Modelo de Maturidade de uma Arquitetura Orientada a Serviços

Ângelo Miguel Belchior Tomé Pilão

Dissertação para a obtenção do grau de Mestre em

Engenharia Informática e de Computadores

Orientadores: Prof. André Ferreira Ferrão Couto e Vasconcelos

Prof. Miguel Leitão Bignolas Mira da Silva

Júri:

Presidente: Prof. Ernesto José Marques Morgado

Orientador: Prof. André Ferreira Ferrão Couto e Vasconcelos

Vogal: Prof. Pedro Manuel Moreira Vaz Antunes de Sousa

Março 2015

Resumo

Na atualidade, as organizações têm-se tornado cada vez mais dependentes dos Sistemas de Informação (SI), para melhorar os seus processos de negócio, adquirindo frequentemente novos serviços. As preocupações com a integração das aplicações, são cada vez mais relevantes, pois existe por parte da organização a necessidade de saber quando é adquirido um novo serviço, qual o sistema de informação que o vai suportar e como vai ser integrado com os sistemas já existentes. As integrações nas aplicações por vezes falham, porque não existe por parte da organização a preocupação na alienação entre a arquitetura empresarial e a arquitetura das tecnologias da informação. Existem algumas metodologias criadas para avaliar em que estado de maturidade se encontra a integração dos serviços nas organizações, mas a maioria das *AQ* são muito genéricas. Para abordar este problema, criamos um conjunto de critérios de resposta para cada *AQ* do modelo de maturidade OSIMM para melhorar o seu método de avaliação. Com base nos resultados das *AQ,* podemos obter qual o nível de maturidade da integração dos serviços para cada dimensão e podemos definir a estratégia, *roadmap* que queremos definir para a nossa organização e quais os benefícios dessa implementação, se necessário. Como método de investigação, usaremos o *Design Science Research* (DSRM) e para avaliação dos resultados faremos entrevistas e questionários, tendo também em conta os comentários por parte da comunidade científica e os princípios de *Österle*.

Palavras-chave: Serviços, Sistemas de Informação, Modelos de maturidade, SOA, *Assessment Questions*, Integração de aplicações

Abstract

Nowadays, organizations have becoming more dependent if Information Systems (IS), to improve their business process, frequently acquiring new services. Concerns about application integration are becoming even more relevant, as there is a need, from the organization's part, to know when a new service will be acquired, which information system will support it and how it will support existing systems. Integration in applications sometimes fail because there is no concern in alienation between enterprise architecture and the architecture of information technology. There are some methodologies created to evaluate which maturity level the integration of services are in, but the majority of AQ's are generic. To address this problem, we create of a set of response criteria for each AQ to the OSIMM maturity model to improve its evaluation method. Based on the results of the AQ, we can get the level of integration of services for each dimension and we can define the strategy, the roadmap that we want to set for our organization and what the benefits of this implementation, if necessary. As research method, we will use the Design Science Research (DSRM) and for the evaluation results we will do interviews and questionnaires, also taking into account the comments by the scientific community and the principles of Österle.

Keywords: Services, Information Systems, Maturity Models, SOA, Assessment Questions, Application Integration

Agradecimentos

Gostaria de agradecer a todos os que fizeram parte do meu percurso académico, e que me ajudaram especialmente a concluir esta última jornada, a minha Tese de Mestrado.

Em primeiro lugar à minha família, pelo apoio incondicional e por me terem acompanhado em cada fase e ajudado a avançar em todos os momentos, dando todos os incentivos e partilhando os sucessos.

Ao meu orientador, Professor André Vasconcelos e ao meu co-orientador Professor Miguel Mira da Silva, que me acompanharam e ajudaram sempre com prazer, compromisso e persistência. Agradeço-lhes todo o conhecimento passado, motivação, inspiração e todas as discussões académicas que me deram, conhecimento e experiência e se refletiram nos resultados da minha dissertação.

A todos os profissionais de empresas e intervenientes, que me receberam e ajudaram a desenvolver a minha investigação no caminho certo, passando toda a sua experiência e conhecimento para que conseguisse resultados interessantes na minha investigação.

Aos meus colegas de investigação, futuros engenheiros, pelos momentos passados, pela motivação e pequenas contribuições que fizeram com que a minha investigação se tornasse mais rica.

A todos os meus amigos nomeadamente o Daniel Rebelo, Tiago Pinto e o Michel Mendes por me ajudarem a ter momentos extra-investigação que foram muito importantes.

À minha namorada, pelo apoio incondicional e por me ajudar a focar no que é mais importante, assumindo comigo as minhas conquistas e derrotas e ajudando-me a ultrapassar cada etapa.

Índice

Resumo iv

Abstract vi

Agradecimentos viii

Índice x

Lista de figuras xiii

Lista de Tabelas xiv

Lista de Siglas xvi

1. Introdução 18

1.1. Método de Investigação 20

1.2. Estrutura da Dissertação 21

2. Problema 23

2.1. Motivação 23

2.2. Problema da investigação 23

3. Trabalho Relacionado 25

3.1. Web Services 25

3.2. Middleware 26

3.3. Service Oriented Architecture 28

3.3.2. Os princípios de SOA 29

3.4. Modelos de Maturidade 29

3.5. Modelos de Maturidade SOA 30

3.5.1. Oracle Maturity Model 31

3.5.2. OSIMM V2 31

3.5.3. SOAMM Microsoft 33

3.5.4. CBDI 34

3.5.5. Comparação ente modelos 35

3.6. Conclusão 36

4. Objectivos da solução 39

5. Proposta 41

5.1. Objectivos da Proposta 41

5.2. Método de Avaliação 42

6. Demonstração 46

6.1. Estruturação do Modelo 46

6.2. Ordem dos Advogados 47

6.3. Análise dos resultados da *assessment* geral 53

6.4. Resumo 54

7. Avaliação 56

7.1. Entrevistas a Profissionais 56

7.1.1. Resultados da avaliação 56

7.2. Avaliação das Demonstrações 58

7.3. Moody and Shanks 58

7.3.1. Resultados da Avaliação 59

7.4. Princípios de Österle, et al. 60

7.4.1. Resultados da Avaliação 60

8. Comunicação 63

9. Conclusão 65

9.1. Lições Aprendidas 66

9.2. Limitações 67

9.3. Contribuições 67

9.4. Trabalho futuro 68

Bibliografia 70

Anexos **76**

Anexo A – Bibliografia utilizada para a criação dos critérios das AQ do modelo OSIMM **76**

Anexo B – Critérios de resposta para as AQ do modelo OSIMM **77**

Business Dimension **77**

Organization & Governance Dimension **79**

Method Dimension **82**

Application Dimension **84**

Architecture Dimension **87**

Information Dimension **89**

Infrastructure Dimension **93**

Anexo C – Formulário do Assessment **97**

Lista de figuras

Figura 1 - O processo DSRM (adaptado de (Peffers, Tuunanen, & Rothenberge, A Design Science Research Methodology for Information Systems Research, 2008)) 20

Figura 2- SOAP, WSDL, UDDI, BPEL..... 25

Figura 3– Implementação dos *application services* pelo core dos *WS standards* (Erl, 2005)..... 26

Figura 4- Comunicação de aplicações em SILO'S através de troca de mensagens com o uso de um middleware (Papazoglou, 2007)..... 27

Figura 5 - OSIMM Maturity Matrix (Group, The Open Group Architecture Framework (TOGAF), 2007) 42

Figura 6- Fases propostas para a construção do método de avaliação 43

Figura 7- *Assessment Business Dimension*..... 47

Figura 8- Assessment Organization & Governance 48

Figura 9– Assessment da Method Dimension..... 48

Figura 10 -Assessment Application Dimension 49

Figura 11- Assessment Architecture Dimension 49

Figura 12 – Assessment *Information Dimension* 51

Figura 13 – Assessment Infrastructure & Management Dimension..... 51

Figura 14 - Nível de maturidade atingida na OA..... 52

Figura 15 - *Assessment* das organizações 53

Figura 16 - Análise do resultado das entrevistas 57

Lista de Tabelas

Tabela 1 –Modelos de maturidade para organizações que adoptam SOA 30

Tabela 2- Exemplo de Capability /Maturity..... 35

Tabela 3- Critérios elaborados para as perguntas 1 e 2 da *Business Dimension* 44

Tabela 4 – Respostas do assessment Business Dimension 47

Tabela 5 - Respostas do assessment O&G 48

Tabela 6 - Respostas do assessment Method Dimension..... 49

Tabela 7 - Respostas do assessment Application Dimension 50

Tabela 8 - Respostas do assessment Architecture Dimension..... 50

Tabela 9 - Respostas do assessment Information Dimension..... 51

Tabela 10 - Respostas do assessment Infrastructure & Management Dimension 52

Lista de Siglas

SOA	Service Oriented Architecture
OSIMM	OpenGroup Services Integration Maturity Model
AE	Arquitetura Empresarial
SI	Sistema de Informação
AQ	Assessment Questions
DSRM	Design Science Research Methedology
CMM	Câmara Municipal de Mirandela
OA	Ordem dos Advogados
XML	eXtensible Markup Language
SOAP	Simple Object Access Protocol
MOM	Message Oriented Middleware
CMMI	Capability Maturity Model Integration
IT	Information Tecnhologies

1. Introdução

As organizações estão a tornar-se cada vez mais dependentes dos SI para melhorar os seus processos de negócio, de modo a facilitar a tomada de decisões e implementar novas estratégias de negócio. Muitas das organizações necessitam de avaliar o estado de maturidade dos seus serviços e os gastos na aquisição de novos serviços.

A integração heterogénea de aplicações IT é um problema cada vez maior, dentro e entre organizações (M., 2001). Por vezes, muitas empresas que adotam uma arquitetura orientada a serviços (**S**ervice **O**riented **A**rchitecture) falham, porque adquirem novos serviços ou desenvolvem aplicações que correm em hardware diferentes. Estas integrações falham porque as empresas não tentam perceber a estratégia organizacional da empresa e a sua arquitetura empresarial (AE), visão, objetivos bem como a sua direção (Mahajan, 2006). Não existe um alinhamento entre o *Information Technology Governance* e *Enterprise Architecture Governance* que satisfaça as suas necessidades. (Erl, 2005)

Estas preocupações estão a ganhar cada vez mais importância para as organizações, pois necessitam de ter os seus serviços integrados e não ter uma organização baseada em SILOS, i.e. departamentos. Necessitam de reduzir os custos e ter uma visão mais sistémica da sua estratégia de negócio.

Uma organização que pretenda um *true SOA* tem que adotar paradigmas e *frameworks* para analisar como as organizações são *structed, governed, maintened, funded and used* (Inaganti & Aravamudan , 2007). Uma organização, antes de adotar o SOA, deve fazer uma análise da sua AE quase em *bottom-up* (Erl, 2005). Não sendo essa análise efetuada, existe um risco maior para o desenvolvimento da organização, como consequência de termos alguns processos de negócio mais evoluídos que outros, podendo ser um obstáculo à inovação e evolução da mesma. Se a organização for de grandes dimensões, necessita de alguns anos para detalhar e beneficiar das integrações efetuadas (Alonso, 2003).

Um modelo de maturidade é usado para comparar os processos de uma organização, focada em avaliar e mapear o nível do estado de integração dos processos e com base nessa avaliação propor estratégias, a curto ou longo prazo, que deveriam ser implementadas para não colocar em risco a evolução da própria organização. (Röglinger, 2011)

Por diversas razões, os modelos de maturidade são genéricos e o método de avaliação é baseado em *AQ*, o critério da resposta está dependente da subjetividade de um consultor ou de quem vai avaliar essa arquitetura empresarial, tornando-se por isso ambígua a sua avaliação.

O facto destes métodos não terem em conta o ambiente de cada organização é também um problema (Bieberstein, 2005), pois as soluções apresentadas partem de muitos pressupostos.

Para responder a este problema, foi necessário testar alguns modelos de maturidade e optou-se pela aplicação do modelo OSIMM da OpenGroup, como "objeto" de estudo, sendo o nosso foco as *AQ*. Verificou-se que muitas das respostas não possuíam a mínima precisão, logo não obteríamos um nível de maturidade justo.

Devido a isto, definimos o nosso problema de investigação como: **como avaliar com maior precisão o nível de maturidade de uma organização que adoptou SOA?**

Tendo em conta o problema em análise, propomos uma metodologia que ajude a avaliar mais ponderadamente o nível de maturidade de uma organização que adotou uma arquitetura orientada a serviços (SOA). Primeiro de tudo precisamos de definir 7 critérios de resposta a cada pergunta das 89 perguntas do modelo OSIMM. A definição deste conjunto de critérios de avaliação, será feita com base numa revisão exaustiva da literatura científica existente e nas boas práticas de avaliação de um modelo de maturidade para uma arquitetura orientada a serviços. A escala foi baseada em sete tipos de resposta para uma determinada *AQ*, que permite avaliar em que nível de maturidade está cada serviço numa certa dimensão, como explicado no Capítulo 5 -Proposta.

A validação dos critérios será feita através de entrevistas dirigidas a responsáveis do IT de organizações, com base em duas métricas *"Is usefull for SOA",* ou seja, em que medida os critérios criados por nós são úteis para avaliar o SOA, e *"Is easy to answer",* que nos permite saber se os critérios que definimos são demasiado complexos para responder com uma avaliação de 0 a 5, utilizando o método *likert scale*.

Após a validação deste conjunto de critérios aplicamos o nosso método de avaliação **mais eficaz no cálculo do nível de maturidade de uma organização que adoptou SOA.**

Para demonstrar o uso da nossa solução, aplicamos o método a várias organizações portuguesas, de forma a que o valor de maturidade seja mais preciso com base nas escolhas dos critérios criados por nós para as perguntas do modelo OSIMM. Em termos de caso de estudo é interessante pois vai permitir averiguar quais são as dimensões em que cada organização está mais evoluída e definir o *roadmap* como estratégia de evolução com os critérios que possuem um nível de maturidade mais baixo.

Para avaliar a nossa proposta e os seus resultados, usámos a *Moody and Shanks Framework* (Moody & Shanks, 2003) para avaliar os modelos produzidos; e ainda os quatro princípios propostos por Österle et al. (Osterle, et al., 2011) para validar se a nossa proposta cumpre os objetivos propostos. Depois da avaliação, concluímos que o método proposto é adequado para avaliar o nível de maturidade de uma organização que adotou SOA.

1.1. Método de Investigação

Esta investigação foi baseada na Design Science Research Methodology (DSRM). A fim de compreender melhor o DSRM, é importante perceber o seu conceito, definido por Hevner et al. – *methodology is a system of principles, practices and procedures applied to a given branch of knowledge* (Alan, Salvatore, Jinsoo, & Sudha, 2004). Esta metodologia, no contexto dos SI, exige a criação e avaliação de um artefacto inovador e propositado que resolva um problema específico num domínio específico (Alan, Salvatore, Jinsoo, & Sudha, 2004). Este artefacto pode ser definido como: construções (vocabulário e símbolos), modelos (abstrações e representações), métodos (algoritmos e procedimentos) e instanciações (implementações e protótipos de sistemas) (Alan, Salvatore, Jinsoo, & Sudha, 2004). Nesta investigação, propomos como artefacto, um **método**, i.e, o artefacto irá explicar e dar orientações sobre como resolver o problema encontrado. DSRM consiste num processo iterativo composto por seis etapas, (Figura 1), e inclui métodos rigorosos para a criação e avaliação dos artefactos propostos. De seguida será explicada cada etapa do DSRM.

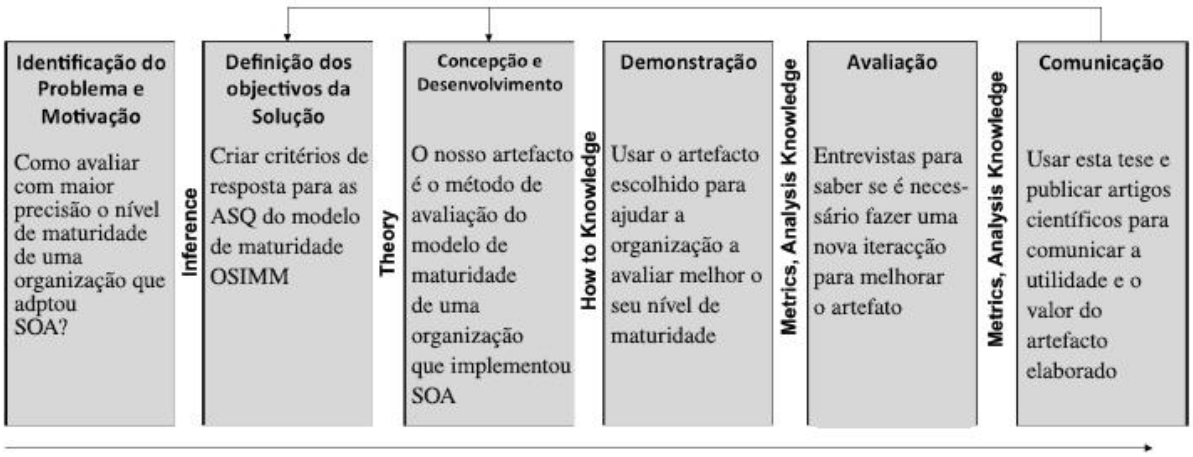


Figura 1 - O processo DSRM (adaptado de **(Peffers, Tuunanen, & Rothenberge, A Design Science Research Methodology for Information Systems Research, 2008)**)

As etapas da metodologia referenciada anteriormente são:

1. **Identificação do Problema e Motivação:** Definir o problema específico da investigação e justificar o valor da solução. Nesta etapa é necessário conhecer o estado do problema e justificar de uma forma clara a importância de uma solução.
2. **Definição dos objetivos da solução:** Inferir os objetivos de uma solução através da definição do problema, do trabalho relacionado e do conhecimento, tendo em conta o que é possível e viável fazer. Os objetivos podem ser quantitativos, tais como identificar em que termos é que a solução a alcançar será melhor que as já existentes, ou qualitativos, com uma descrição de como é esperado que o novo artefacto resolva os problemas.
3. **Conceção e desenvolvimento:** Criar o artefacto, determinando quais as funcionalidades desejadas e a sua arquitetura e finalmente criar realmente o artefacto. O artefacto pode ser qualquer objeto concebido que tenha incorporado na sua concepção a contribuição da investigação.

4. **Demonstração:** Demonstrar o uso do artefacto para resolver uma ou mais instâncias do problema. Isto pode ser feito usando experiências, simulações, provas ou outra atividade apropriada.
5. **Avaliação:** Esta etapa compara os objetivos de uma solução com os resultados observados atualmente obtidos através da demonstração do artefacto. Métodos de avaliação bem executados são utilizados para demonstrar a utilidade, qualidade e eficácia do artefacto concebido. Tais métodos podem incluir simulações, entrevistas com profissionais, e outros. Depois desta etapa, os investigadores podem decidir se é necessário fazer uma iteração regressiva, a fim de melhorar o artefacto.
6. **Comunicação:** Consiste na comunicação da investigação como um todo (problema, objetivos, artefacto, demonstração e avaliação) para o público relevante. A submissão e apresentação de artigos podem fazer cumprir esta etapa.

Para terminar esta explicação é importante referir que, para usar a metodologia DSRM, os investigadores não têm de iniciar na primeira etapa e seguir a metodologia sequencialmente até à última etapa, pois, como podemos ver na Figura 1, existem vários pontos de partida possíveis.

1.2. Estrutura da Dissertação

Este documento está dividido em 10 capítulos, descritos de seguida:

1. **Introdução (Capítulo 1)** introduz o tema, dando um contexto geral acerca da dissertação, descreve o método de investigação utilizado e a sua aplicação à dissertação e também a estrutura do documento.
2. **Problema (Capítulo 2)** descreve em detalhe o problema da investigação e a motivação.
3. **Trabalho Relacionado (Capítulo 3)** apresenta uma visão global da literatura na área de investigação e descreve os conceitos necessários que suportam a proposta, que são cruciais para a coerência da nossa proposta.
4. **Objetivos da Solução (Capítulo 4)** define os objetivos que a nossa solução pretende alcançar.
5. **Proposta (Capítulo 5)** identifica os objetivos da solução e explica o artefacto produzido.
6. **Demonstração (Capítulo 6)** apresenta a demonstração do artefacto nas organizações e as conclusões sobre os resultados.
7. **Avaliação (Capítulo 7)** apresenta uma explicação da estratégia de avaliação usada para avaliar o artefacto, analisar o artefacto, avaliar os resultados e a discussão dos resultados.
8. **Comunicação (Capítulo 8)** descreve a forma como vamos obter a aprovação da nossa investigação por parte da comunidade científica.
9. **Conclusão (Capítulo 9)** apresenta um resumo das conclusões principais, limitações do trabalho desenvolvido, as contribuições desta dissertação e algumas sugestões de trabalho futuro.
10. **Anexos** apresentam:

A. Bibliografia para criação dos critérios das AQ do modelo OSIMM

B. Critérios de resposta para as AQ do modelo OSIMM

C. Questionário de Avaliação da Proposta Tese

2. Problema

Neste capítulo vamos proceder à etapa de **identificação do problema e motivação** do DSRM. Esta etapa serve para definir o problema específico da investigação que está relacionado, como **melhorar** a avaliação do estado de maturidade de uma empresa que adota uma arquitetura orientada a serviços, com base no modelo OSIMM.

2.1. Motivação

As razões que nos conduziram à realização desta dissertação prendem-se com a constatação de que, hoje em dia, nas organizações privadas ou públicas, se gastarem milhares ou até mesmo milhões de euros na compra de software e depois falta a funcionalidade mais importante que é a integração. Sendo essa integração mal executada, surgem problemas nas organizações, como perdas de dados informacionais, mal parametrizados ou simplesmente redundância de dados.

Foi elaborado um estudo exaustivo de como um SOA deverá ser estruturado, Capítulo 3 -Trabalho Relacionado e como poderá ser avaliado a nível de serviços para determinadas dimensões. Os modelos de maturidade surgiram nessa necessidade tendo já sido criados alguns, para medir o progresso da adoção do IT ao longo do tempo.

Com a introdução do SOA como tecnologia, diferentes modelos de maturidade foram propostos e foram desenvolvidos, essencialmente, por empresas de desenvolvimento de software e empresas de consultoria. Como consequência, esses modelos tiveram a função de ser mais uma ferramenta de venda para o cliente, em vez de um instrumento para identificar as necessidades dos clientes.

2.2. Problema da investigação

Após a apresentação das principais motivações inerentes ao trabalho de investigação, é de vital importância a definição do problema, devido à existência de um espectro alargado de problemas associados às principais temáticas em análise: AE, avaliação de modelos de maturidade, SOA.

Deste modo, definimos como principal preocupação o facto de os modelos de maturidade basearem a sua avaliação em *AQ* genéricas (Becker & Knackstedt , 2009) (Irani & O'Keefe, 2001).

Estas preocupações, estão a ganhar cada vez mais importância, devido ao facto de as organizações quererem tirar maior rentabilidade dos SI que adquiriram e manter os seus dados coerentes. Por isso quando a organização pretende adquirir um novo SI, existe a necessidade de saber qual o *return on investment* (ROI) que aquele SI vai suportar e como vai ser integrado com os SI já existentes (The Enterprise IT Performance Maturity Model, 2014). Com base nos resultados das *AQ*, sabemos qual a

estratégia, o *roadmap* que queremos para a nossa organização e quais os benefícios da mesma implementação, se necessário. Uma das preocupações é saber quem avaliará e como será avaliada a necessidade da existência do SI em questão (Becker & Knackstedt , 2009).

Para avaliar estes aspetos foram, ao longo do tempo, sendo propostos vários modelos como HP SOA Maturity Model, Oracle SOA Maturity Model, OSIMM, SOAMM-Microsoft, entre outros (Pai) (Walker).

Definimos o problema da nossa investigação com uma pergunta: **Como avaliar com maior precisão o nível de maturidade de uma organização que adotou SOA?**

A principal motivação para resolver este problema está relacionada com o facto de que, cada vez mais, as organizações de média ou grande dimensão terem conhecimento da tecnologia que possuem e como devem tirar benefícios dos SI que possuem, de forma a economizar recursos e, ao mesmo tempo, estarem atualizadas tecnologicamente. Existe portanto a necessidade de uma metodologia de avaliação dos seus serviços, simples e adaptada à realidade. Portanto, a nossa proposta trará benefícios concretos para a organização, para que estes possam fazer escolhas fundamentadas e adaptadas com os resultados do nosso método de avaliação.

3. Trabalho Relacionado

Neste capítulo apresentamos uma revisão da literatura dos tópicos relacionados com a nossa investigação. Na secção 3.1 referimos qual a tecnologia de suporte a uma arquitetura orientada a serviços, como os *web services* e os *standards* de suporte também vão ser abordados. Na secção 3.2 vamos discutir o que é um *middleware* e quais são as suas principais funções. Além disso vão ser explicados, na secção 3.3, os conceitos básicos de uma arquitetura orientada a serviços SOA, quais os problemas associados a aplicações distribuídas nas organizações e os princípios que devem ser seguidos para possuir um *true SOA.* Nas secções 3.4 e 3.5 explicamos o uso de modelos de maturidade, comparando os modelos e os respetivos métodos de avaliação, nomeadamente para SOA.

3.1. Web Services

Como referido anteriormente, os *web services* são a tecnologia de suporte do SOA. A ideia é oferecer serviços e tem como base um Software-As-Service (SaaS) (Alonso, 2003). A ideia do SaaS, é que o cliente paga para aceder a um serviço pagando uma *subscription* diferenciada conforme o serviço que pretende. A vantagem para as organizações é que o cliente só tem acesso à camada lógica de apresentação, sem ter de instalar nada nos seus servidores, recorrendo apenas à web. Os *web services* são parte integrante de um SaaS, não são usados diretamente pelo *end user* do sistema, mas estão presentes na *application logic layer*. Um *web service* é visto como uma aplicação acedida por outras aplicações através da web, deve ser independente, *loosely coupled*, *self-contained, programmable web-enabled application that can be described, published, discovered, coordinated, and configured using XML artifacts for the purpose of developing distributed interoperable applications* utilizando determinados standards (Alonso, 2003).

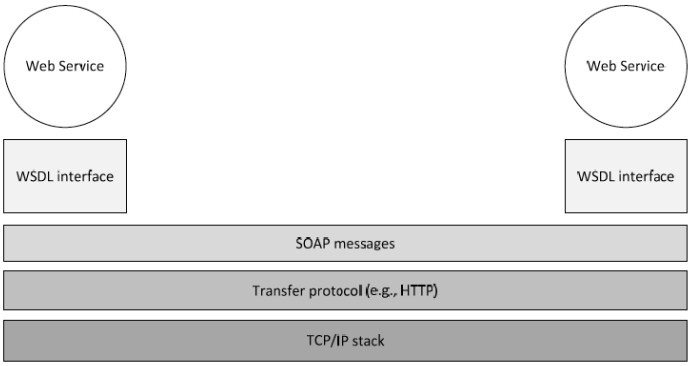


Figura 2- SOAP, WSDL, UDDI, BPEL

Os standards de referência são SOAP, WSDL e UDDI, fundamentais para a interação entre os *web*

services. Atualmente existem muitos *standards*, baseados nos *core standards* como *security, transactions/coordination, reliability, policy*.

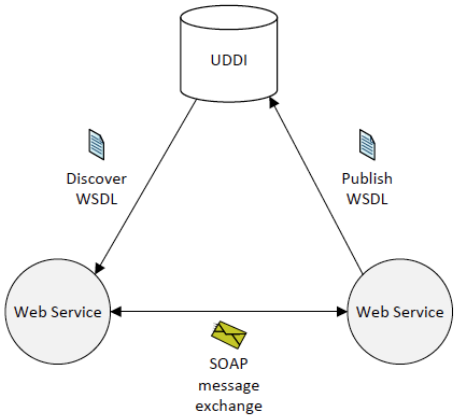


Figura 3– Implementação dos *application services* pelo core dos *WS standards* **(Erl, 2005)**

O SOAP é um dos *standards* usados para a troca de informação entre *web services*. (W3C, 2007) SOAP é um *wire protocol*, que define a comunicação entre uma ou mais aplicações. O *wire protocol* é usado para facilitar o transporte entre sistemas usando a *network*. O HTTP é o protocolo de transporte mais utilizado para as mensagens SOAP porque não é bloqueado pelas *firewalls*.

WSDL

O *Web Service Description Language* (WSDL) é um standard que define como descrever um *Web Service* e como os dados podem ser trocados na interação dos serviços. A especificação de um WSDL é baseada num esquema suportado pela linguagem XML, que descreve o que o serviço deve fazer, é também onde reside o que deve ser invocado e como. As especificações do WSDL podem ser *published by service providers to a service registry to be discovered by service requestor* (Pai).

UDDI

O *Universal Description, Discovery and Integration* (UDDI) é uma *cross-plataform*, um *standard* suportado por XML que serve como registo para os Web Services (Alonso, 2003). O UDDI *registry* é usado como um repositório em que os *service providers* publicam as especificações do WSDL e são acedidos por *service requestors to discover web services*. O objetivo inicial do UDDI era suportar todos os registos a nível global onde as organizações poderiam publicar e procurar *service descriptions*. Com a versão 3, o *standard* do UDDI passou a incluir suporte de registo para organizações privadas e também semipúblicos, em que o objetivo era partilharem entre organizações os serviços que desejariam.

3.2. Middleware

O *middleware* é um software que foi desenvolvido com o fim de facilitar a gestão das interações entre *heterogeneous computing platforms* (Irani & O'Keefe, 2001). O *middleware* funciona como uma ponte entre as aplicações, com a finalidade de comunicar e transferir dados, sem ser necessário nenhuma

mudança substancial, atua como um tradutor. Essa comunicação que é efectuada pelo tradutor é baseada em trocas de mensagens por meio de XML, SOAP e *web services* (W3C, 2007). Através da troca de mensagens é possível trocar dados, invocar ações ou notificar outra aplicação de eventos que estejam a ocorrer.

A figura 4 permite visualizar uma situação em que diferentes aplicações SILO'S comunicam usando um *middleware*. Muitas destas situações são visualizadas nos dias de hoje nas médias ou grandes organizações, em que cada departamento tem as suas próprias aplicações e o seus próprios *data system* para gerir. Neste caso, o *middleware* atua como um tradutor, como referido anteriormente.

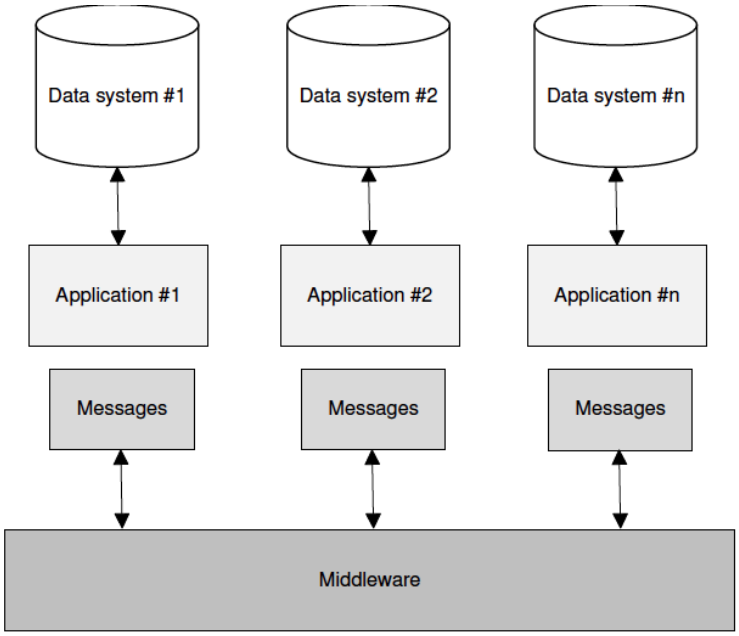


Figura 4- Comunicação de aplicações em SILO'S através de troca de mensagens com o uso de um middleware **(Papazoglou, 2007)**

Um dos fatores mais importantes para que a integração das aplicações seja feita com sucesso com o uso de um *middleware* é que, a mensagem a ser transmitida entre o *middleware* e as aplicações esteja no formato desejado.

Existem dois estilos de comunicação por mensagens: síncronas e assíncronas.

Quando duas aplicações comunicam sincronamente é porque a aplicação que envia o pedido fica à espera de uma resposta por parte do recetor, ou seja, a aplicação que envia é *locked* e enquanto não recebe a resposta não inicia mais nenhuma atividade. Quando a comunicação entre aplicações é assíncrona, por regra, é usada uma *message queue*. Quando as aplicações usam uma *message queue* para comunicar, a aplicação que envia coloca a mensagem numa *queue*, que mais tarde irá ser recebida por uma ou mais aplicações. A vantagem em relação a uma comunicação síncrona reside no facto de, a aplicação que envia não fica *locked* e pode executar outras tarefas em paralelo. O middleware obviamente é baseado em comunicações assíncronas que usam *message queue*,

referidas como *Message-oriented middleware* MOM (W3C, 2007). O MOM permite não só a comunicação entre duas aplicações, mas entre *one-to-many* e também *many-to-many*, sendo muito usado para aplicações distribuídas.

3.3. Service Oriented Architecture

Para percebermos o conceito de *service-oriented architecture* (SOA), precisamos de perceber a arquitetura e qual o problema que o SOA tentou resolver. SOA foi desenvolvido aplicando o conceito de *software architecture*, para resolver o problema de como relacionar sistemas distintos, de maneira que pudessem funcionar como um todo, sistemicamente. (Marks & Bell, Service Oriented Architecture (SOA): A Planning and Implementation Guide for Business and Technology, 2006)

Cada aplicação no SOA é vista como uma função especial ou um elemento para um aplicação maior. Essa função especial é chamada serviço. Esses serviços devem respeitar os princípios do SOA como *reusability, interoperability and integration across all business processes and technology plataforms*. (Tipton & Nozak, 2012) Os serviços são baseados na linguagem XML e os processos de negócio que deles derivam usam o *Business Process Markup Language* (BPML), que é uma extensão do XML. O *Enterprise Service Bus* (ESB), é uma arquitetura usada num ambiente de computação distribuída, flexibilizando as várias comunicações e os protocolos usados.

3.3.1. Problemas no SOA

Nas organizações tradicionais, a estrutura funcional é organizada por departamentos, vendas, recursos humanos, etc. Os próprios departamentos foram adquirindo as suas próprias aplicações e customizando-as conforme os requisitos desse departamento (Britton & Bye, 2004). O problema de uma organização ser em SILO'S, é que cada departamento faz a gestão das suas próprias aplicações, existindo diferentes aplicações na organização, o que dificulta a tarefa de integrar esses SILO'S (Mahajan, 2006). Para uma organização, a visão holística não é desejada, a existência de SILO'S prolonga-se por razões diferentes. Umas das razões relaciona-se com a **autonomia**, pois os departamentos não querem perder o poder na gestão dos projetos independentes que tem de controlar. Outra das razões e a principal é o **receio**, dos *stakeholders* da organização temem a mudança para uma grande aplicação integrada em relação às existentes. No entanto, é necessário a interação entre os SILO'S com um suporte automatizado, para que os processos de negócio abranja várias funções de negócio.

As arquiteturas IT são usadas para dar suporte à estrutura, integrar aplicações e fazer a gestão do problema dos SILO'S. A arquitetura a aplicar difere dos diferentes níveis de abstração que são precisos (Group, Service Integration Maturity Model (OSIMM), 2011).

Existem fundamentalmente quatro sub-arquiteturas numa AE:

- *Bussiness Architechure* – é uma arquitetura que descreve a estratégia de negócio, *governance*, organização e os processos de negócio;

- *Applications architechure* – é uma arquitetura que fornece um modelo como as aplicações devem ser implementadas, as suas interações e as relações com os principais processos de negócio da organização;
- *Data Architechure* – descreve a estrutura de dados tanto física como lógica da organização e como é feita a sua gestão;
- *Tecnhology Architechure* – descreve os recursos de hardware que são necessários para apoiar a implementação dos processos negócios, dados e *application services*.

Nos princípios da automação, os processos de negócio eram suportados por um *mainframe* onde a lógica da aplicação e a gestão de dados, faziam parte do mesmo sistema (Erl, 2005).

Uma típica arquitetura nos dias de hoje, envolve a interação de muitas aplicações distribuídas, que interagem sobre a *network* e com vários standards distintos (Lam, 2005).

3.3.2. Os princípios de SOA

O Service-Oriented Architecture (SOA) é uma solução para ultrapassar as limitações do tradicional *middleware*. A integração não deve ser apenas interna, mas também com aplicações externas. No entanto é necessário ter implementado na organização determinadas tecnologias ou *standards web* que permitem a comunicação entre essas aplicações, através de TCP/IP e HTTP.

Os princípios do SOA são:

- *Reusibilty* - A lógica é dividida em serviços com objetivo de maximizar a sua reutilização
- *Service Description* - os serviços estão associados a um *services-description*, padrões como WSDL.
- *Lossely Coupling* - Os serviços minimizam a dependência entre eles para realizar a sua tarefa.
- *Abstraction* - É omitida a lógica do serviço, a informação é publicada num *service contract.*
- *Composability* - São criados serviços que repartem problemas maiores em sub-problemas.
- *Autonomy* - Os serviços devem ter controlo sobre a própria lógica que encapsulam, autonomia.
- *Statelessness* - Idealmente o serviços devem ser *stateless*.
- *Discoverability* - A possibilidade de os serviços serem encontrados (normalmente por meio de um *registry*).

3.4. Modelos de Maturidade

Nesta secção, vamos explicar em que consiste um modelo de maturidade, qual a sua finalidade e quais os modelos de maturidade adoptados a SOA.

O modelo de maturidade é uma estrutura que é utilizada como referência para a comparação e avaliação de determinados aspetos da organização, como comunicação, serviços, processos,

satisfação, segurança e objetivos estratégicos. Um modelo de maturidade é usado especificamente quando se pretende avaliar a capacidade de implementar estratégias para a organização com a finalidade da evolução ou para controlar o risco da mesma, a partir das ditas estratégias.

Depois de elaborada uma auditoria à organização, o modelo de maturidade será usado para mapear o nível em que a organização se encontra, em termos dos processos e os procedimentos que deve ter em conta no futuro.

O modelo de maturidade deve (Mahajan, 2006):

- Fornecer uma linguagem comum e com a mesma visão;
- Ser capaz de priorizar ações ;
- Fornecer uma maneira de definir o que está implementado (as-IS) e qual o plano para melhorar os processos (to-BE);
- Benefício de possuir documentação de *data management experience* ;

Uma organização possui um bom nível de maturidade, se teve um bom *benchmark* e possuir um risco menor (Inaganti & Aravamudan , 2007).

3.5. Modelos de Maturidade SOA

Sendo o nosso foco de objeto de estudo os modelos de maturidade para uma arquitetura orientada a serviços, nesta secção, vamos analisar a sua estrutura e o seu método de avaliação. Os modelos estudados são: Oracle Maturity Model, OSIMM V2, SOAMM Microsoft, CBDI. Qualquer um destes modelos difere nos estados de maturidade, as dimensões em que atuam e o método de avaliação.

Na Tabela 1 estão listados 4 modelos de maturidade que são distintos em inúmeros aspetos. O questionário do SOAMM-Microsoft e do CBDI não está disponível publicamente.

Modelos de Maturidade	**Autor**	**Níveis de Maturidade**	**Dimensões**	**Ferramenta de avaliação**
Oracle SOA Maturity Model 2006	Oracle	5	8	Questionário
OSIMM V2 2009	The Open Group	7	7	Entrevistas/ Questionário
SOAMM – Microsoft 2007	Microsoft	4	4	Entrevistas (n/a)
CBDI 2007	CBDI	5	7	Entrevistas/ Questionário (n/a)

Tabela 1 –Modelos de maturidade para organizações que adoptam SOA

3.5.1. Oracle Maturity Model

A Oracle tem estado a trabalhar, nos últimos 8 anos, com variadíssimas organizações com diferentes níveis de adoção de SOA. Estas experiências foram efectuadas utilizando o Oracle SOA-MM. O objetivo foi medir o progresso desde a sua implementação do modelo. Mas mais importante, é poder identificar determinadas *capabilities,* que estão a faltar ou em atraso, que impedem a sua evolução.

O Oracle SOA-MM define os conceitos chaves como: *capabilities/domains, maturity, and adoption*. (Hensle & Deb, 2013).

O SOA-MM inclui oito domínios, capturados pelas melhores práticas que a Oracle colecionou ao longo de alguns anos de trabalho, em variadíssimas organizações. Estes níveis de *capabilities*, permitem medir e orientar o progresso para a adopção do SOA (Hensle & Deb, 2013).

Os domínios são: *(Business & Strategy, Architecture, Infrastructure, Information, Projects and Portfolios & Services, Operations and Administration & Management, Organization, Governance).*

Como na indústria de software, quando se refere maturidade, é diretamente associada ao CMM - *Capability Maturity Model* e ao seu sucessor CMMI - *Capability Maturity Model Integration* a Oracle optou por definir esses mesmos níveis para adotar ao seu modelo que são: *Optimized, Managed, Systematic, Opportunistic, Ad Hoc, No SOA.*

Método de avaliação:

O método de avaliação de uma organização utilizando o Oracle SOA-MM, requere a recolha de toda a informação que seja necessária, para a avaliação e é completada com um questionário dirigido a elementos da organização como *executives, enterprise architects, developers, project and program management, operations, etc.*

O consultor, utiliza detalhadamente o *SOA Maturity Model* para corresponder aos níveis de maturidade adoptados para cada nível em cada dimensão.

Esses resultados podem ser representados de variadíssimas formas, dependendo da organização alvo e da quantidade de informação recolhida.

3.5.2. OSIMM V2

O Service Integration Maturity Model (SIMM) foi inicialmente publicado pela IBM em 2006 (Arsanjani & Holley, 2005). Foi adotado pelo Open Group que fez pequenas alterações e então criou o Open Group Service Integration Maturity Model (OSIMM) (Group, The Open Group Architecture Framework (TOGAF), 2007). Em 2009, o Open Group decidiu criar o OSIMM V2, em que fez pequenas alterações no método de avaliação de maturidade e introduziu novas funcionalidades. O objetivo do OSIMM é avaliar a maturidade da integração dos *application services*. O OSIMM tem sete estados de maturidade dos serviços, representados no eixo horizontal e atua em sete dimensões representados no eixo vertical.

No eixo horizontal são representados os estados da *enterprise business* e da maturidade da integração dos serviços IT. Cada um destes sete níveis reflete o estado abstrato de cada organização em termos da maturidade da integração dos seus serviços de negócio e/ou IT.

No eixo vertical são representadas as sete dimensões, *The Business, Organization and Methods* que são focados em aspetos organizacionais. As *Applications, Architecture, Information and Infrastructure* são focados em aspetos mais técnicos.

Deveremos dar atenção aos três primeiros *layers* do modelo de maturidade OSIMM - *Silo, Integrated and Componentized* – são referidos como um *Service Foundation Levels*. São considerados os pré-requisitos para os serviços relacionados com aspetos organizacionais.

O OSIMM define o conjunto de dimensões representado por diferentes vistas como:

- ***Business*** - A dimensão de *business* está focada na arquitetura de negócio, ou seja, as práticas e as políticas de negócio atuais da organização, bem como os processos são desenhados, estruturados, implementados e executados (Group, Service Integration Maturity Model (OSIMM), 2011)
- ***Organization and Governance*** - A dimensão de *Organization and Governance* é focada na estrutura da própria organização, em medir a eficácia no contexto SOA e SOA Governance. (Group, Service Integration Maturity Model (OSIMM), 2011)
- ***Method*** - A dimensão do *method* é focada nos métodos e processos utilizados pela organização na área de IT, de negócios e em todo o *Software Development Lifecycle*, tais como o *management, estimation techniques, project management, quality assurance processes, design methodologies and techniques, and tools for designing solutions* (Group, Service Integration Maturity Model (OSIMM), 2011).
- ***Application*** - A dimensão da *application* é focada na estrutura da aplicação, na sua *functional decomposition, re-usability, flexibility, reliability, and extensibility of the applications.* (Group, Service Integration Maturity Model (OSIMM), 2011)
- ***Architecture*** - A dimensão da *architecture* é focada na estrutura da arquitetura, como a sua *topology, integration techniques, enterprise architecture decisions, standards, policies, web services adoption level, experience in SOA implementation, SOA compliance criteria, and typical artifacts produced* (Group, Service Integration Maturity Model (OSIMM), 2011).
- ***Information*** - Esta dimensão é focada no modo como a informação é estruturada, o método de acesso aos dados da organização a *data characteristics, data transformation capabilities, service and process definitions, handling of identifiers, security credentials, knowledge management, business information model, and content management.* (Group, Service Integration Maturity Model (OSIMM), 2011)
- ***Infrastructure and Management*** - A dimensão de *infrastructure and management* é focada na capacidade da infraestrutura da organização, service management, IT operations, IT management and IT administration, como os SLAs são comprometidos, como a monitorização é

realizada e que tipo de plataformas de integração são disponibilizados. (Group, Service Integration Maturity Model (OSIMM), 2011)

Os sete níveis de maturidade SOA são: *Silo, Integrated, Componentized, Service, Composite Services, Virtualized Services, Dynamically Re-Configurable Services*.

O *heart of OSIMM* são os sete níveis de maturidade dos *enterprise business and IT service-integration*. Cada um dos sete níveis reflete um possível estado abstrato em termos de maturidade na integração dos seus serviços e a solução para o SOA.

Método de avaliação do OSIMM

O OSIMM pode ser usado para avaliar o SOA de um único projeto, ou seja, apenas de uma dimensão ou para a organização na totalidade. O objetivo do método de avaliação do OSIMM é avaliar a maturidade atual e determinar o nível de maturidade que se pretende atingir, que vai ao encontro dos objetivos pretendidos (Group, The Open Group Architecture Framework (TOGAF), 2007). O método de avaliação do OSIMM é iterativo e evolutivo. Quando uma organização adota por uma estratégia SOA, torna-se familiarizada, com o tipo de avaliações OSIMM e pode adicionar os seus próprios indicadores de maturidade ao modelo.

A análise consiste nas três seguintes etapas:

1. Avaliar o estado de maturidade atual da empresa, organização e de IT.
2. Identificar o objetivo pretendido pela organização e a construção de uma visão em relação ao *business, processes, staff and IT Solutions.*
3. A produção de um relatório de recomendação, que identifica os atuais níveis de maturidade nos diversos domínios e descreve as ações para obter o estado pretendido com um *roadmap*.

3.5.3. SOAMM Microsoft

O SOA Maturity Model (SOAMM) é usado pela Microsoft, para avaliar o SOA de um cliente e fornecer um *roadmap,* para identificar prioridades e objetivos para atingir uma maior agilidade no negócio comprometido pelo SOA. O SOAMM consiste em 36 tipos de tecnologia-independentes e serve como um guia em que fornece informação sobre quais as tecnologias que tem de possuir, em cada um dos níveis na sua organização, para criar um valor aproximado a uma arquitetura orientada a serviços. O modelo foi desenvolvido com base nas melhores práticas de um conjunto de equipas da Microsoft e alguns clientes. O SOAMM é baseado no CMM, mas possuindo apenas 4 níveis: *Basic, Standardized, Advanced, Dynamic.*

As dimensões são: *Administration, Consumption, Implementation*

Método de avaliação:

O método de avaliação deste modelo é elaborado pela equipa dos Microsoft Services, que defende uma abordagem pragmática em que os seus esforços SOA são movidos pela visão estratégica e as

necessidades de negócio, sendo desenvolvidos incrementalmente. A abordagem que utiliza baseia-se em 3 componentes principais:

- A transformação da *business architecture* num mapa colorido, designado por *heat map*, que corresponde à capacidade de cada processo;
- Um report resultante da avaliação de maturidade SOA baseado em mais de 300 perguntas, com base nos 36 tipos de tecnologia-independentes fundamentais para aumentar a probabilidade de sucesso e reduzir o risco no desenvolvimento dos projetos SOA;
- Um *roadmap* resultante da avaliação dos serviços e catalogar os serviços existentes no ambiente SOA.

O método de avaliação será de amplo alcance nas mais diversas áreas e focado em identificar os pontos fortes e áreas relacionadas com o SOA que necessitam de ser melhoradas. O *roadmap* irá fornecer informação extra para priorizar ações e atingir os objetivos SOA pretendidos.

3.5.4. CBDI

O modelo de maturidade CBDI SOA foi publicado em 2005. Foi posteriormente atualizado em 2006 e novamente em 2007. Este modelo avalia a maturidade de adoção do SOA em toda a empresa, por isso considera a evolução do SOA de toda a empresa.

Na fase mais madura, denominada *ecosystem*, as organizações podem partilhar entre si os seus *application services*, de suporte aos processos de negócio.

Os estados de maturidade do modelo CBDI são: *Early learning, Applied, Integrated, Enterprise, Ecosystem.* Os estados de maturidade refletem a evolução do SOA e dos serviços de toda a organização. Começa no estado 1 como um projeto piloto, ou só a estrutura pensada, enquanto que no estado 5 os *application services* já estão dissociados ao máximo do processo de negócio e podem ser usados entre SILO'S na organização.

As diferentes dimensões da maturidade refletem o lado mais técnico e organizacional da adopção do SOA.

As dimensões são: *Management, Service architecture, Operational infrastructure, Life cycle infrastructure, Framework and process, Organization, Project & programs.*

Estes aspetos abrangem uma ampla perspetiva sobre a adoção de SOA. Neste modelo é dado menos ênfase à orquestração dos processos de negócio com base em *application services* (Spotter, 2007).

Método de avaliação:

O método de avaliação do CBDI usa o CMMI, avaliando a capacidade da intersecção entre os níveis de maturidade, como está exemplificado na tabela 2. A sua dimensão é a *ability and capacity to perform a specific function, process or task* (Spotter, 2007). Depois de discutidos os elementos para um plano de capacidade, é necessário preenchermos o resto do *roadmap*. Os níveis de capacidade

são avaliados de 1 a 3, baseando-se nos indicadores de capacidade.

O questionário possuí um total de 207 questões, onde é referido que grande parte das organizações ainda estão nos primeiros níveis de avaliação, não possuindo ainda a partilha de serviços. Uma organização que esteja no nível *integrated,* reflete que possui um ESB ou equivalente (Spotter, 2007).

Capability	**Maturity Model**
Record of Services in Use	Applied
Monitoring of Service Usage	Applied
Control over service Usage	Integrated
Policy based control over Service Planning and Provisioning	Enterprise

Tabela 2- Exemplo de Capability /Maturity

3.5.5. Comparação ente modelos

Antes de discutirmos as diferenças entre os modelos, primeiro identificamos as suas estruturas. Os aspetos em que nos focamos, para explicarmos os modelos foram: o nome dos estados de maturidade, as dimensões em que atuam e qual o método de avaliação. Os métodos de avaliação podem ser elaborados com suporte em entrevistas, com o apoio de questionários, ou com dados recolhidos na organização. A principal diferença entre os modelos, é que uns baseiam-se em dados empíricos como o Oracle SOA-MM e o SOAMM-Microsoft, enquanto que os restantes SOA-MM são elaborados com base em *expert interviews*. Isto porque o SOA ainda é uma tecnologia recente e as implementações efectuadas ainda são escassas (Legner & Heutschi, 2007).

Os modelos de maturidade para SOA focam-se principalmente no progresso da adopção do SOA. Além disso, o nível de maturidade difere entre ambos os modelos. O SOAMM da Microsoft e o CBDI SOA MM focam-se essencialmente no âmbito *business services* e *application services* na organização. O âmbito começa a crescer, o quão bem e quantos mais serviços são necessários de integrar em toda organização ou entre organizações.

O Oracle MM e o OSIMM focam-se na maturidade da integração entre *application services* no contexto de organização, mas não dependendo do seu âmbito. As organizações não têm progredido assim tanto na adoção do SOA, por isso a maioria delas encontra-se provavelmente num nível baixo de maturidade 1 (Becker & Knackstedt , 2009) (Pai).

Ambos os modelos OSIMM e CBDI têm os mesmos indicadores e dimensões que abrangem, tanto a nível técnico, como a nível organizacional, para a adopção do SOA, embora no OSIMM, o estado inicial ou seja, os serviços pertencentes ao *foundation level* ainda não tem qualquer tipo de aplicação do SOA, apenas pré-requisitos para atingir o estado inicial do CBDI.

A nível organizacional, pretende-se perceber a estrutura da organização e quais os aspetos relevantes.

Os níveis de maturidade do modelo OSIMM baseiam-se nos serviços. Numa situação com um bom nível de maturidade, os serviços estão virtualizados e podem facilmente ser usados em diferentes processos de negócio.

Com o nível cinco do OSIMM, os processos de negócio podem ser integrados com outros serviços de nível mais baixo, *loosely coupled* em *design time*, o que é fundamental para a organização suportar a reutilização de serviços partilhados a nível interno ou entre organizações.

Os estados do CBDI focam-se explicitamente no âmbito da utilização dos serviços na organização. Os serviços são utilizados, pela primeira vez, como um *project pilot* e depois são centralizados e partilhados entre os SILO'S da organização.

3.6. Conclusão

Adotar uma arquitetura orientada a serviços numa organização, envolve a tradução dos princípios de SOA em arquiteturas concretas e implementações que se integrem com as aplicações existentes. Esta tradução exige que as escolhas sejam feitas sobre determinados *standards*, tecnologias e softwares para implementar um *true SOA*.

No entanto, adotar SOA não é apenas uma implementação técnica. A importância de aspetos organizacionais, o conhecimento e interesse pelos *stakeholder's* são fundamentais para que a implementação inicial seja efectuada com sucesso. (Pai) As organizações devem adotar gradualmente o SOA, para minimizar o risco de fracasso e terem uma gestão mais controlada.

Os modelos de maturidade apoiam esse mesmo processo de adoção do SOA como um *roadmap*. Existem vários modelos de maturidade para adotar SOA numa organização, nós referimos alguns na secção 4.2.1- Modelos de Maturidade, mas estes modelos têm diversas falhas.

A **primeira falha** é que os modelos, exceto o OSIMM, não são independentes da tecnologia, eles são criados por vendedores de software ou empresas de consultoria.

A **segunda falha** reside no facto de modelos como o SOAMM-Microsoft não se preocuparem com a gestão e questões organizacionais. Adotar SOA é um caminho longo e complexo. A compreensão e compromisso de gestão é essencial para facilitar a sua adoção.

A **terceira falha** é onde a minha dissertação se foca, na fraca ferramenta da avaliação do SOA nos modelos já existentes. Usar uma ferramenta de avaliação permite à organização avaliar o seu próprio estado de maturidade, o que necessita para obter um melhor estado de maturidade e como obter um plano de melhoria, *recommend process*.

Embora alguns dos modelos de maturidade apresentados na secção 3.5- Modelos de Maturidade SOA, possuam os seus próprios métodos de avaliação, decidimos refinar o método de avaliação OSIMM, que é o mais completo e tentar criar uma avaliação mais ponderada.

4. Objetivos da solução

Neste capítulo iremos proceder à definição dos objetivos da solução do DSRM, que tem em conta o trabalho relacionado, apresentado nos capítulos anteriores, as limitações existentes nas soluções encontradas e também o que é possível fazer para resolver o problema apresentado.

Tendo em conta o problema que identificámos no Capítulo 2, a principal falha é o facto dos modelos de maturidade basearem a sua avaliação em *AQ* genéricas, logo não existe uma forma de avaliar com maior precisão e com resultados concretos o nível de maturidade de uma organização que adotou SOA.

O principal objetivo é, portanto, criar um método que permita a qualquer organização fazer um *self-assessment* da sua arquitetura orientada a serviços. Com base nesses resultados saber qual a estratégia, o *roadmap* que a organização pretende evoluir. Este objetivo pode ser dividido em duas partes:

- Criação dos critérios para as *AQ* do modelo de maturidade OSIMM, baseados na literatura e no *feedback* dado por especialistas na área de *Enterprise Architecture.*
- Usar este conjunto de critérios para a elaboração do método de avaliação, permitindo uma avaliação mais precisa do nível de maturidade de uma arquitetura orientada a serviços.

Ao juntar estas duas partes conseguimos criar um método que permite avaliar o nível de maturidade do SOA, com resultados concretos e adaptados à organização, tornando tomadas de decisões rápidas.

Existem objetivos específicos que queremos que a nossa proposta de solução cumpra:

- Solução aplicável a qualquer tipo organização;
- Clarificar o nível de maturidade dos serviços de uma organização;
- Ser facilmente compreendida por qualquer decisor/entrevistado, não apenas por especialistas;
- Permitir análises através dos resultados do método de avaliação;
- Construir um roadmap estratégico para a organização.

Estes objetivos ajudaram-nos a definir um planeamento para a construção e desenvolvimento do método proposto, que foi baseado no problema, nas necessidades identificadas e em aspetos que achamos importantes como a abstração, originalidade, justificação e benefício (Osterle, et al., 2011).

5. Proposta

Este capítulo corresponde à fase de **conceção** e **desenvolvimento** do DSRM, na qual criamos o artefacto. O artefacto proposto tem o objetivo de solucionar o problema identificado para a nossa investigação.

O problema (Capítulo 2) que identificámos, é: **Como avaliar com maior precisão o nível de maturidade de uma organização que adoptou SOA?**

Este capítulo está dividido em duas secções, que correspondem aos passos do método em que a nossa proposta se baseia. O **primeiro passo** corresponde aos objetivos que pretendemos que a nossa proposta cumpra e quais os fundamentos para a elaborar. (Secção 5.1)

O **segundo passo** corresponde à estruturação do método de avaliação que implica a elaboração dos critérios para as *AQ* do modelo de maturidade OSIMM. (Secção 5.2)

A nossa proposta vai ajudar os decisores na adoção do SOA para a organização. Através da análise dos resultados do *assessment*, o decisor pode avaliar qual o *roadmap* que a sua organização pretende seguir.

A nossa proposta é aplicar este método de avaliação a quatro organizações e obter um nível de maturidade preciso por dimensão e no global.

5.1. Objetivos da Proposta

Como analisamos anteriormente no Capítulo 3, os métodos de avaliação de maturidade de SOA são muito genéricos, permitindo ter resultados por vezes não precisos. Por isso decidimos criar critérios para obter uma menor desambiguidade nas respostas do modelo OSIMM.

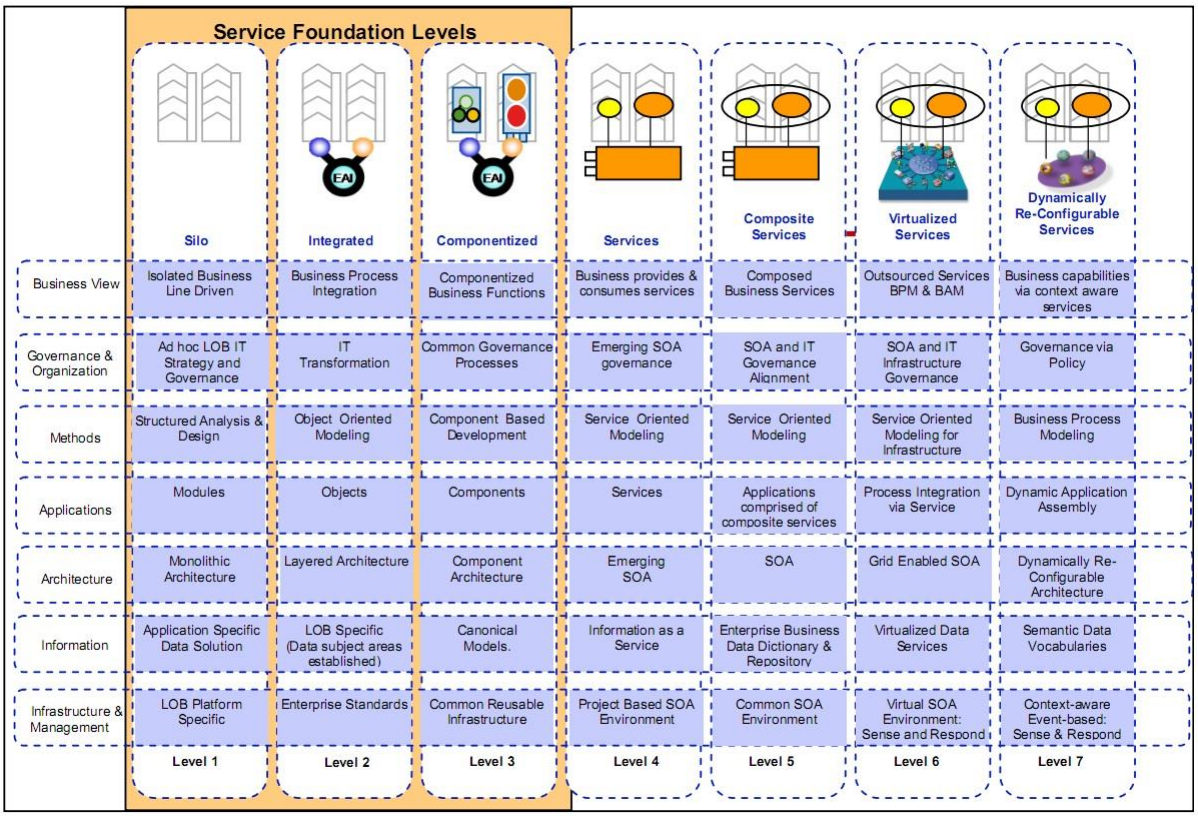


Figura 5 - OSIMM Maturity Matrix **(Group, The Open Group Architecture Framework (TOGAF), 2007)**

O modelo-matriz assente neste trabalho de investigação, no qual vão ser baseadas as nossas *AQ,* é o OSIMM, representado na Figura 5 - OSIMM Maturity Matrix.

O OSIMM é um modelo estandardizado para ajudar as organizações na sua adoção do SOA, aumentando os níveis de maturidade nos sete aspetos da organização e representa uma orientação de como melhorar a maturidade de cada serviço, mediante uma certa dimensão. (Group, Service Integration Maturity Model (OSIMM), 2011)

A nossa proposta vai ajudar os decisores a realizar melhores avaliações, definindo critérios de resposta concretos para as respetivas perguntas, escalonados de 0 a 7, correspondentes aos níveis de maturidade dos serviços do OSIMM, e mais duas opcionais que são *"Do not know"* e *"Do not answer".*

Desta forma, conseguimos que seja possível atribuir um valor quantificável para a resposta a essa pergunta, como explicado na secção 5.2- Estruturação do método de avaliação.

5.2. Método de Avaliação

A abordagem descrita anteriormente começa com a estruturação do problema. Neste passo estruturamos o nosso método de avaliação com as várias fases descritas na Figura 6.

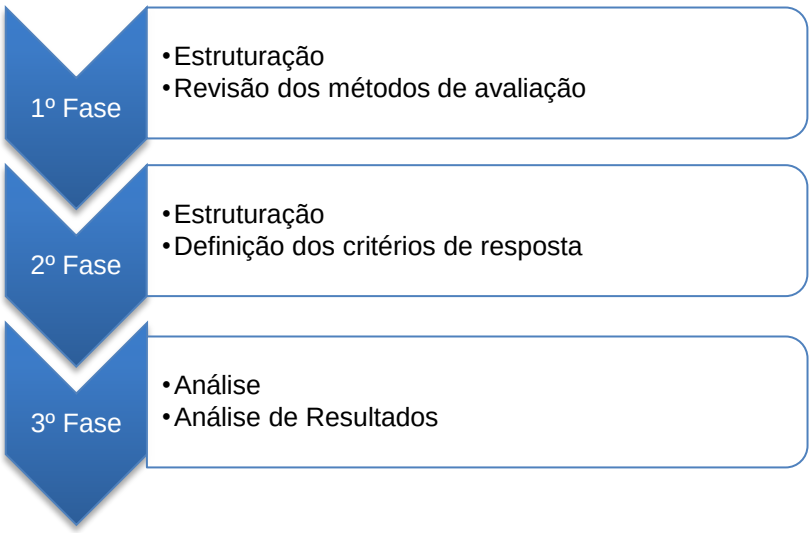


Figura 6- Fases propostas para a construção do método de avaliação

Na **primeira fase** começamos por fazer uma revisão a todos os tipos de métodos de avaliação utilizados nos modelos de maturidade já existentes, como descrito no Capítulo 3.

A **segunda fase** consiste na criação dos critérios de resposta, com base na pergunta, para o nosso método de avaliação que são elaborados tendo em conta as *AQ* que o OSIMM disponibiliza.

Para a criação dos critérios começámos por fazer uma análise de todas as perguntas do OSIMM, sendo estas 89 e criar 7 critérios de resposta para cada pergunta, que vai definir o nível de maturidade dos serviços. Os critérios foram elaborados com base na revisão de literatura.

Para a revisão da literatura seguimos a metodologia proposta por (Jane & Richard, 2002) fazendo uma extensa pesquisa, com algumas restrições temporais. Para a nossa pesquisa literária usámos o motor de busca *Google Scholar* bem como o *IEEE Xplore* e também CBDI Forum que possui *experts* na área de SOA com alguns artigos importantes e aceites pela comunidade científica.

Na pesquisa inicial encontrámos 100 artigos. Através de uma análise do resumo, introdução e conclusão, identificámos os artigos que falavam em *Maturity Models, SOA, Services in IT, etc*. Fizemos então uma revisão cuidada desses artigos, com o objetivo de encontrar aqueles que pudessem ser uma mais valia para a nossa investigação, acabando por selecionar 18 artigos como base para a criação do conjunto de critérios do nosso método de avaliação (ver Anexo A).

O resultado foi a definição de **623 critérios** de respostas para o método de avaliação do OSIMM, que corresponde a 89 tipos de perguntas das dimensões a multiplicar pelos 7 níveis de maturidade dos serviços. Na Tabela 3 estão representadas as duas primeiras perguntas com os 7 critérios de resposta definidos da Business Dimension. Os restantes critérios de resposta encontram-se em Anexos B - Critérios de resposta para as AQ do OSIMM.

Pergunta 1) How are identified the major business drivers for this initiative?
There are no business drivers

There are some business drivers documented and explicits in business process integration
Organization's business drivers are documented as cross-organizational business objectives.
The business drivers are identified throughout organization internal processes
The business drivers are identified throughout organization external processes
The business drivers are identified throughout organization internal and external processes
The bussiness drivers are adapted to frequent business process changes
Do not know
Do not answer
Pergunta 2) What is the business vision and goals, and how are these related to what IT is currently doing?
There is no vision of a future SOA situation;
Some IT people have thought about the future SOA situation;
There is a vision of a future SOA situation within some business critical projects;
Only the board directors have a vision of future SOA situation
Most IT staff of the organization share a vision of future SOA situation;
There is one clear vision about future SOA situation shared throughout the organization
There is one clear vision about future SOA situation shared throughout the organization and on a inter-organizational level;
Do not know
Do not answer

Tabela 3- Critérios elaborados para as perguntas 1 e 2 da *Business Dimension*

Por fim, na ***terceira fase***, temos a análise dos resultados do novo método de avaliação nas organizações.

Ao formular a nossa proposta, validamos o nosso catálogo de critérios através de entrevistas a especialistas do IT de organizações para demonstrar a nossa proposta, respondendo com uma avaliação de 0 a 5 utilizando o método *likert scale* (MUNSHI, 2014) com base em duas métricas ***"Is usefull for SOA"***, ou seja, em que medida o conjunto critérios criados por nós, para as perguntas do OSIMM são úteis para avaliar o SOA e ***"Is easy to answer"*** que nos permite saber se os critérios que definimos para as perguntas do OSIMM são demasiado complexos. Esta avaliação está descrita na Secção 7.2.

Com a informação descrita anteriormente, alegamos que o nosso artefacto propõem melhorias a fim de ajudar os decisores a tomar uma decisão. A solução da nossa investigação propõe **um método mais eficaz no cálculo do nível de maturidade de uma organização que adoptou SOA.**

6. Demonstração

Este capítulo corresponde à **demonstração** é a quarta fase de uma iteração do DSRM, que consiste em demonstrar o uso do artefacto para resolver uma ou mais instâncias do problema.

O objetivo da nossa proposta é criar um método de avaliação **mais eficaz no cálculo do nível de maturidade de uma organização que adoptou SOA.** Para fazer a demonstração escolhemos a Ordem dos Advogados (OA), que é uma organização de caráter privado e heterogénea na sua arquitetura orientada a serviços que possui, embora também tenhamos efectuado outros *assessments* a instituições como a um Instituto Público que é uma organização de caráter público, a Câmara Municipal de Mirandela (CMM) que é uma câmara municipal com um baixo nível populacional, a Marinha Portuguesa que é uma organização Governamental, e a Link que é uma empresa que presta serviços na área de desenvolvimento de software.

Os decisores foram todos elementos do IT pertencentes à organização.

6.1. Estruturação do *Assessment*

Este passo começa com a definição dos critérios aceites pelos entrevistados e o *assessment* efectuado às organizações.

Na Secção 5.2 descrevemos o processo de investigação para a criação de um catálogo de critérios para o método de avaliação de maturidade para uma arquitetura orientada a serviços que fizesse sentido em todas as organizações.

Na nossa demonstração, os entrevistado(s) usaram os 623 critérios devido a enquadrarem-se necessários para o método de avaliação como sendo *"Is usefull for SOA"* e *"Is easy to answer"*, não sendo nenhum critério rejeitado como explicado na Secção 7.2.

Com o nosso método conseguimos representar graficamente o nível de maturidade dos serviços em que a organização se encontra, em cada dimensão e quais as caraterísticas, dependendo da seleção da resposta. Os pesos atribuídos às repostas foram complementados com os que o OSIMM já possui, nós apenas definimos mais descritivamente uma possibilidade de resposta mais assertiva.

Para o *assessment* foi utilizado um questionário elaborado no *Google Forms* que nos permitiu elaborar os gráficos abaixo representados com os casos de estudo efectuados.

O *assessment* com várias organizações permitiu-nos avaliar o nível de maturidade da arquitetura orientada a serviços usando o nosso método de avaliação adaptado com o modelo OSIMM.

6.2. Ordem dos Advogados

Para fazer a demonstração escolhemos a Ordem dos Advogados pela facto de ser uma organização privada que presta serviços a todos os advogados que exercem a profissão de advocacia e porque o seu Chefe de departamento de IT estava interessado em participar na avaliação desta dissertação.

O OSIMM pode ser usado para suportar o SOA *assessment* para uma única dimensão ou para as 7 dimensões de uma organização. O propósito do OSIMM *assessment method* é avaliar o nível de maturidade atual e determinar qual o nível de maturidade desejado *(goal state)* necessário para ir ao encontro dos objetivos do negócio.

De seguida visualizamos os resultados das 7 dimensões a que o chefe de IT da Ordem dos Advogados respondeu e transcrevemos em representação gráfica para uma análise mais ilustrativa.

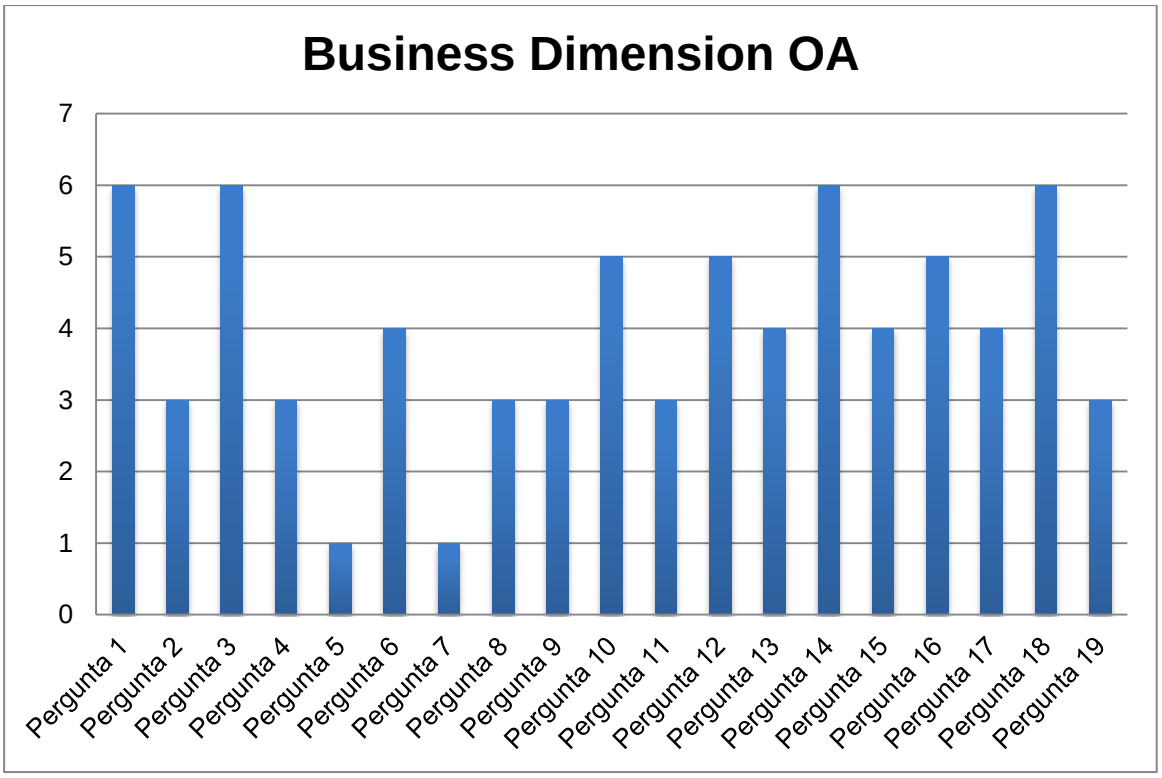


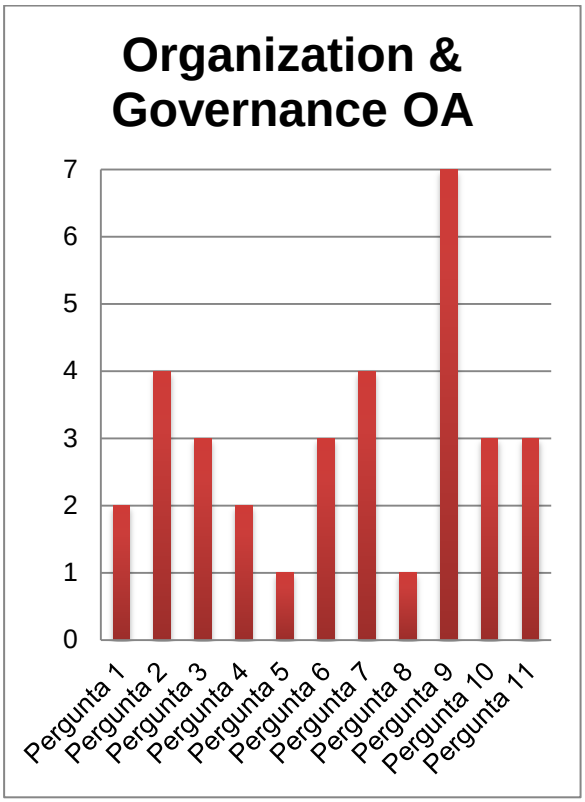
Figura 7- *Assessment Business Dimension*

A Figura 7 representa os critérios que escolheu para as perguntas efectuadas do *assessment* na *business dimension*. A *business dimension* determina o quão a empresa tem os processos formalmente definidos e documentados segundo os seus *business drivers*. Como podemos analisar na Figura 7, o nível de maturidade que se encontra mais baixo é visualizado nas perguntas 5 e 7, em que as respostas foram as seguintes:

Pergunta 5) How are metrics for return-of-investment measured in Business Process Management (BPM)?
1. There are no defined metrics for ROI in BPM
Pergunta 7) What is the current cost model?
1. We don't have a cost model

Tabela 4 – Respostas do assessment Business Dimension

O nível de maturidade atingido pela OA nesta dimensão foi de **3,95** o que refere que os seus serviços estão *"Componentized Business",* ou seja, *"Organization's business drivers are documented as* ***cross-organizational*** *business objectives."*



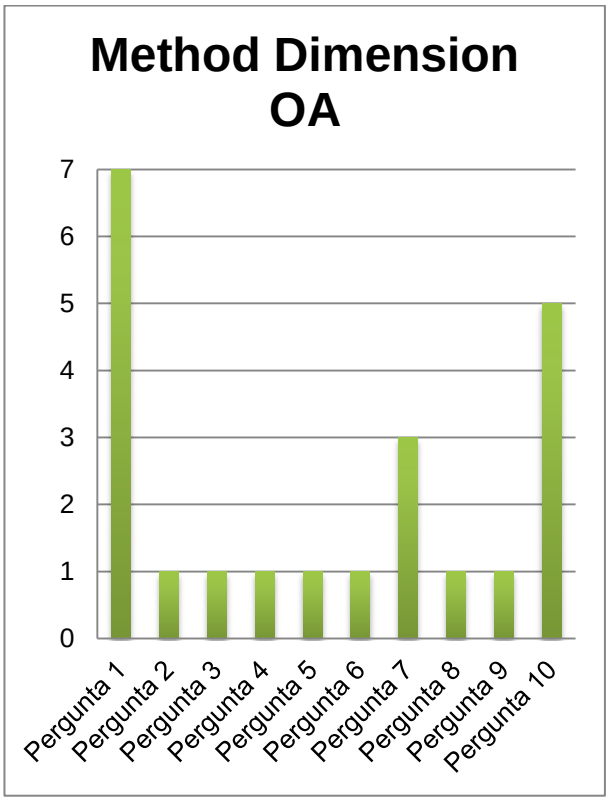


Figura 8- Assessment Organization & Governance **Figura 9– Assessment da Method Dimension**

A Figura 8 representa os critérios que escolheu para as perguntas efectuadas do *assessment* na *Organization & Governance Dimension*. A *Organization & Governance Dimension* determina o quão *"identifying the formal use of service and SOA governance across the organization to develop, deploy, and manage business and IT services (SOA solutions)."* Como podemos analisar na Figura 8 descrita acima, o nível de maturidade que se encontra mais baixo é visualizado nas perguntas 5 e 8, em que as respostas foram as seguintes:

Pergunta 5) Is the interaction between organizations involved in the SOA process defined with clear roles and responsibilities?
1. There is no roles and responsabilities defined
Pergunta 8) What type of SOA training is available in your IT organization?
1. There is no SOA training

Tabela 5 - Respostas do assessment Organization & Governance

O nível de maturidade atingido pela OA nesta dimensão foi de **3,00**, o que refere que os seus serviços estão "*Common SOA Governance Processes",* ou seja, *"Formal use of service and SOA governance* ***across the organization*** *to develop, deploy, and manage business and IT services (SOA solutions)"*

A Figura 9 representa os critérios que escolheu para as perguntas efectuadas do assessment na *Method Dimension.* A *Method Dimension* determina o quão "*can be conducted by identifying the*

formal use of an SOA architectural design, construction, and deployment methodology for the implementation of SOA services". Como podemos analisar na Figura 9 descrita acima, o nível de maturidade que se encontra mais baixo é visualizado nas perguntas 2, 3, 4, 5, 6, 8, 9, em que as respostas foram as seguintes:

Pergunta 2) What design methodologies and best practices are you currently adopting?
1. We don't use SOA design methodologies
Pergunta 3) Do you practice any SOA design techniques?
1. We don't use any formal SOA design techniques
Pergunta 4) What design tools are in practice today?
1. We don't use design tools
Pergunta 5) What is the current practice for service development and management?
1. We don't use service development neither management methodology
Pergunta 6) What is your current project management framework?
1. We don't have project management framework
Pergunta 8) What are your organization's current QA processes?
1. We don' have Quality Assurance (QA) in our organization
Pergunta 9) Do you have an active community that works to evolve your SOA methods and practices?
1. We don't have active community to envolve SOA methods and practices

Tabela 6 - Respostas do assessment Method Dimension

O nível de maturidade atingido pela OA nesta dimensão foi de **2,00**, o que refere que os seus serviços estão *"Object-oriented Modeling",* ou seja, que o *"formal use of an SOA architectural design, construction, and deployment methodology for the implementation of services* ***is limited****."*

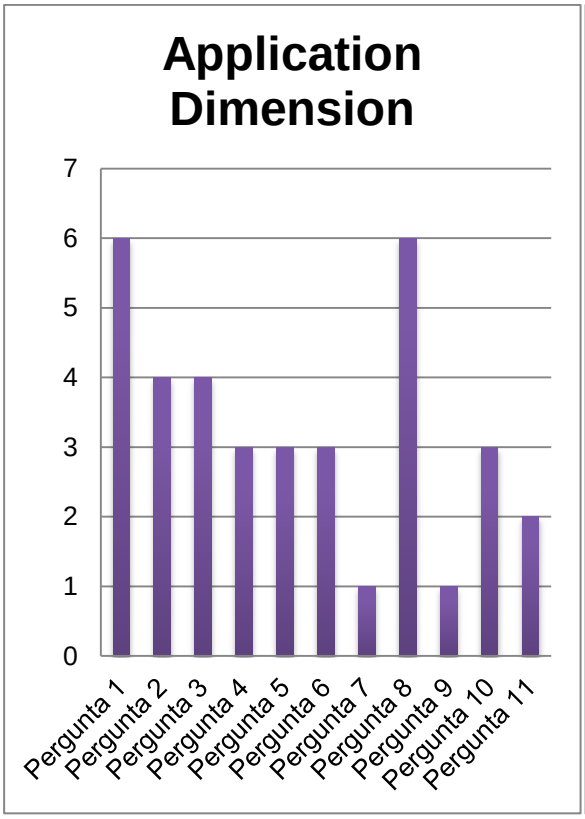


Figura 10 -Assessment Application Dimension

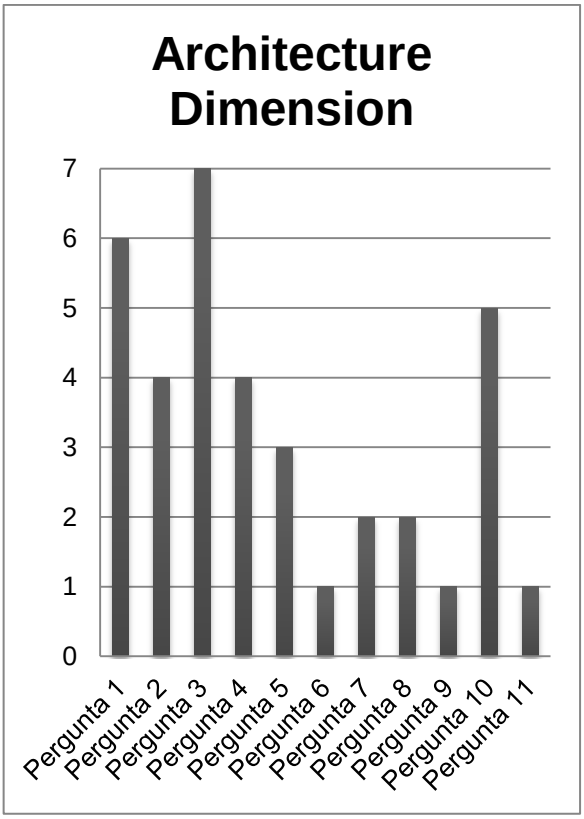


Figura 11- Assessment Architecture Dimension

A Figura 10 representa os critérios que escolheu para as perguntas efectuadas do *assessment* na *Application Dimension*. A *Application Dimension* determina o quão "*can be conducted by identifying the application architectures that are designed and implemented using SOA principles and development practices and utilize constructs such as loose-coupling, separation of concerns, and employ the use of service-enabled technologies such as XML, web services, service bus, service registries, and virtualization.*" Como podemos analisar na Figura 10 descrita acima, o nível de maturidade que se encontra mais baixo é visualizado nas perguntas 7 e 9, em que as respostas foram as seguintes:

Pergunta 7) How is business logic represented within your organization's applications?
1. Business logic has not been decomposed
Pergunta 9) How widely is XML used in your organization? How sophisticated is its use?
1. We don't use XML language in our organization

Tabela 7 - Respostas do assessment Application Dimension

O nível de maturidade atingido pela OA nesta dimensão foi de **3,27**, o que refere que os seus serviços estão *"Componetized/Components",* ou seja, *"Application architectures are designed and implemented using SOA principles and development practices that utilize constructs such as loose-coupling, separation of concerns, and employ the use of service-enabled technologies such as XML, web services, service bus, service registries, and virtualization* ***is cross-organizational****".*

A Figura 11 representa os critérios que escolheu para as perguntas efectuadas do *assessment* na *Architecture Dimension*. A *Architecture Dimension* determina o quão "*can be conducted by identifying those service components that have been designed and are deployed using formal SOA methods, principles, patterns, frameworks, or techniques.*" Como podemos analisar na Figura 11 descrita acima, o nível de maturidade que se encontra mais baixo é visualizado nas perguntas 6, 9 e 11, em que as respostas foram as seguintes:

Pergunta 6) How mature are your services implementations?
1. We don't have metrics to verify how mature are our services
Pergunta 9) How extensive and sophisticated is your organization's use of frameworks in your architecture?
1. There is no framework to support our architecture
Pergunta 11) Does your organization use reference architectures?
1. We don't have metrics to verify how mature are our services

Tabela 8 - Respostas do assessment Architecture Dimension

O nível de maturidade atingido pela OA nesta dimensão foi de **3,27**, o que refere que os seus serviços estão *"Componetized/Component Architecture",* ou seja, *"Service components are designed using formal SOA methods, principles, patterns, frameworks, or techniques* ***is cross-organizational****"*

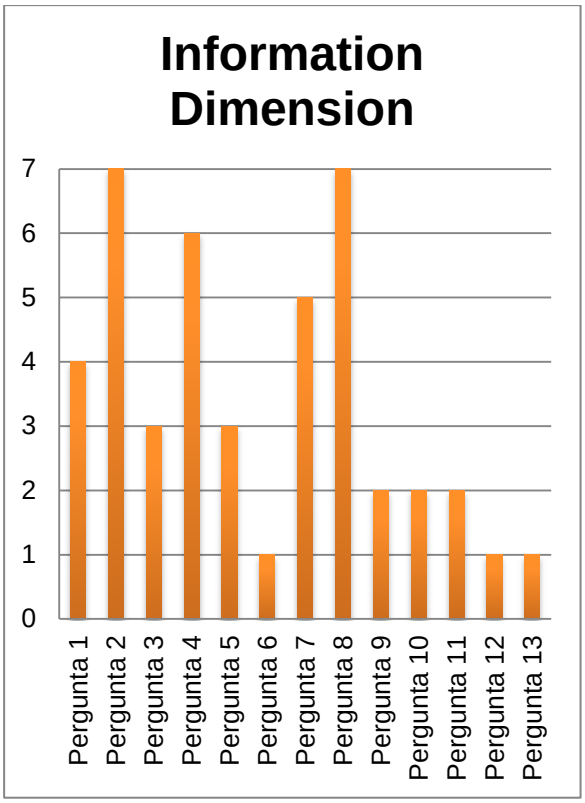


Figura 12 – Assessment *Information Dimension*

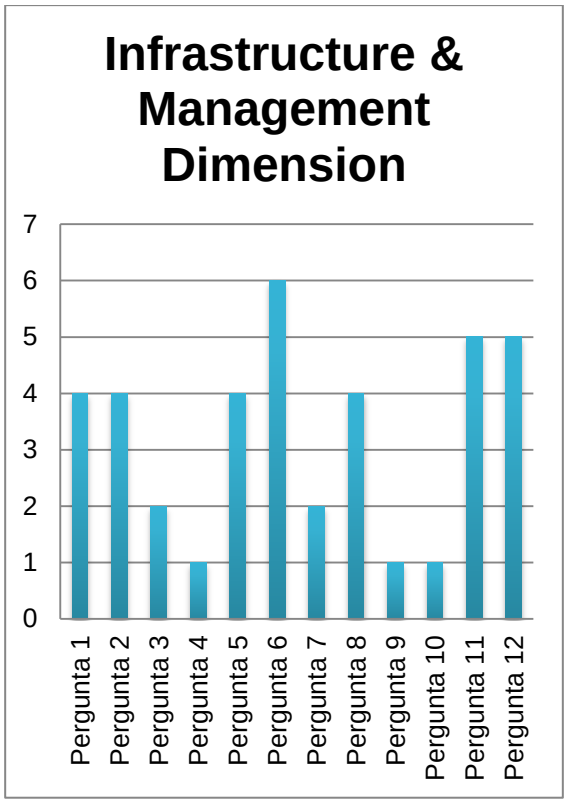


Figura 13 – Assessment Infrastructure & Management Dimension

A Figura 12 representa os critérios que escolheu para as perguntas efetuadas do *assessment* na *Information Dimension*. A *Information Dimension* determina o quão *"can be conducted by identifying the information architecture that supports a master data model (federated data service) and implements a common business data vocabulary."* Como podemos analisar na Figura 12 descrita acima, o nível de maturidade que se encontra mais baixo é visualizado nas perguntas 6, 12 e 13, em que as respostas foram as seguintes:

Pergunta 6) Are the data models in the form of Business Object Models, understandable to and owned by, the business, or as IT object models, understandable only to, and owned by, the IT teams?
1. The data models are not in form of Business Object Models
Pergunta 12) Are there facilities for performing complex inference in order to map data defined in ontologies from one form to another? Does a master data service exist?
1. There is not any defined data mapping in ontologies
Pergunta 13) Does your organization have or are you developing a Business Information Model to standardize data and message formats and concepts across the enterprise?
1. We don't have any Business Information Model

Tabela 9 - Respostas do assessment Information Dimension

O nível de maturidade atingido pela OA nesta dimensão foi de **3,4**, o que refere que os seus serviços estão *"Componetized/Canonical Models",* ou seja, *"The information architecture supports a master data model that implements a common business data vocabulary* ***is cross-organizational****".*

A Figura 13 representa os critérios que escolheu para as perguntas efectuadas do *assessment* na *Insfrastructure & Management Dimension*. A *Insfrastructure & Management Dimension* determina o

quão *"can be conducted by identifying the IT infrastructure that supports the non-functional and operational requirements and SLAs needed to operate an SOA environment."* Como podemos analisar na Figura 13 descrita acima, o nível de maturidade que se encontra mais baixo é visualizado nas perguntas 4, 9 e 10, em que as respostas foram as seguintes:

Pergunta 4) Have you defined any SLAs around security and privacy? How is this measured and monitored?
1. We don't have any SLA's defined for security and privacy
Pergunta 9) What tools are used for configuration management?
1. We don't use any tool to configuration management
Pergunta 10) Do you have mechanisms for looking up global objects by searching on their characteristics?
1. We don't have mechanisms for looking up global objects

Tabela 10 - Respostas do assessment Infrastructure & Management Dimension

O nível de maturidade atingido pela OA nesta dimensão foi de **3,25**, o que refere que os seus serviços estão *"Componetized/Common Re-usable Infrastructure",* ou seja, *"the IT infrastructure supports the non-functional and operational requirements and SLAs needed to operate an SOA environment* ***is cross-organizational****"*

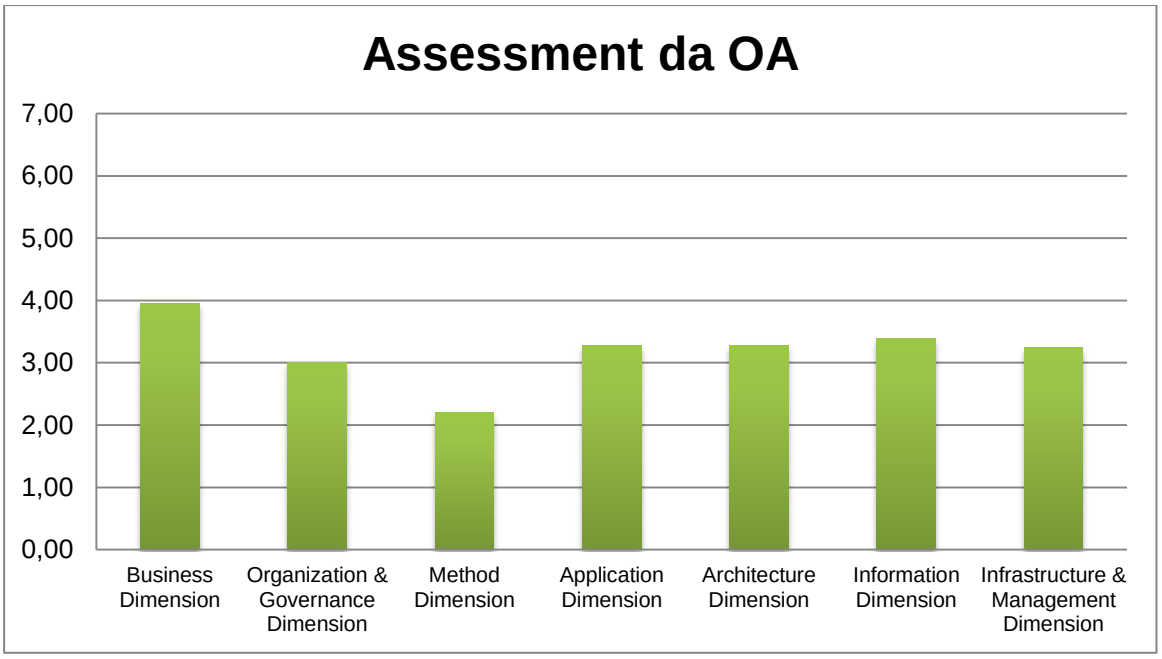


Figura 14 - Nível de maturidade atingida na OA

Depois da análise exaustiva de cada uma das dimensões, em que analisámos dimensão a dimensão o nível de maturidade atingido, conseguimos, com o nosso método de avaliação, verificar quais os critérios que tinham um nível de maturidade baixo com base nas respostas, logo conseguimos definir de um modo facilitado um *roadmap* para a nossa organização.

Acima, na Figura 14, encontra-se o nível de maturidade atingido em todas as dimensões do qual concluímos que o **valor de maturidade atingido pela OA foi de 3,19**, sendo este um numero exato, conseguimos ter perceção se estamos mais perto do nível de maturidade 3 ou 4.

6.3. Análise dos resultados da *assessment* geral

Nesta secção apresentamos o *assessment* de todas as organizações no qual aplicamos o nosso método de avaliação de maturidade de uma arquitetura orientada a serviços.

Os respectivos resultados obtidos foram efectuados à Ordem dos Advogados, Câmara Municipal de Mirandela, Instituto Público e a Link são os seguintes representados na Figura 15.

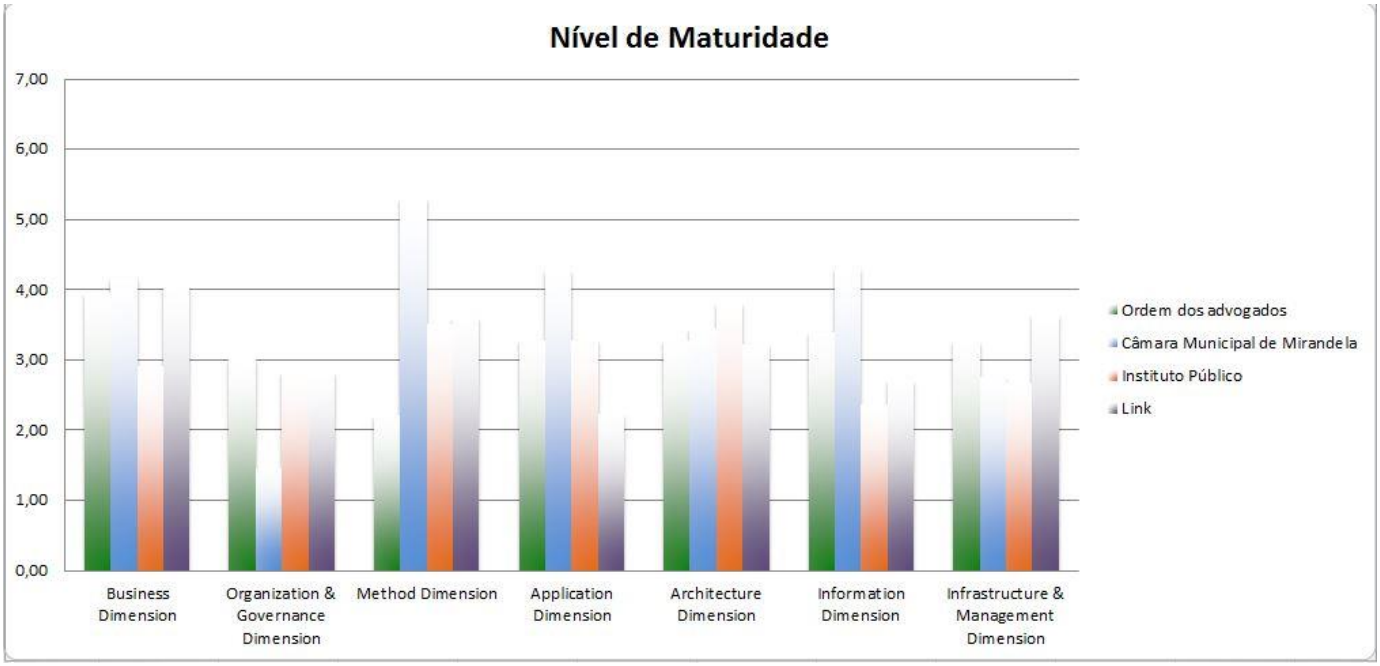


Figura 15 - *Assessment* das organizações

Níveis de Maturidade
Ordem dos Advogados 3,19
Câmara Municipal de Mirandela 3,67
Instituto Público 3,05
Link 3,18

A que têm maior destaque é a CMM que possui um nível de maturidade de 3,87.

Através da análise dos resultados podemos constatar que existe uma certa coerência nos mesmos, isto porque, efetivamente, a CMM não desenvolve o software próprio da câmara, apenas gere, quem efetua o desenvolvimento é a Medidata que, como é óbvio, tem de ter um nível de maturidade maior, até porque uma câmara tem uma estrutura bastante complexa e a Medidata, segundo fontes do Chefe de IT da CMM, é um software usado em grandes câmaras municipais, abrangendo um grande numero de serviços.

A que possuí um nível de maturidade mais baixo é a LINK com 3,18.

A Link, sendo ela uma *software house*, estando envolvida em *SOA Projects*, não obteve um nível de maturidade tão elevado, porque a sua avaliação foi efetuada com base nos serviços que a empresa tinha e não naqueles que prestava a terceiros.

Como podemos verificar pelos resultados obtidos podemos concluir que todas as organizações estão situadas no nível de maturidade 3.

O nível de maturidade 3, segundo OSIMM, refere que “The IT systems in the silos have been analyzed and broken down into component parts, with a framework in which they can be developed into new configurations and systems. There may also be some limited analysis of the business functionality into components. Although components interact through defined interfaces, they are not loosely coupled, which limits agility and interoperability between different segments of the organization (or even different organizations within the business “eco-system”). This causes difficulties in development and deployment of shared business processes. Business and infrastructure components are discrete and re-usable through code and EAI re-use techniques. However, they are often replicated and redundant.”

6.4. Resumo

O caso de estudo que apresentámos permitiu-nos testar a nossa proposta como solução ao problema de investigação que apresentamos no início desta dissertação. O entrevistado reconheceu que a nossa proposta foi bastante útil para entender “onde” e “em que moldes” existe o problema e que o método de avaliação desenvolvido foi importante para uma tomada de decisão futura.

Em suma, provámos que o método que estamos a propor permite ajudar o decisor a comparar alternativas e tomar a sua decisão, o que responde diretamente ao problema da nossa investigação.

7. Avaliação

Este capítulo está relacionado com a etapa de **avaliação** do DSRM . O objetivo desta capítulo é mostrar que a proposta suporta a solução para o problema da nossa investigação, descrito anteriormente. Começaremos com a descrição das entrevistas que fizemos para a avaliação da proposta de tese (Secção 7.1). Depois disto apresentaremos a avaliação das demonstrações (Secção 7.2), usaremos a Moody and Shanks Quality Framework (Secção 7.3) e ainda os quarto princípios de Österle (Seção 7.4) para avaliar o nosso artefacto da investigação.

7.1. Entrevistas a Profissionais

O objetivo das entrevistas com profissionais serve, em primeiro lugar, para validar a nossa investigação, o problema, a sua relevância e o método. É uma técnica usada para conseguir recolher informação qualitativa, que permitirá ao entrevistado dar a sua opinião e experiência sobre um determinado assunto. O objetivo é entender o ponto de vista dos profissionais que trabalham na área de investigação, de forma a conseguir *inputs* daqueles que trabalham diariamente com estas questões, em vez de fazermos generalizações e suposições.

Existem três tipos de entrevistas (Bryman, 2012)

- Não-estruturadas : quando o investigador tem um plano definido, mas o mínimo control sobre as respostas dos entrevistados.

- Semi-estruturadas : quando o investigador usa um guia de questões e tópicos que quer ver cobertos.

- Estruturadas : quando o investigador tem perguntas fixas e elas são respondidas numa ordem específica.

7.1.1. Resultados da avaliação

A nossa solução foi desenhada com base na literatura da área, o que nos deu bases teóricas fortes para o início da nossa investigação. Porém, achámos que seria importante ter também o ponto de vista dos profissionais de IT, que trabalham nesta área, diáriamente. Posto isto, entrevistámos 5 profissionais de IT de organizações com uma maior dimensão até organizações com uma menor dimensão, em média 15 anos de experiência em Tecnologias de Informação.

O nosso objetivo foi perceber quais os critérios que são realmente relevantes para avaliar o nível de maturidade de uma arquitetura orientada a serviços nas organizações através do nosso método.

Como base de trabalho, optamos por usar o modelo OSIMM. A escolha deste modelo foi já justificada anteriormente (Secção 3.5).

As entrevistas foram semi-estruturadas e consistiram no preenchimento de um questionário através do *Google forms* com uma duração de 60 minutos, aproximadamente. Para suportar a entrevista criamos um questionário (ver anexo C) para que os entrevistados nos indicassem quais os critérios que nós criamos que poderiam ser úteis para o nosso método de avaliação para avaliar a maturidade de arquiteturas orientadas a serviços.

Os resultados são apresentados na figura 16:

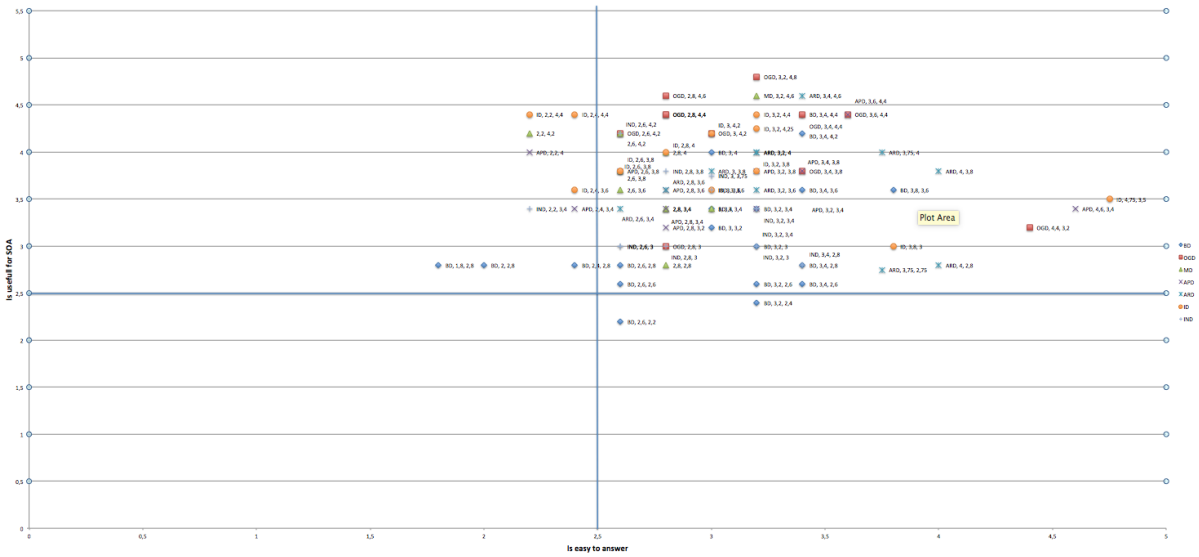


Figura 16 - Análise do resultado das entrevistas

Os resultados revelam que todos os critérios estão diretamente relacionados com a avaliação de uma arquitetura orientada a serviços. Apesar de alguns critérios terem tido uma avaliação mais fraca, nenhum está no 3º quadrante, o que revela que são todos úteis. Depois da análise, constamos que 79 critérios tiveram uma avaliação superior a 2,5 de 0 a 5 quanto às métricas que definimos, *"Is easy to answer"* e *"Is usefull for soa".* Como podemos visualizar na Figura 16, todos os critérios encontram-se no 1º quadrante e apenas 10 critérios encontram-se nos restantes quadrantes.

Posto isto, usámos todos os critérios para a elaboração do nosso método de avaliação, pois este método tem critérios necessários e suficientes para avaliar uma arquitetura orientada a serviços. Neste caso todos os critérios foram importantes.

Depois da análise de resultados construímos a nossa proposta final que está referida no Capítulo 6.

7.2. Avaliação das Demonstrações

O objetivo da avaliação da demonstração é validar se a demonstração feita no estudo de caso serviu para testar e poder afirmar que a nossa proposta é valida para resolver o problema identificado no início da nossa investigação. Nesta avaliação terá de ser justificado como é que a instância do problema identificado foi resolvida através da nossa proposta e quais foram as vantagens e implicações no estudo de caso específico.

Através da aplicação da nossa proposta, conseguimos efetuar uma avaliação mais rigorosa do nível de maturidade da arquitetura orientada a serviços de uma organização segundo uma certa dimensão, tendo em conta as respostas dos entrevistados. Os entrevistados afirmaram que a nossa proposta os ajudará a tomar decisões e que o processo foi simples e perceptível no seu entender.

Através dos resultados, conseguimos analisar dimensão a dimensão e pergunta a pergunta segundo os critérios que escolheu, identificando assim quais são os principais pontos de falha na sua arquitetura orientada a serviços existentes na organização. Depois disto inclusivamente propor melhorias para o entrevistado melhorar a sua própria arquitetura SOA, *um roadmap*, com base no modelo OSIMM.

Resumindo, a demonstração que efetuamos foi bem sucedida pois conseguiu cumprir com todos os objetivos, resolvendo o problema da investigação com a nossa proposta para uma instância específica do problema. Foi por isso um caso de estudo apropriado que nos permitiu obter resultados muito interessantes para o entrevistado e para a dissertação.

7.3. Moody and Shanks

Uma *Framework* para avaliar a qualidade do modelo produzido na perspetiva dos múltiplos *stakeholders* . A *framework* propõe oito fatores de qualidade (Moody & Shanks, 2003):

- **Perfeição**: o artefacto contém todos os requisitos do utilizador.
- **Simplicidade**: o artefacto contém o número mínimo possível de entidades e relações.
- **Flexibilidade**: a facilidade com que o artefacto suporta o negócio e/ou alterações regulamentares sem necessidade de se alterar.
- **Integração**: a consistência entre o artefacto e a organização.
- **Compreensibilidade**: a facilidade com que os conceitos e estruturas do artefacto podem ser compreendidos.
- **Implementabilidade**: a facilidade com que o artefacto pode ser implementado de acordo com restrições definidas.

- **Integridade**: as regras e/ou restrições do negócio são definidas a partir dos requisitos dos utilizadores para garantir a integridade do artefacto.

- **Exatidão**: é definida se o modelo for válido, ou seja, se está de acordo com regras e convenções.

7.3.1. Resultados da Avaliação

Fizemos a demonstração da nossa proposta na Ordem dos Advogados, uma instituição privada, e os resultados seguintes foram obtidos após aplicar a *Moody and Shanks Quality Management Framework* à mesma. Também foram tidas em conta as entrevistas feitas a profissionais e os seus resultados. Os resultados são:

- **Perfeição**: a proposta foi analisada tendo em conta os critérios de avaliação, a informação que está diretamente relacionada com o método. No geral, a nossa proposta está completa, tendo em conta que os critérios de avaliação necessários para avaliar a maturidade de uma arquitetura orientada a serviços estão presentes no catálogo que criámos.

- **Simplicidade**: o nosso artefacto é simples de seguir, tendo em conta a opinião do entrevistado, e confirmámos que é fácil de aplicar.

- **Flexibilidade**: o nosso artefacto é flexível, pois é possível utilizar em qualquer tipo de organização, analisando dimensão a dimensão ou no global.

- **Integração**: a nossa proposta é consistente com as necessidades e objetivos da organização, visto ser construída diretamente com o entrevistado. O resultado da nossa proposta será uma resposta a um método eficaz na avaliação de maturidade de uma arquitetura orientada a serviços, portanto ajudará o entrevistado a tomar uma decisão.

- **Compreensibilidade**: os profissionais consideraram a nossa proposta fácil de entender, pois os conceitos de arquiteturas orientadas a serviços utilizados são iguais aos conceitos de arquitetura tradicionais, existindo por vezes algumas perguntas com uma dificuldade de resposta com base nos nossos critérios devido a serem complexas.

- **Implementabilidade**: os profissionais mostraram interesse em usar a nossa proposta, porém a sua implementabilidade depende das políticas internas de cada organização; de qualquer forma, admitem usá-la como ferramenta de decisão futuramente.

- **Integridade**: é extremamente dependente do entrevistado, uma vez que não existe nenhum outro constrangimento para a definição dos critérios. A nossa proposta confia e baseia-se em entrevistas e observações.

- **Exatidão**: A proposta foi considerada válida para as intenções dos profissionais, porém a exatidão está dependente de cada organização.

Resumindo, quase todos os fatores de qualidade foram atingidos. Por outro lado, fatores como

integridade e compreensibilidade foram parcialmente atingidos. Isto justifica-se pois: para o primeiro fator, não existem regras ou restrições para suportar toda a integridade; para o segundo fator, nem todos os profissionais acharam a proposta fácil de compreender, no princípio. Mas depois de um período de adaptação já se sentiram confortáveis. Por outro lado, fatores como a implementabilidade não foram verificados pois há demasiada burocracia para implementar este tipo de soluções, se bem que nas organizações onde efetuamos o nosso *assessment* não impuseram qualquer constrangimento.

7.4. Princípios de Österle, et al.

Os princípios de Österle, et al. validam, se a nossa solução atinge os objetivos. Nós caracterizámos a nossa investigação de acordo com quatro princípios básicos (Osterle, Memorandum on Design-Oriented Information Systems Research, 2011):

- **Abstração**: o artefacto deve ser aplicável a uma classe de problemas.
- **Originalidade**: o artefacto deve contribuir substancialmente para o avanço do corpo de conhecimento.
- **Justificação**: o artefacto deve ser justificado de uma forma compreensível e deve permitir validação.
- **Benefício**: o artefacto deve trazer benefícios, no imediato ou no futuro, para os respetivos grupos de *stakeholders*.

7.4.1. Resultados da Avaliação

Aplicámos os quatro princípios propostos para avaliar o nosso artefacto. Os resultados são:

- **Abstração**: o nosso artefacto pode ser aplicado a qualquer organização para avaliar o nível de maturidade de uma arquitetura orientada a serviços.
- **Originalidade**: nenhum dos profissionais entrevistados tinha conhecimento de um artefacto semelhante que permitisse avaliar o nível de maturidade de uma arquitetura orientada a serviços desta forma, e também não encontramos no trabalho relacionado nada similar.
- **Justificação**: o artefacto tem como suporte a motivação da investigação e todo o trabalho relacionado. Além de que o próprio artefacto está descrito textualmente e por figuras de uma forma abrangente e muito objetiva, com passos claros e instruções. A demonstração também ajudou a explicar a aplicação real e bem sucedida do artefacto.
- **Benefício:** o artefacto traz, efetivamente, benefícios para o entrevistado e para a organização em si, pois permite que se faça uma avaliação do nível de maturidade de uma arquitetura orientada a serviços mais rápida e eficazmente e com sólidas bases científicas, dando informação ao entrevistado de qual é a melhor opção tendo em conta as suas particularidades e necessidades.

8. Comunicação

Este capítulo está diretamente relacionado com a etapa de comunicação do DSRM, que consiste em comunicar o problema, a sua importância e a proposta de resolução juntamente com o artefacto produzido, explicado e demonstrado a investigadores e outras audiências relevantes, inclusivamente a *enterprise architects.*

Decidimos então usar duas abordagens de comunicação, a primeira está relacionada com entrevistas e demonstrações da nossa investigação a profissionais da área; a segunda é a submissão de artigos científicos em revistas da área que se interessem pela nossa investigação.

Em relação à primeira abordagem, já descrevemos anteriormente, na avaliação, a forma como validámos o nosso artefacto junto dos profissionais de IT. O resultado desta comunicação foi perceber que a nossa proposta é válida, recebendo sugestões e melhorando em vários aspetos.

Em relação à submissão de artigos científicos em revistas da área, estamos a preparar um artigo para submeter a uma conferência científica:

- 17th IEEE Conference on Business Informatics

Nesta submissão, iremos identificar o problema da investigação, e também propor o nosso método para o resolver. Propomos o catálogo dos critérios de avaliação para uma arquitetura orientada a serviços, baseado em cinco entrevistas a profissionais de IT que os validaram.

Faremos as nossas recomendações após a análise dos resultados e ainda propostas de melhoria, que não faziam parte do método proposto inicialmente. O feedback do entrevistado foi muito positivo e indicou-nos que a nossa proposta foi útil para entender o estado dos serviços da organização.

9. Conclusão

Face ao problema identificado: **Como avaliar com maior precisão o nível de maturidade de uma organização que adoptou SOA?**, um problema encontrado quando as organizações necessitam de avaliar o seu estado de maturidade, a nossa investigação propõe como solução um **método mais eficaz no cálculo do nível de maturidade de uma organização que adoptou SOA.** Esta solução baseia-se em definir critérios de reposta escalonados de um a sete, correspondentes aos níveis de maturidade do modelo OSIMM para cada pergunta do mesmo.

A aplicação da nossa proposta inclui entrevistas a profissionais do IT de organizações de média e grande dimensão, relativamente às quais concluímos que existe uma necessidade de avaliarem o quão madura é a sua arquitetura de serviços, até para definir estratégias futuras da organização. Mostraram também muito interesse em perceber quais são as suas falhas a nível de integração dos serviços e como melhorar essas integrações.

Concluímos também, na pesquisa que efetuamos a métodos de avaliação para modelos de maturidade, que não existe um método objetivo e eficaz que permita avaliar a maturidade de uma arquitetura orientada a serviços. No mesmo sentido, as perguntas são efectuadas sem critérios de maturidade explícitos, não sendo estes suficientes para detalhar os sete níveis de maturidade, nem quais os critérios que são passiveis de melhoria e onde é necessário apostar para tornar os serviços mais integráveis.

Esta é uma solução que permite aos entrevistados tomarem decisões mais fundamentadas, uma vez que já tem os critérios definidos sem uma ambiguidade tão relevante.

Como resultado esperado, pretendemos conseguir demonstrar qual é o nível de maturidade dos serviços para cada dimensão das organizações que adoptaram SOA, pretendendo assim elaborar com sucesso a nossa avaliação.

Também o facto de maior parte dos modelos de maturidade serem elaborados por entidades privadas por vezes leva a interesses económicos em que lhes cabe só a eles saberem que critérios definiram para os modelos.

Para entender qual a utilidade do nosso artefacto na realidade, demonstramos o seu uso em organizações como CMM, Instituto Público, Link, Ordem dos Advogados focando a análise da demonstração apenas numa, a Ordem dos Advogados. Todos os critérios usados para a avaliação foram validados anteriormente, por 5 pessoas especialistas em IT.

Para avaliar qual a utilidade do artefacto proposta e os seus resultados, usámos:

- Entrevistas a profissionais

- Demonstração da utilidade do método em 4 organizações portuguesas
- A Moody and Shanks Quality framework para avaliar a qualidade do método produzido
- Os quatro princípios de Österle et al para validar a nossa solução

Estas avaliações mostraram que a nossa proposta é útil e genérica para poder ser aplicada a todos os tipos de organizações. Todos os que tiveram contacto com a nossa proposta, e os seus resultados, mostraram uma boa aceitação desta abordagem para resolver o problema encontrado.

Nas próximas secções iremos detalhar todas as nossas conclusões através da apresentação das lições aprendidas (Secção 9.1), as limitações identificadas (Secção 9.2), as contribuições da nossa proposta, incluindo propostas de melhoria (Secção 9.3) e ainda o trabalho futuro.

9.1. Lições Aprendidas

Durante esta investigação houve muitos aspetos que adquirimos que vale a pena referir como conclusão desta dissertação.

Em relação ao trabalho relacionado, observamos que há um aumento significativo de aplicações, nos últimos anos, em relação a integrações de serviços em todas as organizações, de qualquer tipo de negócio, devido a quererem tirar maior rentabilidade dos SI que adquiriram e manter os seus dados coerentes. As organizações querem saber se o ROI é o esperado aquando a aquisição do SI, necessitando para isso de avaliar a maturidade da sua própria arquitetura, ou seja, elaborar um *self-assessment,* que se enquadra na nossa proposta de dissertação.

Em relação à formulação da proposta, tal como observamos no trabalho relacionado, não existe qualquer consenso em relação às respostas dadas pelos entrevistados num processo de auditoria, pois os critérios de resposta não estão devidamente detalhados, são insuficientes, o que leva a uma avaliação com uma maior ambiguidade. Porém foi interessante e enriquecedor entrevistar diferentes chefes de IT, percebendo as suas motivações e os princípios que seguem para as suas organizações.

Em relação à fase da demonstração do nosso método de avaliação, as lições aprendidas revelam que a precisão do método depende das preferências e opiniões do entrevistado; É necessário uma preparação por parte do entrevistado para que as respostas sejam coerentes; o método produz resultados muito interessantes; é difícil as organizações começaram a pensar em evoluir o seu SOA, até por questões orçamentais e estruturais.

Por último, a lição que aprendemos é que os critérios poderiam ter um detalhe com uma fundamentação melhor, sendo o ideal entrevistar vários *stakeholders,* dos diferentes departamentos da mesma organização. Assim conseguiriamos ter uma avaliação ainda mais rigorosa e os departamentos saberiam que estratégia poderiam definir para evoluir a maturidade dos serviços da sua dimensão.

9.2. Limitações

Em relação às limitações da nossa investigação, estas estão diretamente relacionadas com a nossa pesquisa bibliográfica e com a nossa demonstração.

As limitações da nossa pesquisa bibliográfica está implicitamente relacionadas com os métodos de avaliação dos modelos de maturidade para SOA, não obtendo nenhum resultado de um *assessement* efectuado a outras organizações com base noutros modelos de maturidade, apenas nos baseamos em como a avaliação era efectuada.

Na demonstração, só efetuamos o *assessment* através do nosso método de avaliação, por limitações de tempo, visto que era extenso o nosso questionário, podendo demorar 60 minutos a ser respondido e não achamos correto constrangirmos mais os entrevistados no seu posto de trabalho propondo resolver o método de avaliação do OSIMM, com a finalidade de comparar os resultados de maturidade.

Notamos que existe receio por parte das organizações privadas ou públicas portuguesas, em participar em investigações deste género mesmo que a confidencialidade seja assegurada, até porque neste caso existe uma pessoa que está a analisar a arquitetura da organização.

9.3. Contribuições

As contribuições da nossa investigação estão relacionadas com o facto de termos conseguido identificar um problema real das organizações (Capítulo 2), analisá-lo (Capítulo 3), criar e desenvolver uma proposta (Capítulos 4 e 5), demonstrá-la e avaliá-la numa organização real (Capítulos 6 e 7).

A nossa proposta, um método eficaz para avaliar o nível de maturidade de uma arquitetura orientada a serviços, respondeu de uma forma concreta ao nosso problema de investigação, conseguindo avaliar a maturidade de uma organização, e elaborar um *recommended process* consoante a análise dos resultados através de gráficos com o método proposto.

Isto permitiu-nos ajudar o decisor a tomar uma decisão para o futuro da sua organização, através de um *roadmap*, identificando os aspetos concretos que necessita melhorar, de forma a conseguir atingir o nível de maturidade desejado.

Além disto, durante a formulação da proposta criámos um catálogo de critérios para avaliar o nível de maturidade de uma arquitetura orientada a serviços para uma organização. A própria proposta em si é simples e tem passos bastante bem definidos, que podem ser usados para avaliar apenas uma dimensão especifica ou no global. A utilização generalizada deste método ajudará as organizações a conhecer melhor o que existe no mercado e também a conhecer as falhas existentes nos serviços de organização.

Acreditamos também que a nossa proposta ajudará as organizações a poupar financeiramente em relação a auditorias externas, conseguindo assim com este método fazer uma *self-assessment*.

9.4. Trabalho futuro

Em relação ao trabalho futuro a realizar sobre a nossa proposta existem alguns aspetos que a podem tornar mais generalizável, mais fácil de usar e compreender. O trabalho futuro tem muito a ver com as limitações identificadas anteriormente.

Em primeiro lugar, é necessário fazer mais demonstrações em todo o tipo de organizações (públicas, privadas, de grande e pequena dimensões) para provar de uma forma definitiva que a nossa proposta produz resultados interessantes e úteis que possam ser utilizados para melhorar a maturidade dos serviços das organizações. Para isto é necessário conseguir ultrapassar a barreira criada pelas organizações a investigações deste género.

Em segundo lugar, para tornar a nossa proposta mais fácil de aplicar, seria bom criar um software que produzisse um *report* automático consoante os critérios escolhidos, de forma a tornar a avaliação de maturidade mais intuitiva e atraente para efetuar análises deste género.

Criar também um *recommended process* automático como um guia interativo e explicativo com outputs do método de avaliação, para as organizações saberem qual o *roadmap* que necessitam de definir a fim de ir de encontro à estratégia da mesma.

Bibliografia

Becker, P. J., & Knackstedt , D. R. (2009). *Developing Maturity Models for IT Management – A Procedure Model and its Application.* Business & Information Systems Engineering.

The Enterprise IT Performance Maturity Model. (2014). Compuware Corporation.

Alan, H., Salvatore, M., Jinsoo, P., & Sudha, R. (2004). *Design Science in Information Systems Research.: MIS Quarterly* (Vol. 28). Society for Information Management and The Management Information Systems Research Center.

Alonso, G. (2003). *Web Services: concepts, architectures and applications.*

Arsanjani, A., & Holley, K. (2005). *Increase flexibility with the Service Integration Maturity Model (SIMM).* . IBM developerWorks.

Barron, R., Gauci, J., Krishnamurthy, J., Laird, R., Xu, P., & \. (2013). *SOA Policy, Service Gateway, and SLA Management.* (IBM, Ed.) IBM.

Belton, V., & Stewart, T. J. (2002). *Multiple Criteria Decision Analysis - An Integrated Approach.* Kluwer Academic Publishers.

Bieberstein, N. (2005). *Impact of service-oriented architecture on enterprise systems, organizational structures and individuals.* IBM Systems Journal.

Britton, C., & Bye, P. (2004). *IT architectures and middleware: strategies for building large, integrated systems.* Addison-Wesley.

Bryman, A. (2012). *Social Research Methods.* Oxford University Press.

Computer Security Institute. (2011). *2010/2011 Computer Crime and Security Survey.*

Creswell, J. (2009). *Research Design - Qualitative, Quantitative, and Mixed Methods Approaches* (3rd ed.). SAGE.

de Leusse, P. (2009). *A Governance Model for SOA.* Los Angels : IEEE.

Dietz, J. (2006). *Enterprise Ontology: Theory and Methodology.* Springer.

Dyer, J. S. (2005). MAUT - Multiattribute Utility and Value Theories. In J. Figueira, S. Greco, & M. Ehrgoot, *Multiple Criteria Decision Analysis: State of the Art Surveys.* (pp. 265-295). New York: Springer.

Dyer, J. S., & Sarin, R. K. (n.d.). Measurable Multiattribute Value Functions. *Operation Research, 24*(4), 810-822.

Ekelhart, A., Fenz, S., & Neubauer, T. (2009). Ontology-based Decision Support for Information Security Risk Management. *Fourth International Conference on Systems*, (pp. 80-85).

Erl, T. (2005). *Service-Oriented Architecture: Concepts, technology, and design.* The Prentice Hall Service-Oriented Computing Series.

Feuerlicht, G., & Govardhan, S. (2007). *SOA:Trends and Directions.* Sydney: Department of Information Technologies .

Figueira, J., Greco, S., & Ehrgott, M. (2005). *Multiple Criteria Decision Analysis State of the Art Surveys.* Springer.

Group, T. O. (2007). *The Open Group Architecture Framework (TOGAF).*

Group, T. O. (2011). *Service Integration Maturity Model (OSIMM)* (Vol. Version 2).

Hensle, B., & Deb, M. (2013). *Soa Maturity Model- Gudiing Accelarting SOA Sucess.* Oracle Corporation.

Hevner, A., March, S., Park, J., & Ram, S. (2004). *Design Science in Information Systems Research.* MIS Quarterly, Vol.28, No.1, pp.75-105, Society for Information Management and The Management Information Systems Research Center.

HKSAR, G. o. (2008). *An Overview of Information Security Standards.* The Government of the Honk Kong Special Administrative Region, Hong Kong.

Holland, C. P., & Light, B. (2001). *A stage maturity model for enterprise resource planning systems use.* SIGMIS Database.

Holley , K. E., & Arsanjani, A. (n.d.). *100 SOA Questions: Asked and Answered.* (P. Education, Ed.) Greg Wiegand.

Inaganti , S., & Aravamudan , S. (2007). *Soa Maturity Model.* bptrends.

Irani, M. Z., & O'Keefe, M. R. (2001). *ERP and application integration: exploratory survey. Business Process Management.* Themistocleous.

ISACA. (2009). *Cloud Computing: Business Benefits With Security, Governance and Assurance Perspectives.* Rolling Meadows , USA: ISACA.

ISACA. (2012). *Security Considerations for Cloud Computing.*

Jane, W., & Richard, W. T. (2002, June). Analyzing the Past to Prepare for the Future: Writing a Literature Review. *MIS Quarterly, 26*(2), 13-23.

Kankanhalli, A., Teo, H.-H., Tan, B. C., & Wei, K. w.-K. (2003). An Integrative study of information systems security effectiveness. *International Journal of Information Management, 23*, 139-154.

Lagner, C., & Heutschi, R. (2007). *SOA Adoption in Practice - Findings from Early SOA implementations.* European Conference on Information Systems ECIS .

Lam, W. (2005). *Investigating success factors in enterprise application integration: a case-driven analysis.* European Journal of Information Systems.

Legner, C., & Heutschi, R. (2007). *SOA adoption in practice: findings from early SOA implementations.* . St. Gallen, Switzerland: University of St. Gallen.: Fifteenth European Conference on Information Systems.

Ling, A. P., & Masao, M. (2011). *Selection of model in developing Information Security criteria on Smart Grid Security System.* Busan, Korea: Ninth IEEE International Symposium on Parallel and Distributed Processing with Applications Workshops.

M., Z. I. (2001). *ERP and application integration: exploratory survey. Business Process Management.* Themistocleous.

Mahajan, R. (2006). *SOA and the enterprise: lessons from the city, in IEEE International Conference on Web Services (ICWS'06).* Chicago, USA.: IEEE Computer Society.

Marks, E. A. (2008). *Service-Oriented Architechure Governance for the Services Driven Enterprise.* Wiley.

Marks, E. A., & Bell, M. (2006). *Service Oriented Architecture (SOA): A Planning and Implementation Guide for Business and Technology.*

Microsoft Corporation. (2014). *Office 365 mapping of CSA Security, Compliance, and Privacy requirements.* Microsoft Corporation.

Moody, D., & Shanks, G. (2003). Improving the quality of data models: Empirical validation of a quality management framework. *Information Systems, 28*, 619-650.

MUNSHI, J. (2014). A METHOD FOR CONSTRUCTING LIKERT SCALES.

Norman, A. A., & Yasin, N. M. (2010, September). An Analysis of Information Systems Security Management (ISSM): The Hierarchical Organizations vs. Emergent Organization. *International Journal of Digital Society (IJDS),, 1*(3).

Osterle, H. (2011). *Memorandum on Design-Oriented Information Systems Research* (Vol. 20).

Osterle, H., Becker, J., Frank, U., Hess, T., Karagiannis, D., Krcmar, H., et al. (2011). Memorandum on Design-Oriented Information Systems Research. *20*, 7-10.

Pai, Y. (n.d.). (Soablureprint, Producer) Retrieved from http://www.soablueprint.com/maturity_models

Papazoglou, M. P. (2007). *Web Services: principles and technology.* Prentice Hall.

Paré, P. (2004). *Investigating information systems with positivist case study research.* Communications of the Association for Information Systems.

Peffers, K., Tuunanen, T., & Rothenberge, M. A. (2008). *A Design Science Research Methodology for Information Systems Research.* Journal of Management Information Systems.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2008, Winter). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems, 24*(3), pp. 45-78.

Peter, C., & Sue, H. (1997). *Information, Systems and Information Systems - making sense of the field.*

Pezzini , M., & Medrano, R. (2011). *SOA Governance.* Pearson Education.

Pries-Heje, J., Baskerville, R., & Venable, J. (2004). *Strategies for Design Science Research Evaluation.* 16th European Conference on Information Systems (ECIS), pp. 255-266.

Röglinger, J. P. (2011). *What makes a useful maturity model? A framework of general design principles for maturity models and its demonstration in business process managment.*

Saaty, T. L. (1980). *The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation.* . New York: McGraw-Hill. .

Siponen, M. T. (2003). *Information Security Management Standards: Problems and Solutions.* Adelaide, South Australia: 7th Pacific Asia Conference on Information Systems .

Spotter, D. (2007). *Soa Maturity Assessment Survey.* CBDI.

Standard, I. 2. (2013). Retrieved from ISO 27001 Basics: http://www.iso27001standard.com/en/what-is-iso-27001

Susanto, H., Almunawar, M. N., Tuan, Y. C., Aksoy, M. S., & Syam, W. P. (2011, December). Integrated Solution Modeling Software: A New Paradigm on Information Security Review and Assessment. *International Journal of Science and Advanced Technology, 1*(10), 90-99.

Tardugno , A. F., DiPasquale, R. T., & Matthews, R. (2000). *IT Services: Costs, Metrics, Benchmarking, and Marketing.* Prentice Hall.

Tipton, H. F., & Nozak, M. K. (2012). *Information Security Management Handbook* (Vol. 6). CRC Press.

W3C. (2007). *SOAP: messaging framework.*

Walker, M. (n.d.). Retrieved from http://www.mikethearchitect.com/2009/09/soa-maturity-models.html

Wu, X., Fu, Y., & Wang, J. (2009). *Information systems security risk assessment on improved fuzzy AHP* . Sanya, China: ISECS International Colloquium on Computing, Communication, Control, and Management .

Yang, X., Zhang, S., Liu, F., & Jian, G. (2010). *A Study on Security Evaluation for Information Systems Based on Grey Clustering.* San Diego, California: IEEE.

Anexos

Anexo A – Bibliografia utilizada para a criação dos critérios das AQ do modelo OSIMM

Referências dos xx artigos usados na construção da lista de critérios das AQ do modelo OSIMM (Secção xx) durante a revisão da literatura (algumas estão incluídas nas referências):

- Skelta, Invesys (2012). *ROI on BPM: The Whys and Wherefores of a BPM Investment .* http://www.skelta.com/Industries/ROI-on-BPM.pdf
- CISCO IOS, (2005). *Service level Managment: best Pratices White Paper*
- Solomon, Omokehinde. *Supplier-Customer Relationship in relation to costumer satisfaction*
- PA, (2008). *SOA Design, Development and Deployment Methods, Processes and Tools*
- Naybour, Paul (2010). *What is a Project Management Framework?*
- Byatt.Gareth, Hamilton.Gary, Hodgkinson.Jeff. *What Makes a Good Project KPI Framework?*
- Accenture (2001-2002), *UDDI Frenquently Asked Quesions (FAQ)*
- Scott W.Ambler (2005), *Types of Reuse In Information Tecnhology*
- Simon St. Laurent (1998), Why XML
- Chappel David (2008), Next-Generation Grid-Enabled SOA: Not Your MOM'S Bus, Soa Magazine Issue XV
- Thomas Erl (2005), A Field Guide to Integrating XML and Web Services
- Erick A. Marks (2006), Michael Bell, Service Oriented Architecture (SOA): A Planning and Implementation Guide for Business and Technology
- Frank Niessink and Hans van Vliet (1998), Towards Mature IT Services
- Sourav Mazumder (2006) , SOA: A Prespective on Implementation Risks, Infosys
- Jun Wang (2009), A dynamic data integration model based on SOA
- Eskil Swende (2007), Information Model - SOA in a Business Perspective, published in TDAN.com
- Shaker H. Ali El-Sappagh, Abdeltawab M. Ahmed Hendawi, Ali Hamed El Bastawissy (2011), A proposed model for data warehouse ETL processes
- Masoud Niknam,Jivka Ovtcharova, Towards Higher Configuration Management Maturity

Anexo B – Critérios de resposta para as AQ do modelo OSIMM

Business Dimension

How are identified the major business drivers for this initiative?
There are no business drivers
There are some business drivers documented and explicits in business process integration
Organization's business drivers are documented as cross-organizational business objectives.
The business drivers are identified throughout organization internal processes
The business drivers are identified throughout organization external processes
The business drivers are identified throughout organization internal and external processes
The bussiness drivers are adapted to frequent business process changes
What is the business vision and goals, and how are these related to what IT is currently doing?
There is no vision of a future SOA situation;
Some IT people have thought about the future SOA situation;
There is a vision of a future SOA situation within some business critical projects;
Only the board directors have a vision of future SOA situation
Most IT staff of the organization share a vision of future SOA situation;
There is one clear vision about future SOA situation shared throughout the organization
There is one clear vision about future SOA situation shared throughout the organization and on a inter-organizational level;
Is your current Business Process Architecture formally defined, documented, and governed?
Not have formal Business Process Architecture
Business Process Architecture is not decomposed
The business processes are formally defined, documented and governed
Use design tools to define internal business processes of the organization
The current business processes are possible to change at design time
The current business processes are possible change at design time and run time
The current business process architecture provide a real-time summary of business activities to operation managers at run time

Is your Business Process Architecture complete and up-to-date?
We don't have a Business Process Architecture
The current business processes are not described in business services;
Business processes are described in business services within some business critical projects;
Experimentation is done to describe business processes in business services with best Business Process Architecture
Describing business processes in business services is standardized throughout the organization;
Describing business processes in business services is standardized throughout the organization and follows one framework for enteprise architectures
Business Process Architecture supports dynamically reconfigurable business services and infrastructure services in real time
How are metrics for return-of-investment measured in Business Process Management (BPM)?
There are no defined metrics for ROI in BPM
The are metrics that only take into account customer satisfaction
There are metrics that treat process cycle time and volume
There are metrics to analyze results compliance
There are metrics to rate process error
There are metrics to quantify the response time
There metrics being used in BPM to reduce risks
How agile are your current business processes?
We don't have business processes
We have business processes but not decomposed
We have business process decomposed but only in pilot project(s)
We have only business process decomposed in some business critical project
The decomposition of business process is being standardized throughout the organization
The decomposition of business process throughout the organization is completely standardized
The decomposition of business process throughout the organization is completely standardized and also within business externals collaborations
What is the current cost model?
We don't have a cost model
We have a cost model defined but it has not been applied
Cost model is only being applied to business processes with a higher risk
Cost model is being applied to some departments of the organization
Cost model is transversal to the organization, it splits the costs with all business unit produced or executed by a specific service or product
We use the Activity Process Based costing (ABC) cost model
We use the TDABC cost model based on process and activities execution time
Who owns the portfolio of processes, applications, and services?
There are no portfolio of processes, applications and services
Portfolio of processes, applications and services are owned by IT architects at pilot project level
Portfolio of processes, applications and services are owned by different IT architects within some business critical projects;

Portfolio of processes, applications and services are owned by the central IT departments of business units;
Portfolio of processes, applications and services are owned by business units and offered only internally;
Portfolio of processes, applications and services are owned by business units and offered externally;
Portfolio of processes, applications and services are owned by business units and offered internally and externally;
Who owns application services in your organization?
There are no application services
Application services are owned by IT architects at pilot project level
Application services are owned by different IT architects within some business critical projects;
Application services are owned by the central IT departments of business units;
Application services are owned by business units and offered only internally;
Application services are owned by business units and offered externally;
Application services are owned by business units and offered internally and externally;
Do you have a cost model to charge service consumers for the use of the service?
We don't have cost model
There is a cost model but it is not applied to the use of service consumers
Our cost model is being applied only in pilot project(s)
Our cost model focuses in critical process stages, necessary to the completion of services
Our cost model pays all process stages necessary to completion of services
Our cost model calculates the cost of all internal service stages
Our cost model calculates in real time the cost of all internal and external process stages
How do you currently define the total cost of ownership (including software, hardware, and future maintenance)?
We don't calculate total cost of ownership
We calculate the total cost without basing in any model
We calculate the total cost of ownership but only in some aspects (software,hardware,future maintenence)
The total cost of ownership calculates all costs related with the organization
Our cost model is based on the necessary effort of each activity/process on that aspect
The cost of ownership is dynamically calculated based on every aspect related to the necessary effort for each activity/process on that aspect in internal services
The cost of ownership is dynamically calculated based on every aspect related to the necessary effort for each activity/process on that aspect in internal and external services

Organization & Governance Dimension

What types of skills are common in your IT staff?
Nobody of IT staff has SOA skills
Some people of IT staff have SOA skills
In some business critical project IT staff have SOA skills
In most parts of the organization people of IT staff have SOA skills

IT staff throughout the organization have SOA skills for internal application services
IT staff throughout the organization have SOA skills for external application services
IT staff throughout the organization have SOA skills for internal and external application services
How does IT governance relate to your SOA?
Do not exist any SOA strategy defined
The IT strategy addresses the use of individual software applications;
The IT strategy addresses the potential of application integration with application services;
The IT strategy addresses application integration with application services for different software applications
The IT strategy addresses application integration with application services throughout the organization;
Application integration with internal application services is fully institutionalized in IT strategy
Application integration with internal and external application services is fully institutionalized in IT strategy;
How is the IT governance related or aligned with the SOA, enterprise architecture, and the organization's governance?
There are no application services
There are some application services in critical business processes
The value of service and SOA governance has been recognized but has not been holistically adopted by the enterprise.
SOA governance has been established but has not been adopted holistically by the enterprise.
A formal enterprise-wide SOA strategy and vision has been defined, published, and agreed by the business units across the organization.
The use of SOA and shared services are an accepted element of the organization's strategy, business, and IT models.
SOA governance is part of the organizational culture.
Do SOA governance processes exist, are they documented, and, if so, are they used for services at design time and run time?
There is no SOA Governance processes documented
Some SOA Governance processes are documented if are critical
The organization documented all SOA Governance processes
The organization use design tools to define SOA Governance processes
The services of SOA Governance processes is possible to change at design time
The services of SOA Governance processes is possible use at design time and run time
The SOA Governance processes provide a real-time summary of business activities to operation manager at run time
Is the interaction between organizations involved in the SOA process defined with clear roles and responsibilities?
There is no roles and responsabilities defined
A formalized SOA strategy exits between one or more organizations.
Exist SLA defined in critical SOA business process between organizations
Exist SLA defined in some SOA process between organizations
Exist SLA defined in all SOA process between organizations
The SOA Policy and service Gateway are well inter-organizations
The SLA Management is controlled by external organization
What are the governance functionalities and responsibilities?
Do not exist governance in organization
There are some rules defined in some business processes but only in pilot project(s)

Exist governance with rules and authorities but only on critical process in organization
The rules and authorities are fully integrated and applied throughout processes of the organization
The IT governance has been adopted across the organization and is empowered to manage SOA services
The organization has SOA metrics and performance indicators well defined
SOA metrics are automatically gathered and they are an important key business decisions
How would you describe your IT cost model?
There is no IT cost model
We have one IT cost model but is not yet applied
We have a pilot(s) project(s) or proof of concept(s) of IT cost model with a narrow scope;
Our IT cost model is applied to some business critical projects;
Our IT cost model is cross-cut in the organization
Our IT cost model is calculated based on effort of (activity/process/based costing)
Our IT cost model is calculated base on execution time of process and activities
What type of SOA training is available in your IT organization?
There is no SOA training;
Knowledge and experience about SOA is shared ad hoc;
Knowledge and experience about SOA is shared as best practices between projects;
SOA training is centralized at enterprise level and offered in most parts of the organization;
SOA training is institutionalized and fully integrated into IT training;
SOA training is offered by external organizations;
SOA training is offered within and between organizations;
What is the relationship between the organization's development team and the infrastructure team?
There is no cooperation between organization's development team and infrastructure team to support business process;
Organization's development team and infrastructure team jointly experiment in a pilot project to support business process;
There is organization's development team and infrastructure team in some business critical projects that orchestrate business processes with application services;
Organization's development team and infrastructure team are integrating at business unit level to support business process throughout the organization;
Organization's development team and infrastructure team are fully integrated to support central application services throughout the organization;
Organization's development team and infrastructure team are fully integrated to support business process orchestration throughout the organization with internal application services;
Organization's development team and infrastructure team are fully integrated to support business process orchestration throughout the organization with internal and external application services;
What SOA and governance authorities exist?
There are no SOA and governance authorities
Policies are developed ad hoc within SOA pilot projects;
There are policies to support the governance of services within some business critical projects;
SOA governance is centralized and applied in most parts of the organization;
SOA governance is fully integrated within IT governance and applied throughout the organization;
SOA governance for external services is fully integrated within IT governance;

SOA governance for internal and external services is fully integrated within IT governance;
Do the organization's SOA solutions cross organizational boundaries? Internally? Externally between business partners?
We don't have any type of SOA Solutions in organization
We are documenting the internal and external business processes between them business partners
Only the internal and external process of business critcial processes are defined
The SOA are in implementation phase and have many applications in SILOS
The SOA implementation on organization enable a better Knowledge management (KM) only for internal processes
The SOA implementation on organization enable a better Knowledge management (KM) for internal and external processes
The SOA Implementation enabe to have evaluation and grow the ROI through organization's KPI'S

Method Dimension

What are the current application or systems requirements elicitations and requirements management practices?
We don't have any requirements management pratices
We have guidelines that describe the collection techniques and the types of requirements to be collected.
We have a pilot project for management requirements of current applications
We have metrics to identify the "as-is" of the problem's cause(s)
We have a log that enables tracing the current critical business processes
We have a log that enable tracing all current critical business processes
We have a communication system that enable the stakeholders to know what requirements has been completed
What design methodologies and best practices are you currently adopting?
We don't use SOA design methodologies
The SOA methods and practices are limited to the IT development teams
The SOA methods have been enhanced to address the creation, implementation and deployment of services
SOA methods and practices have been implemented in some departments of the enterprise.
SOA methods and practices have been implemented across the enterprise.
Formal methods are used to create and manage both internal and external partners based services
Formal methods leverage architectural constructs and assets for supporting virtualization and dynamic services and business process modeling.
Do you practice any SOA design techniques?
We don't use any formal SOA design techniques.
We use UML diagrams that emphasizes the static structure of the system using objects, attributes, operations and relationships.
We use UML diagrams that emphasizes the dynamic behavior of the system by showing collaborations among objects and changes to the internal states of objects.
We use SOA design techiques only for critical business processes
All business processes support WSDL by XML format
We use SOA Design techinques to execution languages like BPEL
All business processes already have WS-BPEL that is standard executable language for specifying actions within business processes with web services.

What design tools are in practice today?
We don't use design tools
We have a design tool but has not been applied yet
We have a pilot project for business processes modelation
We use design tools only made for critical business processes
We use desing tools across all business processes in the organization
The modulation of business processes is already an execution language like BPEL
All design methodology of our organization is developed by an external entity
What is the current practice for service development and management?
We don't use service development neither management methodology
We have a service development and management methodology be we are not using them
We are experimenting the service(s) development and management methodology in pilot(s) project(s)
We use one framework of service development that allow us to design, analyze and plan the development of services like ITIL
We use a recognized methodology for the creation, development, deployment, and management of current services
We use Business Process Managment (BPM) to manage our services
We use the best methods to facilitate consistent adoption of SOA and virtualization technologies: ESB and registry.
What is your current project management framework?
We don't have project management framework
We have a project management framework but we don't use them
We have business process with Project management Framework (PMF) but only in pilot project(s)
Only the critical business processes are managed by Project management Framework (PMF)
All processes are standardized to use Project management Framework (PMF)
The Project management Framework (PMF) is only used in some departments of our organization
The Project management Framework (PMF) is used cross-organizational
How is IT project management organized?
We don't have de IT project management organized
The IT project management have all KPI'S
The IT project management have very explicit the development phases
The IT project management have business development
The IT project management have contract negotiation
The IT project management have migration or implementation phase
The IT project management have the Close Out phase after doing the appropriate tests
What are your organization's current QA processes?
We don' have Quality Assurance (QA) in our organization
In our organization we have metrics that implement Processes and Product Quality Assurance (PPQA) but only in pilot project(s)
In our organization we have metrics that implement PPQA but only on some critical business process
In our organization we have metrics that implement PPQA in some departments in our organizaiton

The PPQA are cross-organizaitonal according to standard ISO9000
In the service development there is QA process that ensure specific requirements according to framework
We use QA processes in virtualized services to facilitate service performance
Do you have an active community that works to evolve your SOA methods and practices?
We don't have active community
Our active community focus on SOA methods but pratices have not been formalized across teams
We have one internal active community that only develops SOA methods and pratices
Our active community only work on SOA methods and pratices of critical business processes
A recognized community is empowered to manage, train, and update the enterprise SOA methods and practices
We have one repository that help the active community to develop SOA methods and pratices
SOA methods and practices are developed by external active community
Has your organization developed a repository for best practices and asset re-use?
Our organizaton did not develop a repository for best pratices and asset re-use
We have one internal repository but is not yet used in the organization
Our organization has one UDDI and has been tested only in pilot project(s)
Our organization has one UDDI but only for critical business processes
Our organization has one UDDI for all process and thus we have a global registry to re-use them again
Our organizaiton with UDDI can gain e-business partnerships to use dynamically discover
We use virtualization of the IT service operations methods to facilitate repository service performance

Application Dimension

What is your current application development style?
We don't use any application development style
Application architectures and topologies are monolithic with minimal separation of concerns between architectural layers or application tiers.
In our organization all applications are split by physical and logical concepts in presentation
Our services components were developed using SOA patterns
We use several applications development style according to the artifact
Our architeture application using SOA patterns is decoupled from infrastructure components
Our application supports dynamically reconfigurable
How common is re-use in your organization?
The re-use of services is nonexistent in our organization
The re-use of services is low in our organization
We only apply re-use in critical business processes
We have one list of all tested services and that can be re-used
We re-use services only if they are tested and guarantee the best performance
We re-use services only if they are in global registry
If the service are in UDDI automatically is discover

What types of re-use do you engage in and how is re-usability measured?
We don't use any type of re-use
We don't use QA metrics in our processes and services
We use QA metrics only in critical services or processes that are core of our organization
We re-use code - reuse of source code within sections of an application and potentially across multiple applications.
We re-use modules - The use of pre-built, fully encapsulated modules, components, services, or code libraries in the development of your application.
We use frameworks that re-use collections of classes that together implement the basic functionality of a common technical or business domain.
We re-use patterns that can have multiple levels - analysis, design, and architecture are the most common
How are your organization's applications/systems integrated?
Application architectures and topologies are monolithic and lack integration between other systems across the organization
Exists only total integration between the critical business applications/systems
Applications use minimal integration between other systems
Service integration is achieved using an ESB in some but not all business units.
We use support application and processes patterns to achive sharing services
The integration is dynamically reconfigurable with internal services
The integration across the organization is made externally between business partners.
What types of languages does your organization use?
We only use one language in our organization
We use low level language like machine code or assembly language
We use Object-oriented programming like c#, java
Our organization use web OO programming
Our organization use concepts like distributed computing that support reusability or modular programming
Our organization use BPM tecnhology to develop deliver real-time solutions
Our organization use BI and BAM to monitor performance of business processes
What types of integration technologies has your organization employed?
We don't use integration technologies in our organization
We have integration technologies but we are not applying them
Service components of application architectures employ SOA patterns such as separation of concerns between logical and physical layers of the presentation and business logic.
Integration is usually implemented using point-to-point techniques.
The use of SOA-enabling technologies such as an ESB is consistent across the enterprise.
Extensive use of ESB architecture patterns to support Business Process Management (BPM).
We have Business Activity Monitoring (BAM) to monitoring of business activities, as those activities are implemented in computer systems.
How is business logic represented within your organization's applications?
Business logic has not been decomposed;
We only have documented how is represented the business logic into application using design tools
Decomposition of business logic into basic applications is done in pilot project(s);
Decomposition of business logic into basic applications is done within some business critical projects;

Decomposition of business logic into basic applications is being standardized throughout the organization;
Decomposition of business logic into basic applications is performed consistently in a standard way throughout the organization;
Decomposition of business logic into basic applications is performed consistently in a standard way throughout the organization and within business collaborations;
How reliable are your organization's business-critical applications?
We don't have identified which are critcial business applications
We don't use reliable process in our business-critical applications
We only virtualize the critical processes of our core applications
We virtualize some applications of organization's departments to share resources and to improve the performance
We use activity monitoring to control the processes across our applications
Virtualization is a key element of the IT service operations methods and is used to facilitate service performance
Our virtualization is made by an external entity that cooperate with our organization
How widely is XML used in your organization? How sophisticated is its use?
We don't use XML language in our organization
All documents produced use one structure specified by DTD
The XML is use to structure templates and output models
We only use XML to document management and versions control
We implement XML language for most used services to a better and simple integration like data sharing
We use XML language for all applications in our organization
All of our services are in XML with defined structure that make it possible to put in global registry to be shared, transacted, discovered
What is the rate of change and required time-to-market of your current applications?
We don't have rate of change and time to market in our applications
We don't have applications to upload in time-to-market
We only have defined metrics to rate of change and required time-to-market of our pilot projects
We use metrics to reduce rate of change and required time-to-market only at core applications of our organization
We use metrics to reduce rate of change and required time-to-market for all applications in our organization
The virtualization of our application alows to reduce rate of change and required time-to-market in our organization
The virtualization of our applications is made by an external entity that allows to reduce rate of change and required time-to-market in our organization
Are SOA-enabling technologies, such as ESB, shared data environment, or registry, being used?
The use of web services or other SOA constructs are not present.
We have all documentation to evolve to SOA technologys
We only have one SOA pilot project for core application of our organization
We have a service registry only for internal applications
We have a shared service registry for internal and external applications
ESB integration patterns are used to support application and process integration to achieve sharing of services.
We extensive use of ESB architecture patterns to support Business Process Management (BPM).

Architecture Dimension

How would you characterize your architectural topologies?
We don't have a architecture topologies
We have Monolithic Architecture
Our achitecture is constructed by layers that is it divide the logical infrastructure
We use SOA patterns in our architecture
SOA is implemented across organization with patterns and frameworks applied
We have implemented a dynamically re-configurable architecture
Our architecture is based on Grid-Enabled SOA with virtualized services
What type(s) of data repositories does your organization utilize?
We don't have data repositories in our organization
We use one local repository in our organization
We use only one information repository but with an external replication of our organization
We use multiples centrals repositories with version control (track record)
We use one RDBMS - Relational Database Management System
We use a workgroup repository that can be any J2EE Web Server repository
We use one EDMS - Electronic Document Managements System with document tracking, versiononing and search and retrevial
What is the standard communication style in your architecture?
We don't use any communication style on our architecture
We use Component-based architecture that focuses on the decomposition of the design into individual functional or logical components that expose well-defined communication interfaces containing methods, events, and properties
We use one Layered architectural style where each layer aggregates the responsibilities and abstractions of the layer directly beneath it.
We use a client-server architectural style describes distributed systems that involve a separate client and server system, and a connecting network.
We use Object-oriented Architectural Style that is a design paradigm based on the division of responsibilities for an application or system into individual reusable and self-sufficient objects, each containing the data and the behavior relevant to the object.
We use one message bus architectural style that use a software system that can receive and send messages using one or more communication channels, so that applications can interact without needing to know specific details about each other to communicate to principals process in our organization
We use a Service-Oriented Architecture within communication style that enables application functionality to be provided as a set of services, and the creation of applications that make use of software services.
How is integration achieved in your architecture?
We don't have integration in our architecture
We documented all services present in our architecture - DTDs and XSD Schemas
We use design tools to create independent services
We integrate core applications to separate logic and data
We create webservices for all applications

We use SOA patterns and methods to cummunicate between all applications
We use ESB that implement mutual communication between interacting software applications in a Service-Oriented Architecture (SOA)
What methods do you use to develop your architecture?
We don´t use any method to develop our organization
We use one conceptual model that is a model made of the composition of concepts
We first design the architecture according organizational requirements
We use frameworks like TOGAF to develop the architecture
We use agile methodology for enterprise architecture
We develop an architecture following ISO 15704-2000 - Requirements for enterprise-reference architectures and methodologies
We develop Standards and Architecture for eGovernment Applications (SAGA)
How mature are your services implementations?
We don't have metrics to verify how mature are our services
We have metrics only for critical business services to evaluate the maturity level of implemented services
Our methodologys and technics for service implementation still in pilot project phase
We have methodologys and technics to develop and maintenance of services implementation (Service Process Managment)
Through SLA's we can track and evaluate the needs of an IT service
We use an evaluation model with some criteria that allow us to evaluate how mature are our services implementations
We use a model that allow us to diagnose and make a recommendation report that contain the level of maturity that we have and the level that we want to achieve
How extensive is your SOA?
We have a Monolithic architecture
We have a well defined enterprise architecture
Is not possible to extend our SOA
We know what are the SOA Principles and guidelines to extend SOA
We are developing a service portfolio and a business service to extend SOA
We save an ESB that give us some of the key features for any SOA, namely Service Registry, Transformation Component, Routing mechanism, even adapters
We have business activity monitoring to monitor all business activities of our organization
What architectural principles define your approach?
There is no SOA reference architecture;
There is an informal contact about how to develop application services;
Best practices are shared between SOA projects;
A SOA reference architecture emerges at enterprise level that is used in most parts of the organization;
Central application services throughout the organization follow one standard SOA reference architecture;
Internal services follow an organizational SOA reference architecture for their development;

Internal and external services follow an inter-organizational SOA reference architecture for their development;
How extensive and sophisticated is your organization's use of frameworks in your architecture?
There is no framework to support our architecture
We have one framework defined but not applied in our organization
There is an experimental framework for our architecture
We use only one service framework that is centralized at enterprise level and used in most parts of the organization;
There are one or more frameworks supporting different projects in our architecture;
There is one inter-organizational service framework that support all internal services;
There is one inter-organizational service framework that support internal and external services;
How are architectural decisions made in your organization?
There is not architectural decisions related to services;
Architectural decisions about service development are made ad hoc;
Architectural decisions about service development are based on initial policies within projects;
Architectural decisions about service development in most parts of the organization are made at enterprise level based on policies;
Architectural decisions about service development throughout the organization are made at enterprise level;
Architectural decisions about service development for internal service development are made at inter-organizational level;
Architectural decisions about service development for internal and external service development are made at inter-organizational level;
Does your organization use reference architectures?
Our organization don't use references architectures
We use reference architecture but have not yet been applied
There is informal contact about how to develop application services;
Best practices are shared between SOA projects;
SOA reference architecture emerges at enterprise level and is used in most parts of the organization;
Central application services throughout the organization follow one standard SOA reference architecture;
Internal and external services follow an inter-organizational SOA reference architecture for their development;

Information Dimension

Is there a common data model across all applications?
There are no standards for interpretation of information exchanged between individual software applications;
We have one data model defined but not applied in our organization

Experimentation is done in achieving semantic interoperability between application services in a pilot project;
There are some standards in line to achieve semantic interoperability between application services of some business critical projects;
Standards for semantic interoperability between application services are proposed on enterprise level with common data model;
There is semantic operability of information exchanged between internal application services throughout the organization;
There is semantic operability of information exchanged between internal and external application services using common data model;
Are there independent data models for different applications?
We use independent data models between different applications
Information is shared across some applications using Extraction, Transformation, Load, Manipulate (ETLM)
Formal business information models have emerged, often accessed though XML schema style interfaces.
Business data models vocabularies are standardized within a business unit or process area.
Business data models vocabularies are standardized for use across the enterprise.
We use the same business data models that can easily be expanded and enhanced between different applications
Business data models can be expanded and enhanced as required to support new services, external partners, and business process reconfiguration.
Are mapping rules used to convert between different data models?
We don't use mapping rules to convert between different data models
We have defined rules but they are not applied in conversion/transformation between different data models
We applied mapping rules only in data models that support the core business processes
The mapping rules are made across differents data models in our organization
We use agile data method to migrate between differents data models
We use data mediation between different data models
We use SOA Dynamic Data Iintegration Model
Is there difficulty in moving data from one application to another? For all applications? For only some applications?
In our organization we don't migrate data
There are difficulties to migrate our data for all applications
The applications does not support yet data standards to migrate between applications
There are data migration only between critical application in our organization
We use one mediator to migrate our data between applications of our organization
All applications support data standards and its easier migrate data between storage types or formats
We have ETL systems commonly integrate data from multiple applications

Does your organization have a common data model, (or mappings between multiple data models)? How is this defined? By programming objects in APIs? By XSD schemas? By written documents? By other computer-based modeling tools? By other non-computer-based modeling tools?
We don´t have any data model and neither made any mapping
We have one data model but we are not applying them in our organization
We use common data models only for core business applications
The interoperability is achieved through XSD Schemas, written document and computer modelling tools
Business data is defined using semantic web constructs, or ontologies (e.g., UN/CEFACT Core Components, ISO 11179).
There is one mediator between types of schemes and the data models
The data models can be expanded through external entities collaboration or the use of API's
Are the data models in the form of Business Object Models, understandable to and owned by, the business, or as IT object models, understandable only to, and owned by, the IT teams?
The data models are not in form of Business Object Models
The data models are in form Business Object Models and understable as IT object models
Our organization use one data model in form of Business Object Models (BOM) developed by IT teams
The business object models are improved using collaboration teams
Data models in the BOM form are develop by external entities
There is one Document Object Model made to interopability between data models
The data models in the form of Business Object Models are related to the type of business
If there are mapping rules across different models, are these understandable to and maintained by the business or by IT staff? Are such mapping rules performed by the infrastructure?
We are not mapping the rules in different models
We have only mapping rules between different models for critical business processes and are maintainded by the business or by IT staff
We use data mapping with rules across differents models in pilot project
We have multiple business units that using meta-data relationships between differents models
The mapping between different models is made by using mediator that is mantained by IT staff
We have one registry with metadata that is used for contruct rules between differents models and it is performed by the infrasteucture
Business data vocabularies can easily be expanded and enhanced by IT staff and performed by the insfrastructure
Are the data models defined by a language that includes taxonomies, ontologies, or other high-level logical representations?
We don't have data models in organization
We have one data model defined, but not applied in our organization
The data models are defined by standard language like UML
Data models are developed using modelling language like BPMN

The data models are developed using a process execution language for core business processes like BPEL
The data models are developed follow the standard ISO 11179
The data models are developed using an entity framework
Do you maintain a global directory or database of data objects, with global identifiers? Or do you have mechanisms for mapping these objects between different databases/directories? Are these mechanisms electronic or manual? Are all such objects mapped, or is this done only for certain applications and sets of objects? Are these mappings undertaken automatically by the infrastructure?
Data objects are not stored;
Data objects are stored in an global directory or in database with global identifiers;
Data objects are stored in different databases or directories but with mapping mechanisms;
We have data objects of most application services throughout the organization are stored in one global registry;
We have electronic mechanisms to mapping the data objects with a global directory where there are storing service descriptions of all internal application services throughout the organization in a standard way;
There is one inter-organizational global registry storing service descriptions of internal and external application services in a standard way;
Data object are stored in a global registry with global identifiers and mapping automatically with our infrastructure
Do you have mechanisms for looking up global objects by searching on their characteristics?
Global objects cannot be discovered;
Global objects can be discovered in an experimental service registry or spreadsheet and they are promoted ad hoc by informal contact;
Global objectss and their characteristics can be discovered in one or more small-scale experimental registries;
Application services and their characteristics can be discovered in one service registry that covers application services in most parts of the organization;
There is an extensive service registry to discover and select global objects throughout the organization based on their characteristics in a standard way;
There is an extensive inter-organizational service registry to discover and select internal global objects based on their characteristics in a standard way;
There is an extensive inter-organizational service registry to discover and select internal and external global objects based on their characteristics in a standard way;
How is the transformation of data between applications achieved? Is an ESB used to perform the transformation? Is this achieved by bespoke adapters as required? Or via a comprehensive set of APIs? Or by calling a service?
We don't use data transformation between applications
Are only defined which are the standards of each application
All data use canonical format that is standard for XML
The data transformation is dynamic using API's on our applications
All applications already support data interoperability between them through invocation of a service
Our organization have an ESB that is a mediator on our application communication

The data transformation is dynamic using API's on our applications
Are there facilities for performing complex inference in order to map data defined in ontologies from one form to another? Does a master data service exist?
There is not any defined data mapping in ontologies
We have a pilot project to data mapping in ontologies
We have mapped all data in ontologies for critical business processes
We don't have any master service in our organization
We have a master data service to do a coreography across organization
The data mapping in ontologies is dynamic in our organization
The master data services dynamically do the coreography between our services
Does your organization have or are you developing a Business Information Model to standardize data and message formats and concepts across the enterprise?
We don't have any Business Information Model
We only have standards data, message formats and concepts that are documented
We have a Business Information Model only for critical appilcations
There is a document object model applied in all applications
The Business information model is developed by IT business team
The Business information model is developed by a partnership with IT Business team and external entity
The Business information model is develop from a Repository Type Information Model (RTIM)

Infrastructure Dimension

How are your current infrastructure usage guidelines?
There are no guidelines for current infrastructure
There are defined guidelines for current infrastructure but they are not yet applied
There are defined guidelines only for pilot projects
There are only guidelines applied in current infrastructure for critical business applications
We use a framework to develop our infrastructure following defined guidelines
We develop guidelines for our infrastructure inter-organizational
We define guidelines using ISO/IEC 23271:2003 Information technology - Common Language Infrastructure
How are your IT SLAs derived from the business SLAs?
The IT SLAs don't derived from the business SLAs
There is not individual metrics performance for business SLAs
We have a pilot project that IT SLAs derived from business SLAs

We only have business SLAs for critical business processes
The IT infrastructure supports the non-functional and operational requirements
Security polices are obligatorily respected, managed and applied by IT team
There is an IT service level manager that is responsible for negotiating, maintaining, and reporting against the SLA with your customers
Have you defined SLAs around quality-of-service? How is this monitored and measured?
We don't have SLAs defined for quality of service of the operational service infrastructure;
Some initial SLAs are defined for quality of service of the operational service infrastructure;
We only monitor and measure the quality-of-service in project pilot(s)
There are best practices for defining SLAs for quality of service of the operational service infrastructure;
Defining and monitoring SLAs about quality of service of the operational service infrastructure is standardized at enterprise level;
Defining and monitoring SLAs about quality of services of the operational service infrastructure is institutionalized at enterprise level;
Defining and monitoring SLA's about quality of service is institutionalized for the operational inter-organizational service infrastructure;
Have you defined any SLAs around security and privacy? How is this measured and monitored?
We don't have any SLA's defined for security and privacy
We have metrics for service level requirements (SLR) of security and privacy in our organization
We have defined SLAs only critical business processes
We have one pilot project to monitor SLA's for security and privacy
The monitoring is made by metrics for SLA's of security and privacy across organization
We use a method of standardizing the practices of IT services and design them
We use one Service level management hat focuses on aligning IT services with the needs of business like ITIL
What level of monitoring is in place today? What management tools are in place today?
We don't have monitoring in our organization
We only have documentation for managment tools that we will use in our organization
There is one pilot project that have management tools associated to critical business applications
We have monitoring and managment tools across the organizaton
We have common service platforms and containers including ESBs and business process management tools
We have Business Activity Monitoring to monitor services of our organization
We have one SOA Policy Management and Governance
What platforms are currently in use for integration?
We don't have any integration platform
There is a set of adaptors created for message communication between applications, using proprietary protocol like ftp or databases
We have Business Process Management that allow to make real-time intelligent work and resource management decisions
There is a data management with metadata information and versioning that ensures the data is kept consistent
We have message bus for enabling reliable messaging between enterprise applications

We have a Technical dashboard for tracking messages in message bus and viewing execution history of orchestrations
We have a comprehensive API management platform that let us build and test APIs, define run-time governance policies, migrate APIs between environments, and monitor and report on API usage
Which assets are placed under version control?
That are not assests placed under version control
We have assests placed under version control only for critical business applications
We have assests placed under version control only for applications that are in test phase
We have assets placed under version control only for database
We have one service monitoring and performance management supports by deployment of new services
We have one registry with metadata that is used to manage enterprise service assets
We have Definitive Media Library (DML) that save all masters copies of versions
What is your current change management process?
There is not any change management process
There is defined metrics for change management process
There is change management process but only for pilot projec(s)
There is a change management process only for critical business processes
There is one change managment process across the organizaiton
We have change management process that is applied following one framework like COBIT5
There is one available specialized team to change mangament process
What tools are used for configuration management?
We don't use any tool to configuration management
There is documentation of all tools for configuration management
We only use tools for configuration management in critical application
We use tools for configuration management for all applications in our organization
The tool allow us to save information of all tested configuration versions
The configuration management follow IT service management standard, defined in ITIL based in standard ISO/IEC 20000
We use one dynamic configuration management
Do you have mechanisms for looking up global objects by searching on their characteristics?
We don't have mechanisms for looking up global objects
We have doumentation for what mechanism do we will need
We have one pilot project for experimental service registry
We have one repository only for critical business applications
We have one repository for all internal application services in organization
We have one repository for all internal and external application services in organizaiton
We have dynamic search by objects following a global registry
What does your current operational architecture look like?
We don't have one operation architecture
We have Operating Level Agreements (OLAs) that supports operational architecture

We have one pilot project that supports operational architecture
We have one operational architecture only for critical applications of our organization
We have one operational architecture for all internal application services
We have one operational architecture for all internal and external application services
We have one operational architecture that supports and manage IT services infrastructure
How does your operational architecture support the non-functional requirements for applications and services?
Our operation architeture don't support the non-functional requirements for applications and services
There is only defined which non-functional requirements exists for applications and services
We have one pilot project that supports the non-functional requirements for applications and services
We have a document management system for all requirements of applications and services
We have a system that tests the applicatons and services before they upload to time-to-market
We have one management backup system for all application and services of our organization
We use one IT governance framework such as ITIL for technical implementations of operational architecture

Anexo C – Formulário do Assessment

Página 1 de 8

Survey - Assessment do OSIMM (Maturity Model)

This interview is being done in a Master Thesis context.
The purpose of these interviews is to evaluate my thesis proposal based on the criteria proposed by me according to the maturity model OSIMM.

Name of respondent
This is an obvious question, not relevant for analyzing the results.

Function title of respondent
This is an obvious question, not relevant for analyzing the results.

Email
This is optional

Business Dimension

1) How are identified the major business drivers for this initiative?

2) What is the business vision and goals, and how are these related to what IT is currently doing?

3) Is your current Business Process Architecture formally defined, documented, and governed?

4) Is your Business Process Architecture complete and up-to-date?