquitetura 3D.	Automação de serviços	n/t	Pesquisa-Ação, Estudo de caso
p2	An approach for e-Commerce on-demand service-oriented product line development	Acta Polytechnica Hungarica 10	2013	Nigéria; África do Sul; Turquia	Redução de custo e aumento do time-to-market de serviços no contexto de e-commerce	Reutilização de serviços através da Hibridização de Linha de Produtos de Software com SOA: SOPL	Desing e análise de serviços (SOAD) com base nos conceitos Arquitetura de Linha de Produtos Orientada a Serviços (SOPLA)	Identificação, especificação e desenvolvimento de serviços	n/t	Survey, Estudo de caso
p5	The influence of SOA governance mechanisms on IT flexibility and service reuse	Journal of Strategic Information Systems 22	2013	Alemanha	Governança SOA(qualificação dos funcionários, Uso de padrões, Aplicação de normas comuns, Implementação de processos de gestão de serviços e Modularidade)	Governança SOA	Aplicação de mecanismos de Governança SOA e modularidade	n/t	n/t	Survey
p6	A pragmatic approach for analysis and design of service inventories	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 7221	2012	Holanda	Mapeamento definindo os tipos de serviços de um serviço de negocio	Projetar bem serviços para serem utilizados	Utilizar modelagem UML para arquitetar os serviços	Análise e Projeto	n/t	Proposta de metodologia

p11	Integrated asset analysis framework for model-driven development of SOA based solutions	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 7221	2012	Índia	Modelo de referencia de analise integrado de ativos reutilizáveis de múltiplos repositórios nas diferentes camadas de abstração da solução	Desenvolvimento orientado a modelos de soluções SOA	Estrutura integrada descoberta de ativos e análise para auxiliar um modelo orientado o desenvolvimento de soluções baseadas em SOA.	Analise	n/t	Proposta de metodologia
p14	Promoting reuse in web services by managing variability	Proceedings of 2012 International Conference on Multimedia Computing and Systems	2012	Marrocos	Gestão de variabilidade de serviços	Utilizar variabilidade	Utilizar MDA para instanciar as variabilidades escolhidas	Desenvolvimento	n/t	Proposta de modelo
p15	ReputationNet: A Reputation Engine to Enhance ServiceMap by Recommending Trusted Services	SCC '12 Proceedings of the 2012 IEEE Ninth International Conference on Services Computing	2012	Austrália; USA; Reino Unido	Reputação do serviço	Recomendação de serviços	Recomendação baseada em reputação e uso real do serviço	Projeto	n/t	Proposta de metodologia
p18	Service-oriented domain analysis method	Proceedings - 2012 International Conference on Computer Science and Service System	2012	China	Recursos de Analise de Domínio	Analise de domínio em serviços	Combinar serviço e análise de domínio orientada características, elaborando análise de domínio orientada a serviços e processo de modelagem, bem como as diversas fases dos modelos de domínio	Analise	n/t	Estudo de caso

p19	The implementation experience of an advanced service repository for supporting service-oriented architecture	38th Latin America Conference on Informatics	2012	Brasil	Metadados utilizados em um repositório	Utilizar repositório de serviço	uma ferramenta para facilitar a gestão, a persistência, colaboração e pesquisa de serviços e sistemas disponíveis no catálogo/repositório.	Projeto	n/t	Pesquisa-Ação
p37	Registry and discovery of services with variability based on 2-level UDDI	Proceedings - 9th IEEE International Symposium on Parallel and Distributed Processing with Applications Workshops	2011	Coréia	n/t	reusabilidade em linha de produtos de software	Registro e descoberta de serviço em UDDI de 2 níveis	Requisitos e implantação	n/t	Proposta de modelo, Estudo de caso
p40	Systematic reuse of web services through software product line engineering	9th IEEE European Conference on Web Services	2011	México; Espanha	Utilizar os princípios da engenharia de SPL e MDD	Reusabilidade em linha de produtos de software	Uso do mapeamento entre um modelo de recurso e um modelo BPMN e os recursos selecionados no modelo de recursos para criar um modelo BPMN derivado que representa uma versão da composição de serviços.	Análise e projeto	n/t	Proposta de método
p43	Adapter patterns for resolving mismatches in service discovery	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 6275	2010	Coréia	Padrões adaptadores para evitar incompatibilidade na descoberta de serviços	Incompatibilidade de serviços	Utilização de padrões adaptadores para evitar a incompatibilidade na descoberta de serviços	Análise e projeto*	n/t	Proposta
p44	An approach to developing reusable domain services for service oriented applications	Proceedings of the ACM Symposium on Applied Computing	2010	Coréia	Abordagem de desenvolvimento baseada em recursos	Utilizar análise de variabilidade	Utilizar modelagem orientada a features para identificar serviços	Análise/projeto	n/t	Proposta, Estudo de caso

p48	Bridging Software Product Lines and Service-Oriented Architectures for Service Identification Using BPM and FM	ICIS '10 Proceedings of the 2010 IEEE/ACIS 9th International Conference on	2010	Coréia	Identificação de serviços	Identificação de Serviços	um método de identificação de serviço que une os modelos recursos (FMS) de SPLs e modelos de processos de negócios (BPMs) em SOA	n/t	n/t	proposta
p54	Reusing legacy systems in a service-oriented architecture: A model-based analysis	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 6413	2010	Colômbia	Engenharia Orientada a Modelo(MDE) e Domain Specific Languages (DSL)	análise da reutilização de sistemas legados em arquiteturas SOA	Uso de engenharia orientada a modelos (MDE) e abordagem do uso de Domain Specific Languages (DSL) para representar e analisar os serviços candidato SOA e sistemas legados	Projeto e implementação	n/t	Proposta
p58	SOPLE-DE: an approach to design service-oriented product line architectures	Proceedings of the 14th international conference on Software product lines: going beyond	2010	Brasil	Processo para desenhar linhas de produtos orientadas a serviços	Linha de produto de software orientada a serviços	uma abordagem para fornecer orientação para a equipe, especificar os artefatos a serem produzidos , e as atividades associadas com funções específicas e da equipe como um todo	Projeto e desenvolvimento	n/t	Proposta de abordagem
p60	Towards a Variability Model for SOA-Based Solutions	Proceedings of the 2010 IEEE International Conference on Services Computing	2010	Índia	Não identificado	Metodologia baseada em Engenharia Orientada a Variação para identificar e controlar variâncias de serviços	Utilização de uma técnica denominada Design de Serviço Orientado à Variação (VOSD) para descobrir e derivar automaticamente variâncias de serviços a partir de um repositório reusável.	Design de serviços	Não identificado	Proposta de metodologia

p64	A metric for composite service reusability analysis	Proceedings of the 2010 ICSE Workshop on Emerging Trends in Software Metrics	2009	Irã	Custo	Utilização de uma abordagem baseada em Probabilidade de Descrição e Incompatibilidade Lógica para mensurar a reusabilidade de um serviço composto.	Definição de uma métrica quantitativa para mensurar a reusabilidade de serviços compostos em um contexto potencial com base na medição e análise lógica e probabilística do nível de reuso.	Monitoramento	reusable variant, which is *Max Rp: Variante reusável; *MP SD : probabilidade da descrição de um serviço; *MMP SD : incompatibilidade probabilística da descrição de um serviço.	Proposta de abordagem, Estudo de caso.
p67	A service lifecycle and information model for service-oriented architectures	Computation World: Future Computing, Service Computation, Adaptive, Content, Cognitive, Patterns, ComputationWorld 2009	2009	Áustria	um modelo de ciclo de vida de serviços para SOA	Modelo de Processo SOA	modelo de ciclo de vida de serviços e informações para SOA	n/t	n/t	Proposta de metodologia
p69	An approach to enhancing reusabilities in service development	ACM International Conference Proceeding Series 321	2009	Coréia	Projeto e desenvolvimento de serviço baseado em SPL	Reusabilidade com Linha de Produtos	Um projeto e Desenvolvimento de serviços baseado em serviços de domínio	Análise		Proposta de metodologia
p74	Design Web services: Towards service reuse at the design level	Journal of Computers 4	2009	China	Modelo de design de serviço	Metodologia de projeto de serviços	uma abordagem de design de serviços orientada a padrões que consiste em três fases : análise de negócios, modelagem de serviços e desenvolvimento de componentes de software.	Design	n/t	Proposta de metodologia

p86	A Formal Approach to Devising a Practical Method for Modeling Reusable Services	Proceedings of the 2008 IEEE International Conference on e-Business Engineering	2008	Coréia	processo sistemático de identificação e concepção de serviços reutilizáveis.	Engenharia de Reutilização	um processo sistemático de identificação e concepção de serviços reutilizáveis com cinco atividades (Identificação do processo de negocio, Especificação do processo de negocio, Identificação de serviços reutilizáveis, especificação de serviços reutilizáveis e projeto de serviços reutilizáveis)	Análise e projeto	n/t	Proposta
p99	Variation-oriented engineering (VOE): Enhancing reusability of SOA-based solutions	IEEE International Conference on Services Computing	2008	Índia	Modelagem da variabilidade	Variabilidade em SOA	conjunto de mecanismos chamado Engenharia Orientada a Variação (VOE)	Análise/Projeto/Implementação	n/t	Proposta
p110	Building business processes or assembling service components: Reuse services with BPEL4WS and SCA	ECOWS 2006: Fourth European Conference on Web Services	2006	China	Combinação de BPEL4WS e SCA	Reutilização de serviços	padrões de reutilização com a combinação de BPEL4WS e SCA	Desenvolvimento	n/t	n/t

Apêndice B – Artigos excluídos

ID	Título
p3	Daniele Romano. Analyzing the change-proneness of service-oriented systems from an industrial perspective. Proceedings of the 2013 International Conference on <i>Software Engineering</i> , ICSE 2013.
p7	Chien-Hsiang Lee; San-Yih Hwang; I-Ling Yen. A service pattern model for flexible service composition. IEEE 19th International Conference on <i>Web Services</i> , 2012
p8	Benno Albert; Rodrigo Santos, Cláudia Werner. A study on software components governance based on SOA governance elements. Sixth Brazilian Symposium on <i>Software Components, Architectures and Reuse</i> , 2012.
p9	Lucas Francisco da Matta Vegi, Douglas Alves Peixoto, Liziane Santos Soares, Jugurta Lisboa-Filho, and Alcione de Paiva Oliveira. An infrastructure oriented for cataloging services and reuse of analysis patterns. Lecture Notes in Business Information Processing Volume 100 LNBIP, Issue PART 2, 2012, Pages 338-343. 9th International Conference on Business Process Management, BPM 2011.
p10	Olaf Zimmermann. Architectural decision identification in architectural patterns. 10th Working IEEE/IFIP Conference on <i>Software Architecture</i> , WICSA 2012.
p12	Sneed, H.M. ; Anecon GmbH, Vienna, Austria ; Schedl, S. ; Sneed, S.H. Linking legacy services to the business process model. IEEE 6th International Workshop on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA), 2012.
p13	Chong Wang; Keqing He; Wen Zhu; Zaiwen Feng. Personalized reuse of business process through the metamodel for process model registration. 9th International Conference on Grid and Cooperative Computing (GCC), 2010.
p16	Shanmugasundaram, G; Venkatesan, V.Prasanna; Punitha Devi, C. Research opportunities in service reusability of service oriented architecture. International Conference on Emerging Trends in Science, Engineering and Technology (INCOSSET), 2012.
p17	Mcheick, H.; Dodge, S.; Karam, M. Service oriented specification and study of quality performance attribute. International Conference on Communications and Information Technology (ICCIT), 2012.
p21	Leotta, M.; Reggio, G.; Ricca, F.; Astesiano, E. Towards a lightweight model driven method for developing SOA systems using existing assets. International Symposium on Web Systems Evolution (WSE), 2012.
p23	Zhang, Y.; Fu, W.; Chen, L.; Xu, B. A formal model of service computing and its applications on service discovery. 9th International Conference on <i>Web Services</i> (ICWS), 2011.
p24	Taher, Y.; Haque, R.; Parkin, M.; Van Den Heuvel, W.-J.; Richardson, I.; Whelan, E. A multi-layer approach for customizing business services. 12th International Conference on Electronic Commerce and Web Technologies (EC-Web), 2011.
p25	Mannava, V.; Ramesh, T. A novel event based autonomic design pattern for management of webservices. 1st International Conference on Advances in Computing and Information Technology (ACITY), 2011.
p26	Budan Wu; Junliang Chen. An ontology based service creation process. IEEE International

	Conference on Services Computing (SCC), 2011.
p27	Hwai-Jung Hsu; Feng-Jian Wang. Detecting artifact anomalies in temporal structured workflow as reusable assets . IEEE 35th Annual Computer <i>Software</i> and Applications Conference Workshops (COMPSACW), 2011.
p28	Boškovic, Marko; Gašević, Dragan; Mohabbati, Bardia; Asadi, Mohsen; Hatala, Marek; Kaviani, Nima; Rusk, Jeffrey; Bagheri, Ebrahim. Developing families of <i>software</i> services: A semantic web approach . Journal of Research & Practice in Information Technology, Vol. 43 Issue 3, p179-208. Aug2011.
p29	Bardia Mohabbati; Marek Hatala; Dragan Gašević; Mohsen Asadi; Marko Boškovic. Development and configuration of service-oriented systems families . Symposium on Applied Computing, 2011.
p30	Guo Ying; Zhang Xiao-yan; Wang Jun; Yang Meihong. Domain Service Acquisition and Domain Modeling Based on Feature Model . 14th IEEE International Conference on Computational Science and Engineering, 2011.
p31	Stuart N. Wrigley; Raúl García-Castro; Cássia Trojahn. Infrastructure and workflow for the formal evaluation of semantic search technologies . Workshop on Data infrastru ^{ctur} Es for supporting information retrieval evaluation, 2011.
p32	Sebastian Adam; Oezguer Uenalán; Norman Riegel. Is it beneficial to match reusable services earlier? 17th international working conference on Requirements engineering: foundation for <i>software</i> quality, 2011.
p33	Kazemi, A.; Azizkandi, A.N.; Rostampour, A.; Haghighi, H. Measuring the Conceptual Coupling of Services Using Latent Semantic Indexing . IEEE International Conference on Services Computing (SCC), 2011.
p35	Huy Tran, Uwe Zdun, Schahram Dustdar. Name-based view integration for enhancing the reusability in process-driven SOAs . Lecture Notes in Business Information Processing Volume 66, pp 338-349, 2011.
p36	Jia Zhang; Wei Tan ; Alexander, J. ; Foster, I. Recommend-As-You-Go: A Novel Approach Supporting Services-Oriented Scientific Workflow Reuse . IEEE International Conference on Services Computing (SCC), 2011.
p38	Noyan Ilk; J. Leon Zhao; Paulo Goes; Paul Hofmann. Semantic enrichment process: An approach to <i>software</i> component reuse in modernizing enterprise systems . Journal Information Systems Frontiers, Volume 13 Issue 3, July 2011.
p39	George Feuerlicht. Simple metric for assessing quality of service design . Lecture Notes in Computer Science Volume 6568, pp 133-143, 2011.
p41	Joonseok Park; Mikyeong Moon; Keunhyuk Yeom. Variability modeling to develop flexible service-oriented applications . Journal of Systems Science and Systems Engineering, Volume 20, Issue 2, pp 193-216, June 2011.
p42	Liang Bao; Chao Yin; Weigang He ; Jun Ge. Extracting reusable services from legacy object-oriented systems . International Conference on <i>Software</i> Maintenance (ICSM), 2010.
p45	Bispo, C.P.; Maciel, R.S.P.; David, J.M.N.; Ribeiro, Italo. Applying a model-driven process for a collaborative service-oriented architecture . 14th International Conference on Computer Supported Cooperative Work in Design (CSCWD), 2010.
p46	Natalia Kryvinska; Christine Strauss; Lukas Auer; I. Ivanochko. Assessment of business value from services re-use on SOA-based e-business platform . 12th International Conference on Information Integration and Web-based Applications & Services, 2010.

p47	Deng Yue; Li Cui Cui ; Liu Wan Jun ; Feng Wen Xiang. Based on SOA architecture and component <i>software</i> reuse architecture research. The 2nd IEEE International Conference on Information Management and Engineering (ICIME), 2010.
p49	Matthias Galster. Describing variability in service-oriented <i>software</i> product lines. Fourth European Conference on <i>Software</i> Architecture, 2010.
p50	Flavio Mota Medeiros; Eduardo Santana de Almeida; Silvio Romero de Lemos Meira. Designing a set of service-oriented systems as a <i>Software</i> Product Line. Fourth Brazilian Symposium on <i>Software</i> Components, Architectures and Reuse, 2010.
p51	Hock-koon, A.; Oussalah, M. Expliciting a composite service by a metamodeling approach. Fourth International Conference on Research Challenges in Information Science (RCIS), 2010.
p52	Liang Bao; Chao Yin; Weigang He ; Jun Ge. Extracting reusable services from legacy object-oriented systems. International Conference on <i>Software</i> Maintenance (ICSM), 2010.
p53	Bispo, C.P.; Maciel, R.S.P.; David, J.M.N.; Ribeiro, Italo. Applying a model-driven process for a collaborative service-oriented architecture. 14th International Conference on Computer Supported Cooperative Work in Design (CSCWD), 2010.
p55	Ren HongMin; Liu Jin ; Zhang JingZhou. Service-oriented lightweight <i>software</i> asset repository open framework. Seventh International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), 2010.
p56	Lixin Du. SOA-based integration of enterprise vertical applications. International Conference on Computer Application and System Modeling (ICCASM), 2010.
p57	Hassan Gomaa; Koji Hashimoto; Minseong Kim; Sam Malek; Daniel A. Menascé. <i>Software</i> adaptation patterns for service-oriented architectures. ACM Symposium on Applied Computing, 2010.
p61	Elgedawy, I.; Ramaswamy, L. Rapid identification approach for reusable SOA assets using component business maps. IEEE International Conference on <i>Web Services</i> (ICWS), 2009.
p62	Elgedawy, I. Reusable SOA assets identification using e-business patterns. World Conference on Services - II, 2009.
p63	Motahari-Nezhad, H.R.; Jun Li; Stephenson, B.; Graupner, S. Solution reuse for service composition and integration. World Conference on Services, 2009.
p65	Hyun Jung La, Jin Sun Her, Sang Hun Oh, Soo Dong Kim. A practical approach to developing applications with reusable services. Studies in Computational Intelligence Volume 253, pp 95-106, 2009.
p66	Iman Saleh; Gregory Kulczycki; M. Brian Blake. A reusable model for data-centric <i>Web Services</i>. 11th International Conference on <i>Software</i> Reuse: Formal Foundations of Reuse and Domain Engineering, 2009.
p68	Junhuai Li; Yile Wang; Jing Zhang; Zhuobin Zhang. A web service adapter with contract-oriented methodology. Eighth IEEE/ACIS International Conference on Computer and Information Science, 2009.
p70	Chengjun Wang. Architecture based service development for service reuse of varying levels of granularity. PIAGENG 2009: Intelligent Information, Control, and Communication Technology for Agricultural Engineering. 2009.
p71	Zhilei Ma; Leymann, F. BPEL fragments for modularized reuse in modeling BPEL processes. Fifth International Conference on Networking and Services, 2009.
p72	Rong Liu; Wu, F.; Yasodhar Patnaik; Kumaran, S. Business Entities: An SOA Approach to

	Progressive Core Banking Renovation. IEEE International Conference on Services Computing, 2009.
p73	Sheng, Q.Z.; Benatallah, B.; Maamar, Z.; Ngu, A.H.H. Configurable composition and adaptive provisioning of Web Services. IEEE Transactions on Services Computing, Volume 2, Issue 1, 2009.
p75	Pietschmann, S.; Voigt, M. ; Meissner, K. Dynamic Composition of Service-Oriented Web User Interfaces. Fourth International Conference on Internet and Web Applications and Services (ICIW), 2009.
p76	Elgedawy, I.; Ramaswamy, L. Rapid identification approach for reusable SOA assets using component business maps. IEEE International Conference on <i>Web Services</i> (ICWS), 2009.
p77	Kaviani, N.; Mohabbati, B.; Lea, R.; Gasevic, D. ReCoIn: A framework for dynamic integration of remote services into a service-oriented component model. IEEE Asia-Pacific Services Computing Conference (APSCC), 2009.
p78	Elgedawy, I. Reusable SOA assets identification using e-business patterns. World Conference on Services - II, 2009.
p79	Atkinson, C.; Bostan, P.; Fink, F. Reuse-oriented deployment of software components: Congregation in service-oriented development. Fourth International Conference on <i>Software Engineering Advances</i> (ICSEA), 2009.
p80	Gary Shao; Robert McGraw. Service-oriented simulations for enhancing situation awareness. Spring Simulation Multiconference, 2009.
p81	Rong Zhang; Koji Zettsu; Takafumi Nakanishi; Yutaka Kidawara; Yasushi Kiyoki. SOBEX: Distributed service search engine that exploits service collaboration context. 3rd International Universal Communication Symposium, 2009.
p82	Motahari-Nezhad, H.R.; Jun Li; Stephenson, B.; Graupner, S. Solution reuse for service composition and integration. World Conference on Services, 2009.
p83	Valérie Monfort; Slimane Hammoudi. Towards adaptable SOA: Model driven development, context and aspect. Service-Oriented Computing. Lecture Notes in Computer Science Volume 5900, pp 175-189, 2009.
p84	Hadaytullah, H.; Koskimies, K.; Systa, T. Using Model Customization for Variability Management in Service Compositions. IEEE International Conference on <i>Web Services</i> (ICWS), 2009.
p85	Matjaz B. Jurica, Ana Sasa, Bostjan Brumen, Ivan Rozman. WSDL and UDDI extensions for version support in Web Services. Journal of Systems and <i>Software</i> , Volume 82, Issue 8, Pages 1326–1343, August 2009.
p87	Uddam Chukmol. A framework for web service discovery: service's reuse, quality, evolution and user's data handling. 2nd SIGMOD PhD workshop on Innovative database research, 2008.
p88	He Yuan Huang; Hua Fang Tan; Jun Zhu; Wei Zhao. A lightweight approach to partially reuse existing component-based system in service-oriented environment. 10th international conference on <i>Software Reuse: High Confidence Software Reuse in Large Systems</i> (ICSR), 2008.
p89	Budan Wu; Zhi Jin; Bin Zhao. A modeling approach for service-oriented application based on extensive reuse. IEEE International Conference on <i>Web Services</i> (ICWS), 2008.
p90	Si Won Choi; Soo Dong Kim. A quality model for evaluating reusability of services in SOA. 10th IEEE Conference on E-Commerce Technology and the Fifth IEEE Conference on Enterprise Computing, E-Commerce and E-Services, 2008.
p91	Jianwu Wang, Jian Yu, Paolo Falcarin, Yanbo Han, Maurizio Morisio. An approach to domain-

	specific reuse in service-oriented environments. High Confidence <i>Software Reuse</i> in Large Systems. Lecture Notes in Computer Science Volume 5030, 2008.
p92	Ilk, N.; Zhao, J.L.; Hofmann, P. On Reuse of Source Code Components in Modernizing Enterprise Systems. IEEE Symposium on Advanced Management of Information for Globalized Enterprises (AMIGE), 2008.
p93	Wang Chengjun. Pattern oriented service development for coarse-grained service reuse. International Symposium on Knowledge Acquisition and Modeling (KAM), 2008.
p94	Kalapriya Kannan; Biplav Srivastava. Promoting reuse via extraction of domain concepts and service abstractions from design diagrams. IEEE International Conference on Services Computing (SCC), 2008.
p95	Budan Wu; Zhi Jin. Service-oriented modeling: An extensive reuse method. 32nd Annual IEEE International Computer <i>Software</i> and Applications (COMPSAC), 2008.
p96	Dan, A. , Johnson, R.D. , Carrato, T. SOA service reuse by design. 30th International Conference on <i>Software Engineering</i> (ICSE), 2008.
p97	Kuster, M.W.; Ludwig, C. Software Reuse through Resource Registries. 19th Australian Conference on <i>Software Engineering</i> , 2008.
p98	Narendra, N.C.; Ponnalagu, K.; Srivastava, B.; Banavar, G.S. Variation-oriented engineering (VOE): Enhancing reusability of SOA-based solutions. IEEE International Conference on Services Computing, 2008.
p100	Xiaoxia Cao; Huaikou Miao; Qingguo Xu. Verifying Service-Oriented Requirements Using Model Checking. IEEE International Conference on e-Business Engineering (ICEBE), 2008.
p101	Paola Inverardi; Massimo Tivoli. A reuse-based approach to the correct and automatic composition of web-services. International workshop on Engineering of <i>software</i> services for pervasive environments, 2007.
p102	Soo Ho Chang; Soo Dong Kim. A Service-Oriented Analysis and Design Approach to Developing Adaptable Services. IEEE International Conference on Services Computing, 2007.
p103	Soo Ho Chang; Soo Dong Kim. A variability modeling method for adaptable services in service-oriented computing. 11th International <i>Software</i> Product Line Conference, 2007.
p104	Oliveira, M.; Goncalves, E.M.; Bacili, K.R. Automatic Identification of reusable Software development assets: Methodology and tool. IEEE International Conference on Information Reuse and Integration, 2007.
p105	George Feuerlicht; Amalka Wijayaweera. Determinants of service reusability. Proceedings of sixth conference on New Trends in <i>Software</i> Methodologies, Tools and Techniques: Proceedings of the sixth (SoMeT), 2007.
p106	Olaf Zimmermann; Thomas Gschwind; Jochen Küster; Frank Leymann; Nelly Schuster. Reusable architectural decision models for enterprise application development. Proceedings of the Quality of <i>software</i> architectures 3rd international conference on <i>Software</i> architectures, components, and applications (QoSA), 2007.
p107	Grace Lewis, Edwin J. Morris, Dennis B. Smith, Soumya Simanta. SMART: Analyzing the reuse potential of legacy components in a service-oriented architecture environment. Conference and Exhibit American Institute of Aeronautics and Astronautics (AIAA), 2007.
p109	Lewis, G.; Morris, E.; Smith, D. Analyzing the Reuse Potential of Migrating Legacy Components to a Service-Oriented Architecture. Proceedings of the 10th European Conference on <i>Software</i> Maintenance and Reengineering, 2006.

p111	Zhuopeng Zhang; Hongji Yang; William C. Chu. Extracting reusable object-oriented legacy code segments with combined formal concept analysis and slicing techniques for service integration. Proceedings of the Sixth International Conference on Quality <i>Software</i> , 2006.
p112	Tan, P.S.; Lee, E.W. Standardisation of Functionality for Widespread Reuse of <i>Software</i> as Services. IEEE International Conference on Industrial Informatics, 2006.
p113	Haibin Zhu. Building reusable components with service-oriented architectures. IEEE International Conference on Information Reuse and Integration, 2005.
p114	Haibin Zhu. Challenges to reusable services. IEEE International Conference on Computing, 2005.
p115	R. Todd Stephens. Implementation of enterprise reuse. Proceedings of the 43rd annual Southeast regional conference - Volume 1, Pages 7-8. New York, USA. 2005.
p116	Lewis, G.; Morris, E.; Smith, D. Service-Oriented Migration and Reuse Technique (SMART). 13th IEEE International Workshop on <i>Software</i> Technology and Engineering Practice, 2005.
p117	Mazen M. Aquil Saleh. <i>Software</i> product line engineering based on <i>Web Services</i>. Doctoral Dissertation. George Mason University Fairfax, VA, USA 2005.

Apêndice C – Artigos inacessíveis

ID	Título
p4	Coscia, J. L. O.; Mateos, C.; Crasso, M.; Zunino, A. Anti-pattern free code-first <i>Web Services</i> for state-of-the-art Java WSDL generation tools . Journal International Journal of Web and Grid Services, Volume 9, Issue 2, May 2013.
p20	Liu, Y. The study on <i>software</i> reuse based on service composition and dependency-aware service oriented architecture . International Journal of Digital Content Technology and its Applications, Volume 6, Issue 1, Pages 161-168, January 2012.
p22	Park, J.; Moon, M.; Yeom, K. Two-level service-oriented architecture based on product-line . IEICE Transactions on Information and Systems . Volume E95-D, Issue 12, Pages 2971-2981, December 2012.
p34	Rajam, S.; Cortez, R.; Vazhenin, A.; Bhalla, S. Modified MVC-design patterns for service oriented applications . Frontiers in Artificial Intelligence and Applications, Volume 231, Pages 108-126, 2011.
p59	Chukmol, U.; Benharkat, A.-N.; Amghar, Y. Towards a user-oriented framework for web service discovery, reuse and evolution . Fifth International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), 2009.
p108	Anderson, W. What COTS and <i>software</i> reuse teach us about SOA . Sixth International IEEE Conference on Commercial-off-the-Shelf (COTS)-Based <i>Software</i> Systems (ICCBSS), 2007.

Apêndice D – Artigos excluídos que abordam métricas de reuso em SOA

ID	Título
P33	Kazemi, A.; Azizkandi, A.N.; Rostampour, A.; Haghighi, H. Measuring the Conceptual Coupling of Services Using Latent Semantic Indexing . IEEE International Conference on Services Computing (SCC), 2011.
P39	George Feuerlicht. Simple metric for assessing quality of service design . Lecture Notes in Computer Science Volume 6568, pp 133-143, 2011.

Apêndice E – Protocolo do Mapeamento Sistemático

PROTOCOLO

Tópico de pesquisa: Um mapeamento sistemático sobre
Reutilização em Arquiteturas Orientada a Serviços

E1. Introdução

A Arquitetura Orientada a Serviços (SOA) tem sido amplamente utilizada pelas organizações como uma estratégia no alcance da flexibilidade, da integração e da diminuição de custos inerentes ao desenvolvimento de *softwares*.

A adoção de SOA possibilita que funcionalidades sejam utilizadas como serviços na construção de novas aplicações sem a necessidade de reimplementá-las [Josutis 2007] [Choi e Kim 2008]. A esta ação, considerada pela literatura como um dos atributos de qualidade de SOA, dá-se o nome de reusabilidade.

De acordo com Dias et al [2012], a reusabilidade evita a redundância de código, diminui esforços, custos e tempo associados a implementação e ainda aumenta a qualidade das aplicações. Esta afirmação é reforçada por Karthikeyan e Geetha [2012], segundo os quais serviços reutilizáveis aumentam o retorno do investimento (ROI) e também reduzem custos relacionados a projeto, desenvolvimento, teste e manutenção de *softwares*. Segundo Choi e Kim [2008] a reusabilidade também é tida como um dos principais critérios para a avaliação da qualidade de serviços.

Karthikeyan e Geetha [2012] citam como benefícios da reusabilidade em SOA o aumento da produtividade, a minimização da duplicação de serviços e o aumento da confiabilidade destes. Outro benefício citado pelos autores é a capacidade de se criar novos processos de negócio a partir de serviços existentes para atender as necessidades dos clientes e as exigências do mercado.

Devido às vantagens mencionadas, a busca pelo reuso tem sido um dos motivos para a ampla adoção de SOA, visto ser este um dos princípios de projetos orientados a serviços [Shanmugasundaram et al 2012] [Erl 2005]. No entanto, apesar dos benefícios apresentados, a dificuldade em definir quais serviços reutilizar e qual processo adotar para construir serviços reutilizáveis tem sido reportada em pesquisas empíricas no âmbito de SOA [Brahe 2007] [Barkeville 2010] [Trkman et al 2011] [Auer 2011] [Janssen 2011] [Luthria et al 2012].

Apenas disponibilizar um serviço não é fator resultante para que ele seja usado pelos consumidores. Serviços precisam ser desenvolvidos atendendo a necessidade de uma grande quantidade de consumidores de um grupo e não apenas a necessidade de um consumidor, ou seja, um mesmo serviço tem que ser utilizado por muitos usuários [Shanmugasundaram et al., 2012].

De acordo com Shanmugasundaram et al [2012] muitas pesquisas focam qualidades de atributos de SOA, mas não exploram especificamente o atributo de reusabilidade. A investigação da veracidade desta premissa, os benefícios propostos pela reusabilidade e as dificuldades relatadas empiricamente inerentes ao processo de reuso motivaram a realização

de um Mapeamento Sistemático da Literatura com o propósito de identificar: o estado da arte neste contexto; os fatores que influenciam positivamente e negativamente a reusabilidade, as estratégias comumente adotadas para a melhoria contínua do reuso, as métricas associadas a mensuração desse atributo, entre outros aspectos.

A pesquisa é academicamente relevante por não ter sido encontrado, até o estudado, um trabalho semelhante que agregue diferentes aspectos da reusabilidade em SOA através de um Mapeamento Sistemático. Em nível prático, a pesquisa pode ser útil para profissionais e interessados em SOA que visam compreender as características associadas à reusabilidade e não contam com uma revisão estruturada sobre o tema.

E2. Questões de Pesquisa

Para guiar a execução do mapeamento, foram definidas algumas questões de pesquisa. O objetivo, ao final da pesquisa, é responder estas questões.

Q1. Quais são os fatores que influenciam a reusabilidade em SOA?

***Objetivo:** identificar os fatores que tem alguma influência positiva ou negativa na reusabilidade nos serviços ou em SOA como um todo.*

Q2. Quais são as estratégias adotadas para melhorar a reusabilidade em SOA?

***Objetivo:** descobrir as estratégias adotadas para tentar potencializar o reuso em SOA. Por exemplo: utilizar registro de serviços, utilizar técnica de linha de produto, utilizar métricas de avaliação, etc.*

Q3. Qual fase do ciclo de vida de desenvolvimento a estratégia é adotada?

***Objetivo:** identificar em que fase ou ciclo de vida do serviço ou na SOA, onde a estratégia é aplicada. Por exemplo: na fase de requisitos, na fase de implementação do serviço, na fase de planejamento da adoção, etc.*

Q4. Quais são as métricas utilizadas para se medir reusabilidade em SOA?

***Objetivo:** identificar as métricas que foram utilizadas para se medir reusabilidade em SOA.*

Q5. Que método de pesquisa foi utilizado para validar a proposta?

***Objetivo:** identificar a maturidade da proposta escrita no artigo através do método de pesquisa utilizado. Por exemplo: estudo de caso, experimento, survey, etc.*

E3. Processo de Busca

A estratégia usada para construir os termos de busca segue a mesma abordagem utilizada em (Kitchenham et al., 2006) uma vez que é sistematizada e define passos para derivar strings de busca a partir de questões de pesquisa e seus pontos de vista, experts e papers relevantes.

A estratégia é utilizar os operadores booleanos OR e AND para incorporar sinônimos e integrar termos.

SOA: "SOA" / "service oriented architecture" / "SOSE" / "service oriented *software* engineering"

Reuse: "reuse" / "reusability"

String de Busca:

("SOA" OR "service-oriented" OR "service oriented") AND ("reuse" OR "reusability")

O processo de busca será executado em bibliotecas digitais (busca automática).

A tabela 1 abaixo mostra a lista de bibliotecas digitais que serão consideradas na busca automática.

Tabela 1. Bibliotecas digitais

ACM Digital Library	http://portal.acm.org/
IEEE Xplore	http://ieeexplore.ieee.org/
Scopus	http://www.scopus.com/

Strings utilizadas na busca avançada de cada biblioteca digital:

ACM:

(Abstract:(("SOA" or "service oriented" or "service-oriented") and ("reuse" or "reusability" or "reusable"))) or (Title:(("soa" or "service oriented" or "service-oriented") and ("reuse" or "reusability" or "reusable"))))

IEEE Xplore:

("SOA" OR "service-oriented" OR "service oriented") AND ("reuse" OR "reusability")

Scopus:

TITLE-ABS-KEY(("service oriented" OR "service-oriented" OR "SOA") AND ("reuse" OR "reusability" OR "reusable"))

E4. Critérios de Inclusão e Exclusão

Com o objetivo de identificar estudos primários relevantes entre as pesquisas de reusabilidade em SOA, foram definidos os seguintes critérios de inclusão e exclusão.

Critérios de inclusão

1. Estudos primários publicados em conferências e periódicos que discutem como melhorar a reusabilidade em SOA ou em serviços.

Critérios de exclusão

1. Trabalhos que apenas citam reuso como benefício de SOA mas não possuem uma proposta para melhorar ou avaliar a reusabilidade de SOA como um todo ou de serviços;
2. Trabalhos que não tratam diretamente sobre como potencializar a reusabilidade em SOA;
3. Trabalhos que tratam a reusabilidade apenas de um *software* específico;
4. Estudos duplicados. Quando um estudo for publicado em mais de uma conferência, workshop ou periódico, a versão mais completa será considerada;
5. Short papers com menos de três páginas;
6. White papers, relatórios técnicos, livros não serão considerados;
7. Estudos secundários: revisões da literatura, mapeamentos, etc.

E5. Processo de Seleção dos Estudos Primários

O processo de seleção dos papers devem seguir os seguintes passos:

- 1) Executar a busca automática nas bibliotecas digitais utilizando a string de busca definida;
- 2) Aplicar o critério de inclusão no título e no resumo do artigo. Caso seja aceito, registrar o título do artigo, conferência/journal e o ano de publicação em uma planilha;
- 3) Depois que a busca automática for realizada em todas as bibliotecas digitais definidas, deve-se eliminar os artigos duplicados;
- 4) Verificar conflitos de inclusão entre os pesquisadores. Caso um pesquisador inclua um artigo que outro pesquisador não incluiu, utilizar a opinião do terceiro pesquisador para resolver o conflito;
- 5) Realizar análise mais crítica dos artigos pré-selecionados, verificando o resumo, introdução, núcleo da proposta e conclusão aplicando os critérios de exclusão e respondendo as questões de pesquisa definidas.
 - a. Caso o artigo se enquadre em algum dos critérios deve ser marcado para exclusão, registrando na planilha o critério adotado;
 - b. Para responder as questões de pesquisa, deve-se utilizar o método *keywording*, que tem como objetivo ir capturando as palavras-chaves que responde a questão de pesquisa;
 - c. Cada artigo deve ser analisado pelo menos por dois pesquisadores, garantindo maior qualidade e diminuindo o viés;
 - d. A fim de calibrar as análises, será feito um piloto que funcionará da seguinte forma: todos os pesquisadores farão a análise independente de 3 a 5 artigos selecionados e discutirão os resultados. Isso ajudará a calibrar a análise;
 - e. Com as análises realizadas, serão verificados os conflitos de análise entre os pesquisadores. O terceiro pesquisador deve resolver os conflitos;
- 6) Com os conflitos resolvidos, verifica-se a lista final dos artigos selecionados para a escrita do relatório de mapeamento sistemático.

Planilha A

A planilha A deverá ser utilizada para registrar os artigos incluídos através dos critérios de inclusão.

Titulo	Conferencia	Ano

Planilha B

A planilha B é utilizada para registrar o resultado da análise, onde serão verificados os critérios de exclusão ou respondida as questões de pesquisa, dos artigos pré-selecionados.

Título	Conferência	Ano	Excluir?	Q1 - Fatores que influenciam	Q2a - Macro estratégia	Q2b - Micro estratégia	Q3 - Fase do ciclo de vida	Q4 - Métricas	Q5 - Método de pesquisa

Referências do protocolo

Kitchenham, B. A..Guidelines for performing Systematic Literature Reviews in *Software Engineering*, Version 2.3, EBSE Technical Report, EBSE-2007-01, 2007.

Budgen, David; Turner, Mark; Brereton, Pearl; and Kitchenham, Barbara. Using Mapping Studies in *Software Engineering*, Proceedings of Psychology of Programming Interest Group (PPIG), Lancaster University, 2008, pp. 195-204.

Petersen, Kai; Feldt, Robert; Mujtaba, Shahid; and Mattsson, Michael. Systematic Mapping Studies in *Software Engineering*, 12th International Conference on Evaluation and Assessment in *Software Engineering* (EASE), University of Bari, Italy, 26 - 27 June, 2008.

Kitchenham, Barbara; Mendes, Emiliaand Travassos, Guilherme. A systematic review of cross vs within company cost estimation studies, Proceedings EASE, 2006, pp. 89-98.