

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**ARQUITETURA APLICADA PARA O DESENVOLVIMENTO
DE SISTEMAS EM PLATAFORMA SOA**

ANDRÉ AMARO TOFFANELLO

ORIENTADOR: RICARDO STACIARINI PUTTINI

**DISSERTAÇÃO DE MESTRADO EM ENGENHARIA
ELÉTRICA**

PUBLICAÇÃO: 464/10 ENE/PG

BRASÍLIA/DF: 26 de fevereiro 2010

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**ARQUITETURA APLICADA PARA O DESENVOLVIMENTO
DE SISTEMAS EM PLATAFORMA SOA**

ANDRÉ AMARO TOFFANELLO

DISSERTAÇÃO DE MESTRADO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE.

APROVADA POR:

**RICARDO STACIARINI PUTTINI, Doutor, UnB
ORIENTADOR**

**RAFAEL TIMÓTEO DE SOUSA JUNIOR, Doutor, UnB
EXAMINADOR INTERNO**

**GEORGES DANIEL AMVAME NZE, Doutor, UnB
EXAMINADOR EXTERNO**

**FLÁVIO ELIAS, Doutor, UnB
SUPLENTE**

DATA: BRASÍLIA/DF, 26 DE FEVEREIRO DE 2010.

FICHA CATALOGRÁFICA

TOFFANELLO, ANDRÉ AMARO
ARQUITETURA APLICADA PARA O DESENVOLVIMENTO DE SISTEMAS EM
PLATAFORMA SOA [Distrito Federal] 2010.

Preâmbulo XII, 82 p., 297 mm (ENE/FT/UnB, Mestre, Engenharia Elétrica, 2010).

Dissertação de Mestrado – Universidade de Brasília, Faculdade de Tecnologia. Departamento de Engenharia Elétrica.

1. Sistemas de Informação
2. Arquitetura de Desenvolvimento de Sistemas
3. Arquitetura Orientada a Serviços

I. ENE/FT/UnB.

REFERÊNCIA BIBLIOGRÁFICA

TOFFANELLO, André Amaro 2010. Arquitetura Aplicada para o Desenvolvimento de Sistemas em Plataforma SOA. Projeto Final de Graduação Dissertação de Mestrado, Publicação 464/10 ENE/PG Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 82p.

CESSÃO DE DIREITOS

NOME DO AUTOR: André Amaro Toffanello

TÍTULO DA DISSERTAÇÃO: ARQUITETURA APLICADA PARA O DESENVOLVIMENTO DE SISTEMAS EM PLATAFORMA SOA.

GRAU/ANO: Mestre FEV/2010.

É concedida à Universidade de Brasília permissão para reproduzir cópias deste Projeto Final de Dissertação de Mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desta dissertação de mestrado pode ser reproduzida sem a autorização por escrito do autor.

André Amaro Tofanello
Rua 30 sul, lote 07, Bloco C apartamento 203
CEP 71.925-360 – Brasília – DF - Brasil

Dedicatória

Dedico este trabalho aos meus pais pelo amor, carinho e apoio durante toda vida, a minha família em especial ao meu filho Giovanni Leal Toffanello. Aos amigos.

AGRADECIMENTOS

Ao meu orientador, mais uma vez Professor Ricardo Staciarini Puttini, pelo constante apoio, incentivo, dedicação e orientação essenciais para o desenvolvimento deste trabalho e para o meu desenvolvimento como pesquisador, profissional e pessoal.

RESUMO

Esta dissertação tem o objetivo de apresentar metodologia aplicada para o desenvolvimento de Soluções Informatizadas através das técnicas de orientação a serviço, SOA – Arquitetura Orientada a Serviço. Traz como experimento, à descrição de uma implantação de plataforma tecnológica Oracle SOA Suite.

Este Trabalho discorre sobre a percepção da necessidade presente em apresentar uma arquitetura de desenvolvimento de sistemas de informação, que quebra o então paradigma dos atuais modelos de desenvolvimento. Considerada pelo mercado como um novo paradigma, a Arquitetura Orientada a Serviço ou simplesmente SOA, vem refletindo a inovação a cerca da abordagem estratégica das organizações em arquitetar seus sistemas de informação específicos.

Quebrar paradigmas representa, em uma organização de TI, onde as atitudes do passado presente que antes davam certo, podem não ser adequadas para os dias de hoje; então, é necessário realmente aposentar certos modelos, mais aquilo que está sedimentado tem dificuldades em mudar.

O princípio da SOA foca na construção de porções autônomas de software, reutilizáveis, abstraídos da lógica tradicional de programação, chamados de serviços. Os serviços possuem contratos formais que informam o que o serviço faz e como ele se comunica. Definir padrões de serviços requer muita disciplina uma vez que para obedecer ao princípio da reutilização é necessária orquestração, para que possam suprir as necessidades do processo de negócio.

Os estudos deste trabalho apresentam uma abordagem por processos de negócio, que visa unir cada vez mais a área de Tecnologia da Informação – TI com as áreas de negócios de uma organização através do levantamento, melhoria e modelagem de processos de negócio, apoiado por uma plataforma tecnológica que suporte uma concepção de uma arquitetura que aporte métodos de orientação a serviços.

Por fim é apresentado um estudo de caso da metodologia de referência da Agência Nacional de Vigilância Sanitária – ANVISA, aplicada as necessidades da área de TI a fim de

estruturar o desenvolvimento de soluções informatizadas que proporcione a utilização de orientação a serviço, com o objetivo de racionalizar a arquitetura dos sistemas de informação em benefício do alinhamento aos processos de negócio.

ABSTRACT

This paper aims to present methodology for the development of informatics solutions through the techniques of SOA - Service Oriented Architecture. The experiment was developed with the implementation of Oracle SOA Suite technology.

This work discusses the perceived need to present development architecture of information systems, to break the paradigm of current development models. Considered by the market as a new paradigm, SOA bring an innovation about reflecting the strategic approach for organizations to devise their specific information systems.

Break paradigms represents, in an IT organization where traditional models for systems development may not be appropriate for necessities today, so must be retire these models, what may not be an easy task.

The principle of SOA focuses on building software portions autonomous, reusable, abstracted from the traditional logic programming, called services. The services have formal contracts that tell what the service does and how it communicates. Set service standards require much discipline as to obey the principle of reuse is necessary orchestration, so that they can meet the needs of the business process.

Studies of this work presents an approach by business processes, which aims to unite increasingly the area of Information Technology - IT with the business areas of an organization through the survey, modeling and improving business processes, supported by a platform technology that claims a architecture concepts that input methods of service orientation.

Finally is presented a case study of the methodology of reference of the National Sanitary Surveillance Agency - ANVISA, applied to the needs of IT in order to structure the development of computerized solutions to provide the use of service orientation, with the objective of optimize the architecture of solutions to align IT area with the business process.

SUMÁRIO

1. INTRODUÇÃO.....	1
2. SOA - ARQUITETURA ORIENTADA A SERVIÇO	4
2.1. ARQUITETURA DE SOFTWARE.....	4
2.1.1. Visões da Arquitetura de Software	5
2.2. PRINCÍPIOS DA ARQUITETURA ORIENTADA A SERVIÇO	6
2.3. DEFINIÇÕES DA ARQUITETURA ORIENTADA A SERVIÇO	10
2.4. ARQUITETURA DE REFERÊNCIA SOA.....	12
2.5. BENEFÍCIOS DA ARQUITETURA ORIENTADA A SERVIÇOS.....	17
3. ABORDAGEM POR PROCESSOS DE NEGÓCIO - (BPM).....	19
3.1. CICLO DE VIDA BPM.....	20
3.2. MODELAGEM DE PROCESSO DE NEGÓCIO	23
3.3. BENEFÍCIOS DA MODELAGEM.....	24
3.4. PADRÕES DE MODELAGEM E NOTAÇÕES	25
4. ESTUDO DE CASO - AGÊNCIA NACIONAL DE VIGILÂNCIA SANITÁRIA – ANVISA.....	33
4.1. VISÃO GERAL DA ARQUITETURA	34
4.2. ARQUITETURA PARA AUTOMAÇÃO DE PROCESSOS – ANVISA.....	37
4.3. BENEFÍCIOS ALCANÇADOS.....	51
5. CONCLUSÃO	54
5.1. TRABALHOS FUTUROS.....	55
REFERENCIA BIBLIOGRÁFICA.....	56
ANEXO I - PLATAFORMA TECNOLÓGICA	58

ÍNDICE DE TABELAS

Tabela 1 - Definições a cerca da SOA	11
Tabela 2 – Métricas da Adoção de SOA	51
Tabela 3 – Requisitos de Softwares da Plataforma Oracle empregada na ANVISA.	58

ÍNDICE DE FIGURAS

Figura 1 - Modelo de Referencia de SOA. <i>Service-Oriented Architecture (SOA) Compass: Business Value, Planning, and Enterprise Roadmap</i> [2].....	13
Figura 2 – Visão Lógica da ESB – <i>Enterprise Service Bus. Service-Oriented Architecture (SOA) Compass: Business Value, Planning, and Enterprise Roadmap</i> [2].....	16
Figura 3 – Ciclo de Vida BPM. <i>ABPMP – CBOK, [1]</i>	21
Figura 4 – Exemplo de Diagrama Processo de Negócio Simples. <i>Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento, ABPMP</i> [1].	26
Figura 5 – Exemplo de Diagrama de Processo de Negócio mais detalhado e complexo. <i>Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento, ABPMP</i> [1]	26
Figura 6 – Exemplo de Diagrama de Processo de Negócio tradicional com raias. <i>Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento, ABPMP</i> [1].....	27
Figura 7 - Exemplo de diagrama de Processo de Negócio com participantes. <i>Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento, ABPMP</i> [1].....	27
Figura 8 – Exemplo de Diagrama de Processo de Negócio em alto nível. <i>Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento, ABPMP</i> [1].....	28
Figura 10 - Exemplo de Diagrama de Processo de Negócio em baixo nível. <i>Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento, ABPMP</i> [1].....	28
Figura 11 – Cenário atual de Integração de Sistemas na ANVISA	35
Figura 12 – Cenário Típico de SOA na ANVISA.....	36
Figura 13 – Visão Lógica da SOA - ANVISA.....	37
Figura 14 – Padrão de Projetos Business <i>delegate</i>	38
Figura 15 – Padrão de projetos para Localização de Serviços	39
Figura 16 – Integração entre camada e sub-camadas da arquitetura.....	44
Figura 17 – Comunicação entre dois Serviços	45
Figura 18 – Diagrama de Seqüência de Chamada de Serviço.....	46
Figura 19 – Dashboard do BAM	49
Figura 20 – Canal de Comunicação entre Web-Services	64

Lista de Acrônimos

ANVISA	Agência Nacional de Vigilância Sanitária.
API	Application Programming Interface.
BAM	Business Active Monitoring.
BPEL	Business Process Execution Language.
ESB	Enterprise Service Bus.
KPI	Key Performance Indicator.
MQ	Message Queue - recurso de enfileiramento de mensagens utilizado evitar que um servidor seja sobrecarregado por um número excessivo de requisições feitas concomitantemente.
REST	Representational State Transfer.
SOA	Arquitetura Orientada a Serviço. (Service Oriented Architecture)
SOAP	Simple Object Access Protocol.
UDDI	Universal Description, Discovery and Integration.
JCA	Java Connector Architecture - uma arquitetura desenvolvida de modo a facilitar a comunicação entre servidores J2EE e EIS 's,
JMS	Java Messaging Service - é uma arquitetura existente J2EE que permite troca de mensagens de maneira assíncrona

1. INTRODUÇÃO

Service Oriented Architecture (SOA) - Arquitetura Orientada a Serviços, é um modelo de arquitetura para o desenvolvimento de sistemas de informação onde preconiza que as funcionalidades dos sistemas são implementadas na forma de serviços. Frequentemente estes serviços são disponibilizados através de um "barramento de serviços" (*Enterprise Service Bus - ESB*) que disponibiliza interfaces ou contratos, acessíveis através de *web services* ou outra forma de comunicação entre aplicações. SOA é baseado nos modelos da computação distribuída e utiliza o mecanismo requisição/resposta para estabelecer a comunicação entre os sistemas clientes e os sistemas que implementam os serviços.

Além da perspectiva estritamente técnica, a arquitetura orientada a serviços também se relaciona com determinados conceitos e conjuntos de "melhores práticas" que criam processos para definir tarefas de encontrar, implementar e gerenciar os serviços disponibilizados.

O paradigma de SOA redefine a forma e os métodos empregados para a análise e construção dos sistemas de informação, para isso, é necessário que observe os processos de negócios de uma organização, utilizando métodos de mapeamento e modelagem destes processos. As melhores práticas para mapeamento e modelagem de processos de negócio são definidas pela organização profissional sem fins lucrativos chamada ABPMP - Associação de Profissionais de Gerenciamento de Processos de Negócio, esta organização é independente de fornecedores, dedicada ao desenvolvimento dos conceitos de gerenciamento de processos de negócio e suas práticas.

O BPM – *Business Process Modeling*, é uma abordagem disciplinada para identificar, desenhar, executar, documentar, medir, monitorar, controlar e melhorar processos de negócio automatizados ou não para alcançar os resultados pretendidos consistentes e alinhados com as metas estratégicas de uma organização. BPM envolve a definição deliberada, colaborativa e cada vez mais assistida por tecnologia, melhoria, inovação e gerenciamento de processos de negócio ponta-a-ponta que conduzem a resultados de negócios, criam valor e permitem que uma organização cumpra com seus objetivos de negócio com mais agilidade. [1]

Há vários conceitos fundamentais, básicos, que definem BPM, incluindo noções como: [1]

- BPM é uma disciplina de gerenciamento em um conjunto de tecnologias habilitadoras.
- BPM aborda um trabalho ponta-a-ponta e diferencia entre conjuntos de subprocessos, tarefas, atividades e funções.
- BPM é um conjunto contínuo, em curso, de processos com o foco no gerenciamento de processos de negócio ponta-a-ponta nas organizações.
- BPM inclui modelagem, análise, desenho e medição de processos de negócio de uma organização.
- BPM requer um compromisso significativo da organização que frequentemente introduz novos papéis, responsabilidades e estruturas às organizações tradicionais orientadas a funções.
- BPM é habilitada por tecnologia através de ferramentas para modelagem, simulação, automação, integração, controle e monitoramento de processos de negócio e de sistemas de informação que suportam esses processos.

O sucesso na adoção de SOA para o desenvolvimento de sistemas de informação em uma organização exige forte relacionamento com a área de negócio através da modelagem de processos de negócio, uma vez que a representação nas plataformas tecnológicas promove ganhos no entendimento, na documentação e na construção de sistemas de informação orientada estritamente ao negócio da organização.

O estudo de caso apresentado nesta dissertação é uma Arquitetura de Referência para o Desenvolvimento de Sistemas em SOA na ANVISA – Agência Nacional de Vigilância Sanitária. Este estudo mostra como uma organização se prepara para este novo paradigma de desenvolvimento de sistemas de informação. Com uma arquitetura apoiada na tecnologia Oracle SOA Suíte, a ANVISA promove automação de processos de negócio iniciada pela

modelagem dos processos, permeando pela eleição de serviços candidatos, pela implementação dos serviços na infra-estrutura tecnológica até a interação com o usuário final.

A dissertação está assim organizada.

O capítulo 2 apresenta um estudo sobre arquitetura de software, princípios e definições da Arquitetura Orientada a Serviço. Estes princípios e definições promovem o estabelecimento de bases para o desenvolvimento, as estratégias e os métodos de construção e a disponibilização de serviços.

O capítulo 3 apresenta definições sobre a Modelagem de Processos de Negócio baseados no BPM CBOK [1]. As melhores práticas apresentadas para modelagem de processos de negócio sustentam a estratégia da proposta de Arquitetura de Desenvolvimento de Sistemas de Informação desenvolvida para a ANVISA. Este capítulo aborda aspectos necessários para o entendimento da Automação de Processo de Negócio para uma implementação de SOA.

No capítulo 4 é iniciada a apresentação do estudo de caso da ANVISA, que por intermédio da sua Gerencia Geral de Tecnologia da Informação apostou em uma nova arquitetura de Desenvolvimento de Sistemas de Informação totalmente aderente aos preceitos da Plataforma SOA.

Por fim, no capítulo 5 são apresentadas as conclusões acerca da dissertação e desenvolvimentos de trabalhos futuros.

O Anexo I traz a implementação da arquitetura de referência refletida em uma implementação prática na plataforma Oracle SOA Suite.

2. SOA - ARQUITETURA ORIENTADA A SERVIÇO

Este capítulo apresenta um estudo sobre arquitetura de software, princípios e definições da Arquitetura Orientada a Serviços, das quais estabelecem bases para o desenvolvimento, estratégias e métodos de construção e disponibilização de uma Arquitetura específica de sistemas de informação orientada a serviços.

2.1. Arquitetura de Software

A **arquitetura de software** de um sistema de informação consiste de componentes de software, suas propriedades externas, e seus relacionamentos com outros sistemas. O termo também se refere à documentação da arquitetura de software do sistema. A documentação da arquitetura do software facilita: a comunicação entre os agentes, registro das decisões iniciais acerca do projeto de alto-nível, e permite o reuso dos padrões entre projetos.

O campo da engenharia de software tem lidado com problemas associados como a complexidade da informação desde sua criação. Os primeiros problemas de complexidade foram resolvidos pelos desenvolvedores através da escolha das técnicas de estruturação de dados, do desenvolvimento de algoritmos e pela aplicação de conceitos de separação de escopos. Embora o termo arquitetura de software seja relativamente novo na indústria, os princípios fundamentais deste campo vêm sendo aplicados esporadicamente pela engenharia de software desde o início dos anos 80. As primeiras tentativas de capturar e explicar a arquitetura de software se apresentou imprecisas e desorganizadas – frequentemente caracterizadas por um conjunto de diagramas. Durante o decorrer da década de 90 houve um esforço concentrado para definir e codificar os aspectos fundamentais desta disciplina. Inicialmente um conjunto de padrões de projeto, estilo, melhores práticas, descrição de linguagens e lógica formal foram desenvolvidas durante este período.

A disciplina de arquitetura de software é centrada na idéia da redução da complexidade através da abstração e separação de interesses. O glossário do site oficial SOFTWARE ENGINEERING INSTITUTE (Instituto de Engenharia de Software) ¹ descreve que arquitetura de software é a estrutura ou estruturas de um sistema, com todos os elementos

1 SEI (2010). www.sei.cmu.edu/architecture/start/glossary - Página visitada em 2010-01-31.

de software vendo e tendo suas propriedades vistas por todos os outros elementos e relacionamentos.

Sendo a arquitetura de sistema uma disciplina em maturação, sem regras claras, a ação do arquiteto é ainda uma composição de arte e ciência. Os aspectos de arte da arquitetura de software são devido ao fato que os sistemas de software comerciais suportam alguns aspectos de um negócio ou missão. Como os direcionamentos de negócio chave para o suporte sistemas são descrito nos cenários, requisitos não funcionais de sistema, também conhecidos como atributos de qualidade, que determinam como um sistema ira se comportar. Cada sistema é único devido à natureza do negócio que ele suporta tal que o nível dos atributos de qualidades exigidos de um sistema como compatibilidade, extensibilidade, confiabilidade, manutenibilidade, disponibilidade, segurança, usabilidade, dentre outros – irão variar com cada aplicação.

Para trazer a perspectiva do usuário para dentro da arquitetura de software, pode-se dizer que a arquitetura de software dá a direção dos passos que serão tomados e as tarefas envolvidas em cada área de especialidade e interesse do usuário, por exemplo, *stakeholders* de sistemas de software, os desenvolvedores de software, o grupo de suporte ao software do sistema operacional, aos testadores e usuários de negocio final. Neste sentido, a arquitetura de software se torna a ligação das múltiplas perspectivas que um sistema traz nele embutido. O fato de que estas várias perspectivas diferentes possam ser postas juntas em uma arquitetura de software padrão justifica e valida a necessidade de criação da arquitetura de software antes do desenvolvimento do software para que projeto alcance a maturidade.

2.1.1. Visões da Arquitetura de Software

A arquitetura de software é normalmente organizada em visões, as quais são análogas aos diferentes tipos de impressões utilizadas no estabelecimento da arquitetura. Na ontologia estabelecida pela ANSI/IEEE 1471-2000, visões são instâncias de pontos de vista, onde um ponto de vista existe para descrever a arquitetura na perspectiva de um conjunto de *stakeholders* e seus consortes.

Algumas possíveis visões são: [13]

- Visão funcional/lógica.

- Visão de código.
- Visão de desenvolvimento/estrutural.
- Visão de concorrência/processo.
- Visão físico-evolutiva.
- Visão de ação do usuário/*feedback*.

Várias linguagens para descrição da arquitetura de software foram desenvolvidas, mas nenhum consenso foi ainda alcançado em relação a qual conjunto de símbolos ou sistema representação deve ser adotado. É possível que a UML - *Unified Modeling Language* irá estabelecer um padrão para representação de arquitetura de software, porém, é possível considerar que os desenvolvimentos efetivos de software devem contar com a compreensão única das restrições de cada processo de negócio.

2.2. Princípios da arquitetura orientada a serviço

Posto o objetivo específico desta dissertação em apresentar a arquitetura de referência para implementação de processos de desenvolvimento de sistemas de informação em SOA SOA, cito:

"Mais do que uma tecnologia, SOA também influencia regras e processos de negócios, além de muitas vezes implicar reengenharia de software simultaneamente." Gartner Group.

SOA MANIFESTO²

"Orientação a Serviço é um paradigma que molda o que você faz.

Arquitetura Orientada a Serviço (SOA) é um tipo de arquitetura que resulta da aplicação de orientação a serviço."

"Nós temos aplicado orientação para ajudar organizações a, de maneira consistente e sustentável, agregar valor ao negócio, com maior agilidade e efetividade de custos, em alinhamento com a dinâmica das necessidades de negócio."

Através de nosso trabalho, priorizamos:

Valor do negócio em relação a estratégia técnica;

Objetivos estratégicos em relação a benefícios específicos de projetos;

Interoperabilidade intrínseca em relação à integração personalizada;

Serviços compartilhados em relação a implementações de propósito específico;

Flexibilidade em relação à otimização; e

Refinamento evolutivo em relação à busca da perfeição inicial.

Isso é, mesmo valorizando os itens à direita, valorizamos mais os itens à esquerda..

Tradução Prof. Ricardo Staciarini Puttini"

AUTHORS

Ali Arsanjani Grady Booch Toufic Boubez Paul C. Brown David Chappell John deVadoss	Thomas Erl Nicolai Josuttis Dirk Krafzig Mark Little Brian Loesgen Anne Thomas Manes	Steve Ross-Talbot Stefan Tilkov Clemens Utschig-Utschig Herbjörn Wilhelmsen
---	---	--

O SOA Manifesto foi anunciado em 23 de outubro de 2009 durante o encerramento do 2nd *International SOA Symposium* em Rotterdam na Holanda, com o objetivo de fornecer uma declaração formal de princípios, intenções e ambições do SOA.

Além do texto apresentado no início do capítulo, o resultado do SOA Manifesto constitui de um conjunto de princípios que representam as diretrizes para o desenvolvimento

² http://www.soa-manifesto.org/default_portuguese.html

de sistemas de informação apoiados em uma Arquitetura Orientada a Serviços. A seguir é aprestada uma tradução livre destes princípios que conduziram este e outros trabalhos.

- Respeitar o social e estrutura de poder de cada organização.
- Reconhecer que, em última instância que a implementação de SOA exige mudança em muitos níveis da organização.
- O âmbito da adoção de SOA pode variar de organização para organização e devem manter esforços gerenciáveis e dentro dos limites significativos.
- Produtos e normas por si só, não fornecerá SOA para o paradigma de orientação a serviços para organização.
- SOA pode ser realizado através de uma variedade de tecnologias e padrões.
- Estabelecer um conjunto uniforme de normas organizacionais e políticas baseadas a indústria, de fato, e as normas comunitárias.
- Prosseguir uniformidade para o exterior da organização, permitindo a diversidade no interior.
- Identificar os serviços através de uma colaboração com a organização e interessados tecnologia.
- Maximizar a utilização do serviço, considerando a atual e futuro âmbito de utilização.
- Verificar que os serviços satisfazem as necessidades e objetivos do negócio.
- Evoluir dos serviços e sua organização em resposta ao uso real.
- Separar os diferentes aspectos de um sistema que a mudança em ritmos diferentes.
- Reduzir as dependências implícitas e publicar todas as dependências externas para aumentar a robustez e diminuir o impacto da mudança.
- Em cada nível de abstração, organizar cada serviço em torno de uma coesa e a unidade gerenciável de funcionalidade.

Outros oito princípios considerados de fato são apresentados a fim de canalizar as diretrizes e estratégias para o desenvolvimento de sistemas de informação em SOA, com o foco nos serviços, são eles: [12]

1. **Serviços são REUTILIZÁVEIS:** Esta talvez seja a característica que mais comumente está associada a SOA. Serviço agnóstico tem maior probabilidade de reutilização.

Isto permite que a área de TI forneça respostas rápidas a novos requerimentos das áreas de negócio de uma organização, desde que estes Serviços sejam definidos em conjunto: TI e área de negócios. Serviço não é propriedade de uma equipe de desenvolvimento, o serviço é ativo da organização.

2. **Serviços compartilham um CONTRATO FORMAL:** Todo serviço possui um “contrato” entre o requisitante e o provedor deste serviço. O “contrato” informa o que o Serviço faz e como se comunica (o que deve receber e o que deve entregar). Desnecessário dizer que a deve definir seus padrões de descrição e que isto requer muita disciplina.

3. **Serviços possuem BAIXO ACOPLAMENTO:** O acoplamento se refere a uma medida de dependência. Um baixo acoplamento significa que implementações específicas de um Serviço podem ser substituídas, modificadas e evoluídas sem que os consumidores deste Serviço sintam qualquer descontinuidade. Fica claro que Serviços não devem expressar à lógica (regras) de negócio.

4. **Serviços ABSTRAEM A LÓGICA:** Serviços são como “caixas pretas”, o que significa que a lógica não precisa nem deve ser exposta, simplificando o contrato formal. Guardando os detalhes também permite que possamos modificar a lógica evoluindo-a ao longo do tempo sem comprometer as “obrigações” do Serviço publicadas em seu contrato formal.

5. **Serviços são CAPAZES DE SE COMPOR:** Separar um grande problema em vários pequenos não é uma novidade. Este mesmo princípio pode ser utilizado na construção dos Serviços. Assim, um Serviço pode “chamar” outro(s) para executar a sua tarefa. A composição também é uma forma de reutilização.

6. **Serviços são AUTÔNOMOS:** Autonomia significa a capacidade de se autogovernar. Um Serviço autônomo é aquele que independe de um elemento externo para executar sua lógica.

7. **Serviços evitam ALOCAÇÃO DE RECURSOS por longos períodos:** Devido ao fato de que um Serviço será reutilizado, deve-se tomar o cuidado de não se criar muitas instâncias deste Serviço onerando a infra-estrutura. Isto significa que para um Serviço ter disponibilidade, ele deve evitar reter informações específicas a uma determinada atividade.

8. Serviços devem possuir a CAPACIDADE DE SEREM DESCOBERTOS: Um contrato formal bem descrito e padronizado evita que novos requerimentos resultem em Serviços redundantes. Além disto, a arquitetura deve prover mecanismos que facilitem o descobrimento dos Serviços através de Diretórios e Registros.

Estes princípios embasam o conhecimento que regem, na maioria dos casos, as bases para definições de arquitetura de sistemas de informação baseada em SOA.

Estas características se relacionam de uma forma interdependente. Podemos deduzir várias frases que mostram esta interdependência, como por exemplo: [12]

- BAIXO ACOPLAMENTO promove a DISPONIBILIDADE.
- CONTRATO forma as bases para o DESCOBRIMENTO.
- DISPONIBILIDADE maximiza oportunidades de REUTILIZAÇÃO.
- DESCOBRIMENTO fornece meios para promover REUTILIZAÇÃO.
- AUTONOMIA aumenta a DISPONIBILIDADE.
- BAIXO ACOPLAMENTO permite AUTONOMIA.

2.3. Definições da arquitetura orientada a serviço

É notório que uma definição pragmática de SOA ainda permeia pelas esferas intelectuais da engenharia de software, assim é válido relevar que uma definição precisa e única estaria em algum momento, equivocada ao olhar de outros. Esta afirmação condiz com o princípio de que, a adoção de SOA pode variar de organização para organização e devem manter esforços gerenciáveis e dentro dos limites significativos

Uma arquitetura de software é um conceito abstrato que dá margem a uma série de definições. A definição usada pelo ANSI/IEEE afirma que uma arquitetura de software trata basicamente de como os componentes fundamentais de um sistema de informação se relacionam intrinsecamente e extrinsecamente [ANSI/IEEE, 2000]. Uma arquitetura orientada a serviços tem como componente fundamental o conceito de serviços.

Em SOA é preconizado que as funcionalidades de um sistema de informação sejam implementadas na forma de serviços. Frequentemente estes serviços são disponibilizados através de um "barramento de serviços" (*ESB – Enterprise Service Bus*) que disponibiliza interfaces ou contratos, acessíveis através de comunicação entre aplicações.

SOA é baseada nos modelos da computação distribuída e utiliza o paradigma requisição/resposta para estabelecer a comunicação entre os sistemas clientes e os sistemas que implementam os serviços.

Na tabela 1 são elencados conceitos utilizados para a definição da arquitetura.

Termo	Definição / Comentário
Serviço	É uma função independente, sem estado (<i>stateless</i>) que aceita uma ou mais requisições e devolve uma ou mais respostas através de uma interface padronizada e bem definida. Serviços podem também realizar partes discretas de um processo tal como editar ou processar uma transação. Serviços não devem depender do estado de outras funções ou processos. A tecnologia utilizada para prover o serviço, tal como uma linguagem de programação, não pode fazer parte da definição do serviço.
Orquestração	Processo de sequenciar serviços e prover uma lógica adicional para processar dados. Não inclui uma representação de dados.
<i>Stateless</i>	Não depende de nenhuma condição pré-existente. Os serviços não devem depender de condições de outros serviços. Eles recebem todas as informações necessárias para prover uma resposta consistente. O objetivo de buscar a característica de <i>stateless</i> dos serviços é possibilitar que o consumidor do serviço possa sequenciá-lo, ou seja, orquestrá-los em vários fluxos (algumas vezes chamados de <i>pipelines</i>) para executar a lógica de uma aplicação.
Provedor	O recurso que executa o serviço em resposta a uma requisição de um consumidor.
Consumidor	É quem consome ou pede o resultado de um serviço fornecido por um provedor.
Descoberta	SOA se baseia na capacidade de identificar serviços e suas características. Conseqüentemente, esta arquitetura depende de um diretório que descreva quais os serviços disponíveis dentro de um domínio.
<i>Binding</i>	A relação entre os serviços do provedor e do consumidor deve ser idealmente dinâmica; ela é estabelecida em tempo de execução através de um mecanismo de <i>binding</i> .

Tabela 1 - Definições a cerca da SOA

2.4. Arquitetura de Referência SOA

Ao definir uma arquitetura para desenvolvimento de sistemas de informação orientada a serviços, é necessário ter em mente uma arquitetura de referência no contexto que defina os blocos de construção da SOA: serviços, componentes de serviços e fluxos que, coletivamente, apoiar processos de negócios e os objetivos de negócio. A arquitetura de referência fornece características e definições para cada camada e as relações entre eles e contribui para as especificações de uma arquitetura específica. Estas camadas facilitam a criação de projetos de arquitetura em SOA e ajudam na reutilização de soluções e recursos dentro de uma organização.

A figura 2 mostra uma arquitetura em nove camadas, com cinco camadas horizontais e quatro camadas verticais. As camadas horizontais seguem o princípio básico de um modelo de arquitetura, em camadas, tipo *Architecture Building Blocks* (ABB) onde as camadas acima podem fazer acesso as camadas inferiores, enquanto que as camadas inferiores não podem fazer acesso às camadas acima. As camadas verticais geralmente contêm ABB's que são transversais por natureza, o que implica que eles podem ser aplicados e utilizados pela ABB's em uma ou mais das camadas horizontais. Isso também pode ser chamado de uma arquitetura em camadas parciais, porque qualquer camada acima não é necessário interagir com os elementos de sua camada imediatamente inferior. A Figura 1 mostra um exemplo de arquitetura de referência lógica SOA segundo Bieberstein da IBM em seu livro eletrônico *Service Oriented Architecture (SOA) Compass: Business Value, Planning, and Enterprise Roadmap*. [2].

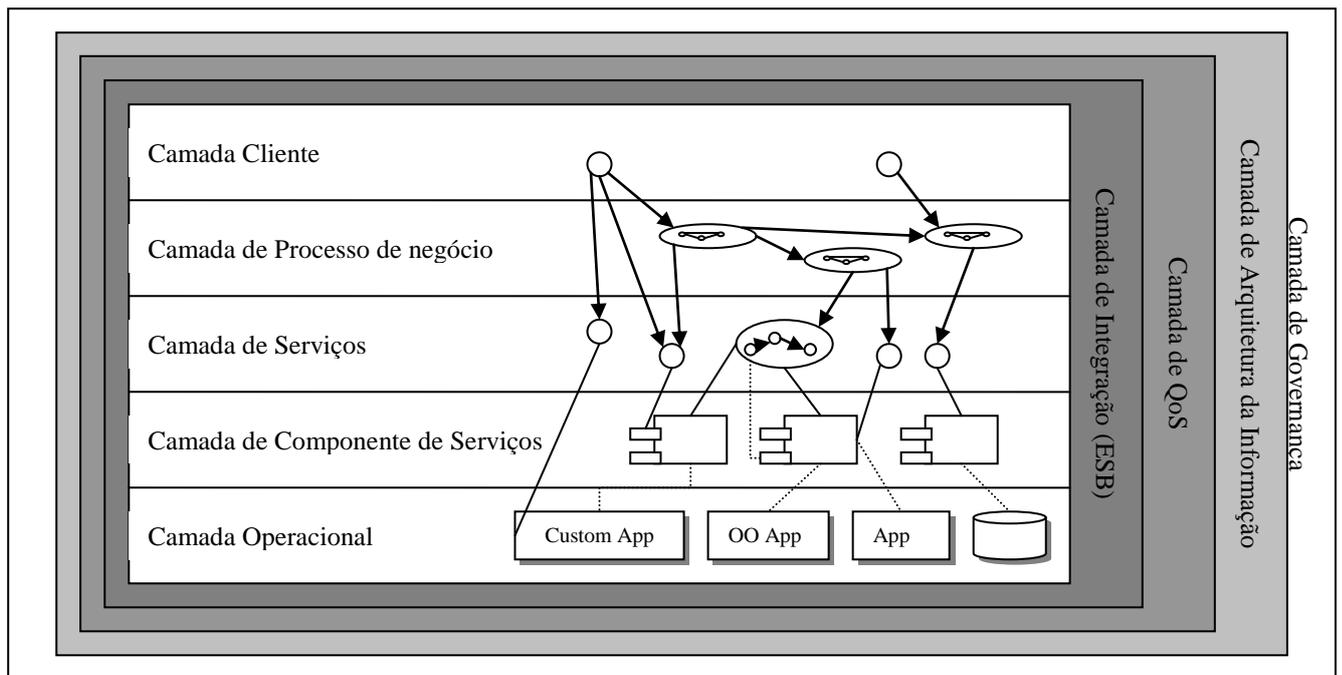


Figura 1 - Modelo de Referência de SOA. *Service-Oriented Architecture (SOA) Compass: Business Value, Planning, and Enterprise Roadmap*[2].

Por exemplo, um canal de acesso específico pode acessar diretamente um serviço e não precisar passar por um processo de negócio. As restrições de acesso, no entanto, são ditadas pelo estilo arquitetônico, diretrizes e princípios que se aplicam a uma solução SOA dada. Este ponto de vista da arquitetura de referência SOA é independente de qualquer implementação de tecnologia específica e, portanto, é uma visão lógica. Instâncias desta arquitetura lógica podem ser desenvolvidas para uma plataforma específica e tecnológica. A seguir estão as definições de cada uma das camadas:

Camada 1- Camada Operacional: Esta camada inclui a infra-estrutura operacional que existe no ambiente de TI da organização, apoiando as atividades organizacionais. A camada operacional inclui todas as aplicações personalizadas, pacotes de aplicações, sistemas legados, sistemas de processamento de transações, plataformas tecnológicas de desenvolvimento, e as diversas bases de dados.

Camada 2 - Camada de Componentes de Serviço: nesta camada em conformidade com os contratos (parâmetros) definidos para os serviços da Camada de Serviços, um

componente de serviço pode realizar um ou mais serviços. Um componente de serviço oferece uma camada intermediária de implementação que agrega funcionalidade de múltiplas, díspares possíveis, sistemas operacionais ao mesmo tempo esconde a complexidade da integração e acesso ao serviço que é exposto para o Cliente. A vantagem deste componente intermediário advém da flexibilidade de mudar os sistemas operacionais sem afetar a definição de serviço. O Componente de Serviço fornece um ponto de execução para a realização do serviço para garantir qualidade de serviço (QoS) e o cumprimento de acordos de nível de serviço.

Camada 3 - Camada de Serviços: Esta camada inclui todos os serviços definidos no portfólio de serviços da organização. A definição de cada serviço, que constitui simultaneamente a sua sintática e semântica de informação, é definida nesta camada. Considerando que as informações sintáticas, essencialmente em torno das operações em cada serviço, a entrada e as saídas mensagens, e a definição das falhas de serviços, a informação semântica é em torno das políticas de serviços, gerenciamento de decisões, serviço de requisitos de acesso, e assim por diante. Os serviços são definidos de forma a que sejam acessíveis e invocada por parte dos canais e consumidores independentes de aplicação e o protocolo de transporte. O passo crítico é a identificação dos serviços que utilizam as diversas técnicas que podem ser empregadas para o mesmo.

Camada 4 - Camada de Processo de Negócios: processos de negócio descrevem como o negócio funciona. Um processo de negócio na visão da TI é uma representação das várias atividades coordenadas e colaborativas em uma organização para executar uma função específica do negócio representada em alto nível. Esta camada representa os processos como uma orquestra ou uma composição de serviços de baixo acoplamento, aproveitando os serviços representados na camada de serviços. A camada é também responsável pela gestão do ciclo de vida completo dos processos junto com sua orquestração e coreografia. Os dados e fluxo de informações entre as etapas de cada processo também está representado nesta camada. Processos representados nesta camada são o meio de ligação entre as necessidades de negócio e sua manifestação como soluções de TI de nível usando ABB`s de outras camadas horizontais e verticais na arquitetura de pilha. Clientes, portais e sistemas B2B na camada cliente utilizam os processos de negócio nesta camada como uma das maneiras de se invocar a funcionalidade do sistema de informação.

Camada 5 - Camada Cliente: Esta camada descreve os diversos canais através dos quais a TI entrega os sistemas de informações. Os portais podem ser na forma de diferentes tipos de usuários (por exemplo, os clientes externos e internos que funcionalidade de acesso a aplicação por meio de mecanismos de acesso como sistemas B2B, portais, clientes ricos, e outras formas). O objetivo desta camada é a padronização do protocolo de acesso e formato de dados para permitir a criação rápida de *front-ends* para os processos de negócios e serviços expostos das camadas abaixo. Algumas dessas normas têm surgido na forma de *portlets*, arquitetura de componentes de serviços (SCA) componentes, e *Web Services for Remote Portlets* (WSRP). A adesão aos mecanismos padrão para os componentes da camada de apresentação para os processos de negócio e ajuda na prestação de serviços modelo de soluções sob a forma de padrões de arquitetura padrão, que ajuda a comunidade de desenvolvedores a adotar frente comum de padrões de consumo final de serviços.

Camada 6 - Camada de Integração: Esta camada oferece a capacidade para os clientes de serviço possa localizar os prestadores de serviços e iniciar as invocações de serviço. Através de três capacidades básicas de mediação, roteamento e transformação de dados e protocolo, esta camada ajuda a promover um ambiente de serviços em que os serviços podem se comunicar uns com os outros enquanto é parte de um processo de negócio. Os principais requisitos não funcionais, tais como segurança, latência e qualidade de serviço entre as camadas adjacentes na arquitetura de referência são executadas pela estrutura apresentada na figura 3. As funções desta camada são normalmente e cada vez mais a ser coletivamente definido como o *Enterprise Service Bus* (ESB). Um ESB é uma coleção de padrões de arquitetura que utiliza padrões e protocolos abertos para implementar as três capacidades básicas desta camada e fornecer uma camada de ligação entre os clientes do serviço e o prestador do serviço, expondo os serviços somente através da ESB. Produtos ESB geralmente adicionar alguns recursos especializados para fornecer capacidades diferenciadas no mercado.

As capacidades de integração são mais comumente usados por ABBs residentes entre Camada 2 através da camada 5. Como exemplo, na camada 5 pode haver muitos clientes de serviços da organização acessando através de diferentes tipos de canais. Cada tipo de canal pode usar protocolos diferentes, HTML, WML (para usuários móveis) e *Voice XML* (para usuários IVR), para citar alguns. Cada um desses protocolos e formatos de mensagem pode

ser transmitido através de um *Extensible Stylesheet Language Transformations* (XSLT) do ESB antes que um serviço efetivo seja invocado. Esta transformação XSLT é geralmente um recurso fornecido pelo ESB. A beleza da ESB na Camada de Integração é que uma característica ou função que pode ser exposta de uma forma a seguir os padrões e protocolos abertos para o acesso pode ser conectado ao ESB para que ele esteja habilitado a tomar parte em um serviço baseado ambiente. A Figura 2 mostra a visão lógica da ESB e apresenta como são ofertados os recursos para conexão entre os clientes e prestadores de serviços, além de: para os serviços serem descobertos utilizam-se o Registro, para os serviços serem geridos, para invocações de segurança e interfaces de programação de aplicativos (API) para facilitar o desenvolvimento de serviços de conectividade.

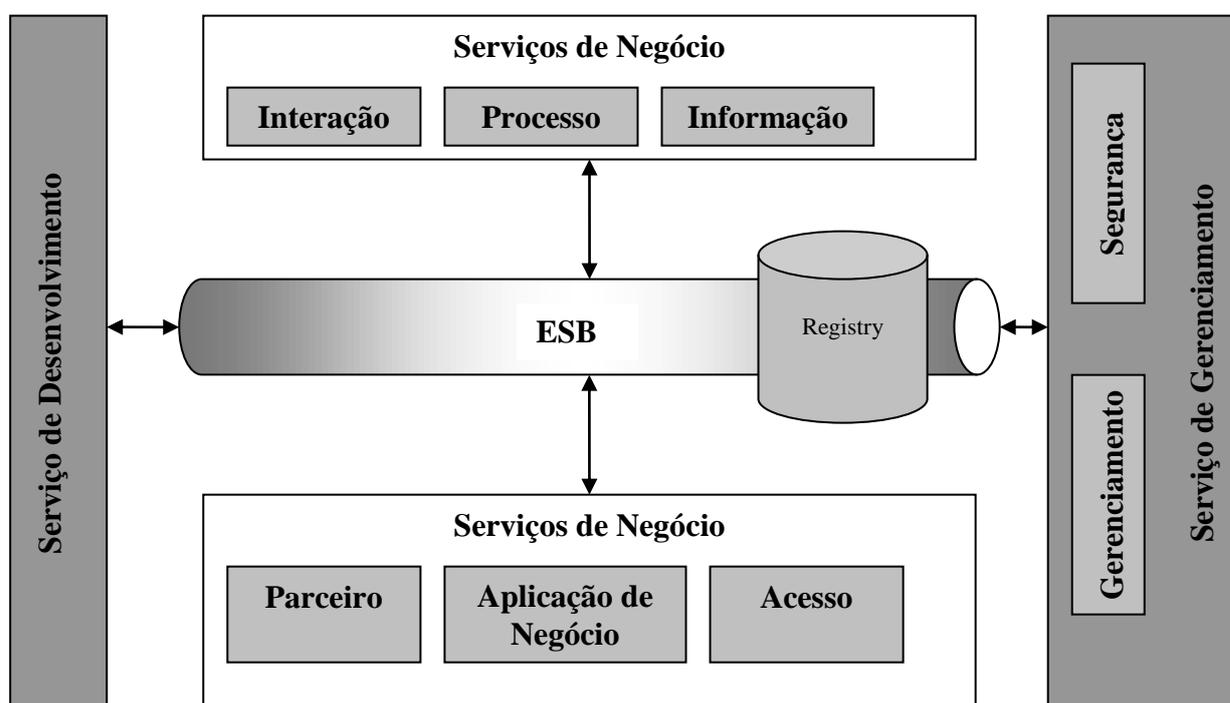


Figura 2 – Visão Lógica da ESB – *Enterprise Service Bus. Service-Oriented Architecture (SOA) Compass: Business Value, Planning, and Enterprise Roadmap* [2].

Camada 7 - Camada QoS: nesta camada que se concentra em implementação e gestão dos requisitos não-funcionais (NFR) que os serviços precisam implementar. Apesar do SOA trazer alguma proposição de valor real pelo do novo estilo arquitetônico, o modelo de programação que suporta o desenvolvimento do SOA num primeiro contato percebe-se alguns desafios inerentes que são triviais para resolver. Os desafios que surgem durante a tentativa de cumprir os princípios essenciais do SOA: abstração, padrões e protocolos abertos,

computação distribuída, infra-estruturas heterogêneas, e assim por diante. A adesão a estes requisitos de conformidade, muitas vezes torna a execução do NFRs muito mais complicado. Esta camada fornece os recursos de infra-estrutura para realizar o NFRs. Ele captura os elementos de dados que fornecem as informações em torno da não implementação de NFRs em cada uma das camadas horizontais. A padronização das NFRs supriu a segurança, disponibilidade, escalabilidade e confiabilidade.

Camada 8 – Arquitetura da Informação: Esta camada assegura uma representação adequada dos dados e informações que é exigido em uma SOA. A arquitetura de dados e representação de arquitetura de informação (juntamente com os seus principais considerações e orientações para o seu projeto e uso) em cada camada específica horizontais são as responsabilidades desta camada.

Camada 9 - Camada de Governança: Esta camada assegura a gestão adequada do ciclo de vida dos serviços. Ele é responsável por priorizar quais serviços de elevado valor deve ser aplicado, para cada uma das camadas na arquitetura, e para proporcionar uma racionalização baseada em como o serviço preenche um negócio ou objetivos da organização. Cumprimento de políticas, tanto em tempo de design e em tempo de execução que os serviços devem implementar é uma das principais responsabilidades desta camada. Essencialmente, esta camada fornece uma estrutura eficiente que supervisiona o projeto e implementação de serviços para que eles cumpram com as várias áreas de negócio da organização suas políticas de regulação e exigências.

Outra arquitetura conhecida é a OASIS [5], esta arquitetura não será apresentada nesta dissertação, porém vale salientar sua importância para os entendimentos dos estudos.

A arquitetura de referencia discutida neste capítulo irá ajudar na identificação e especificação do desenvolvimento de uma arquitetura de desenvolvimento de sistemas em SOA. Esta visão lógica da arquitetura de referência SOA pode ser usadas indistintamente.

2.5. Benefícios da Arquitetura Orientada a Serviços

Os principais motivos para a arquitetura baseada em SOA são para facilitar o gerenciamento do crescimento dos sistemas corporativos de larga escala, para facilitar o

provisionamento da escalabilidade da Internet para uso por serviços e reduzir custos nas organizações para cooperação das organizações.

O valor do SOA é que ele oferece um paradigma escalável único para organizar grandes sistemas em rede que requerem interoperabilidade para realizar o valor inerente aos componentes individuais.

Certamente, o SOA é escalável por que ele faz a menor suposição possível sobre a rede e também minimiza qualquer suposição de confiança que são freqüentemente feitas em sistemas de escala menor.

Um arquiteto usando os princípios do SOA é mais bem equipado, conseqüentemente, para desenvolver sistemas que são escaláveis, evolutíveis e gerenciáveis. Pode ser fácil decidir como integrar as funcionalidades através dos limites proprietários. Por exemplo, uma grande organização adquire ou funde-se com outra e precisa determinar como integrar a infraestrutura de TI adquirida no portfólio global de TI existente.

Através desta habilidade inerente para escalar e evoluir, o SOA habilita um portfólio de TI que também é adaptável para diferentes necessidades de um domínio de problema específico ou arquitetura de processo. A infraestrutura que SOA incentiva é também a mais ágil e responsável que aquela construída em um número exponencial de pares de interfaces. Conseqüentemente, o SOA pode também oferecer uma sólida fundação para agilidade nos negócios e adaptabilidade.

3. ABORDAGEM POR PROCESSOS DE NEGÓCIO - (BPM)

O desenvolvimento do estudo de caso da ANVISA – Agência Nacional de Vigilância Sanitária que será apresentado no capítulo 3, foi orientado para uma abordagem por processo de negócio na utilização da arquitetura orientada a serviços, onde basicamente une gestão de negócios e tecnologia da informação com foco na otimização dos resultados das organizações através da melhoria dos processos de negócio. Foram estruturados métodos, técnicas e ferramentas para analisar, modelar, publicar, otimizar e controlar processos envolvendo recursos humanos, aplicações, documentos e outras fontes de informação.

Segundo o Guia para Gerenciamento de Processo de Negócio BPM CBOOK [1], Gerenciamento de Processos de Negócio (BPM) é uma abordagem disciplinada para identificar, desenhar, executar, documentar, medir, monitorar, controlar e melhorar processos de negócio automatizados ou não para alcançar os resultados pretendidos consistentes e alinhados com as metas estratégicas de uma organização. BPM envolve a definição deliberada, colaborativa e cada vez mais assistida por tecnologia, melhoria, inovação e gerenciamento de processos de negócio ponta-a-ponta que conduzem a resultados de negócios, criam valor e permitem que uma organização cumpra com seus objetivos de negócio com mais agilidade. BPM permite que uma organização alinhe seus processos de negócio à sua estratégia organizacional, conduzindo a um desempenho eficiente em toda a organização através de melhorias das atividades específicas de trabalho em um departamento, a organização como um todo ou entre organizações.

Um “processo” é um conjunto definido de atividades ou comportamentos executados por humanos ou máquinas, automatizados/manuais, para alcançar uma ou mais metas. Os processos são disparados por eventos específicos e apresentam um ou mais resultados que podem conduzir ao término do processo ou a transferência de controle para outro processo.

Processos são compostos por várias tarefas ou atividades inter-relacionadas que solucionam uma questão específica. No contexto do gerenciamento de processos de negócio, um “processo de negócio” é definido como um trabalho ponta-a-ponta que entrega valor aos interessados. A noção de trabalho ponta-a-ponta é chave, pois envolve todo o trabalho cruzando limites funcionais necessários para entregar valor.

Neste contexto a SOA fica apresentada como uma estratégia de negócio de uma organização, onde geralmente a disposição inicial oriunda na área de TI – Tecnologia da Informação, que orienta a construção de ativos de software, baseados na automação de **processos de negócio**.

A arquitetura orientada a serviços prevê que todos os sistemas de informação desenvolvidos na organização, podem e devem ser utilizadas por qualquer área de negócio que os necessite, por meio de aplicações que simplesmente possam “invocar” os serviços necessários à composição da mesma.

Além do ponto de vista técnico, a Arquitetura Orientada a Serviços também interage com determinadas políticas e conjuntos de "boas práticas" que pretendem criar um processo para facilitar a tarefa de encontrar, definir e gerenciar os serviços disponibilizados.

3.1. Ciclo de vida BPM

De acordo com o CBOK [1], a prática gerencial de BPM pode ser caracterizada como um ciclo de vida contínuo (processo) de atividades integradas de BPM. Enquanto que diversas variações de ciclos de vida BPM são reconhecidas¹, a maioria dos ciclos pode ser sumarizada por um conjunto gradual e interativo de atividades que incluem: (1) Planejamento; (2) Análise; (3) Desenho e Modelagem; (4) Implantação; (5) Monitoramento e Controle; e (6) Refinamento. À medida que os processos de negócio se movem através do ciclo de vida, são habilitados ou restringidos por uma variedade de fatores incluindo os quatro fatores primários de valores, crenças, liderança e cultura conforme ilustra a Figura 3.

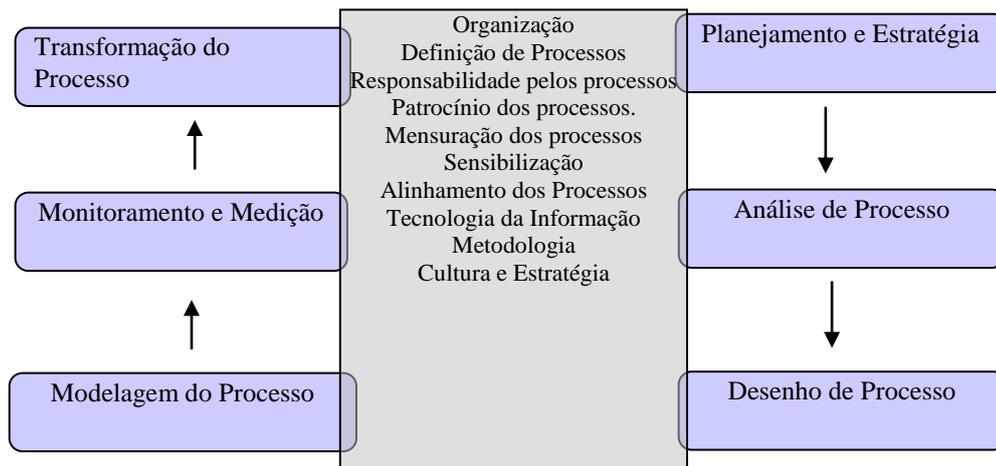


Figura 3 – Ciclo de Vida BPM. *ABPMP – CBOOK, [1]*.

Planejamento e Estratégia: Nesse modelo, o ciclo de vida BPM começa com o desenvolvimento de um plano e uma estratégia dirigida a processos para a organização. O plano inicia por um entendimento das estratégias e metas da organização desenhadas para assegurar uma proposição de valor atrativa para clientes. O plano fornece uma estrutura e o direcionamento para gerenciamento contínuo de processos centrados no cliente. Estabelece fundação para uma abordagem BPM holística para assegurar o alinhamento com a estratégia organizacional e a integração de estratégia, pessoas, processos e sistemas ao longo de seus limites funcionais. Essa fase estabelece a estratégia e o direcionamento do processo BPM. Também identifica papéis e responsabilidades organizacionais apropriadas de BPM, patrocínio executivo, metas, expectativas de medições de desempenho e metodologias. Caso se espere que atividades transformadoras significativas possam ocorrer, são analisadas mudanças organizacionais em estratégias.

Análise de Processo: A análise de processos de negócio incorpora várias metodologias com a finalidade de entender os atuais processos organizacionais no contexto das metas e objetivos desejados. A análise assimila informações oriundas de planos estratégicos, modelos de processo, medições de desempenho, mudanças no ambiente externo e outros fatores, a fim de entender completamente os processos de negócio no escopo da organização como um todo.

Desenho de Processo: As atividades de desenho de processo focam no desenho intencional e cuidadoso de como o trabalho ponta-a-ponta ocorre de modo a entregar valor aos clientes. A seqüência de atividades é documentada, incluindo o desenho do trabalho realizado, em que tempo, em qual local, por quais atores de processo e utilizando qual metodologia. O desenho define o que a organização quer que o processo seja e responde questões como: o quê, quando, onde, quem e como o trabalho ponta-a-ponta é realizado. Um importante componente de desenho é também assegurar que métricas e controles gerenciais apropriados estejam implementados para medição de desempenho e conformidade. Em um ciclo de vida iterativo de BPM, atividades iniciais de desenho podem focar na padronização ou automação de atividades atualmente realizadas *ad-hoc*, enquanto atividades de desenho mais maduras podem focar no redesenho ou transformação radical do processo, ou melhorias incrementais desenhadas para otimização.

Modelagem do Processo: Entender o processo envolve tipicamente a modelagem de processo e uma avaliação dos fatores ambientais que habilitam ou restringem o processo. Para organizações que estão menos maduras na prática BPM, pode ser a primeira vez que o processo de negócio ponta-a-ponta tenha sido documentado. Organizações mais maduras podem focar mais em fatores ambientais, nuances e exceções aos processos de negócio.

Monitoramento e medição: A contínua medição e monitoramento de processos de negócio fornecem a informação necessária para que gestores de processo ajustem recursos a fim de atingir objetivos dos processos. No contexto do ciclo BPM, medição e monitoramento também provêem informações-chave de desempenho de processo através de métricas relacionadas às metas e ao valor para a organização. A análise de informações de desempenho de processo pode resultar em atividades de melhoria, redesenho ou reengenharia.

Transformação de Processo: Transformação de processo implementa o resultado da análise iterativa e o ciclo de desenho. Trata desafios de gerenciamento de mudança organizacional e está orientado à melhoria contínua e otimização de processo. Nesse contexto, “processos otimizados” são aqueles que consistentemente atingem as metas predefinidas em termos de eficiência e efetividade. São gerenciados de tal forma que sejam capazes de responder a mudanças ambientais visando resultados consistentes.

3.2. Modelagem de processo de negócio

Modelagem de processo combina um conjunto de processos e habilidades que fornecem uma visão e entendimento do processo de negócio e habilita análise, desenho e medição de desempenho [1].

A Modelagem de Processo de Negócio é um conjunto de atividades envolvidas na criação de representações de um processo de negócio existente ou proposto. Modelagem de processo de negócio provê uma perspectiva ponta-a-ponta de processos primários, de suporte e gerenciamento de uma organização.

“Modelo” é uma representação simplificada que suporta o estudo e desenho de algum aspecto de algo, conceito ou atividade. Modelos podem ser matemáticos, gráficos, físicos, ou narrativos na sua forma ou alguma combinação desses elementos. Os modelos possuem ampla série de aplicações, que incluem:

- Organização (estruturação),
- Heurística (descoberta, aprendizado),
- Previsões (predições),
- Medição (quantificação),
- Explanação (ensino, demonstração),
- Verificação (experimentação, validação) e
- Controle (restrições, objetivos).

“Processo”, nesse contexto, significa um processo de negócio e pode ser expresso em vários níveis de detalhe, desde uma visão contextual altamente abstrata mostrando o processo dentro de seu ambiente, até uma visão operacional interna detalhada que pode ser simulada para avaliar várias características de seu desempenho ou comportamento. Devido aos processos de negócio ser realizados por pessoas interagindo uma com as outras, pessoas interagindo com sistemas de informação ou funções sistêmicas de informação completamente

automatizadas, um modelo de processo de negócio completamente desenvolvido representará tipicamente várias perspectivas servindo a diferentes propósitos.

Um modelo de processo pode conter um ou mais diagramas, informação objetos no diagrama, informação sobre relacionamento entre objetos, informação sobre relacionamento entre objetos e seu ambiente, e informação sobre como objetos representados se comportam ou desempenham.

Os processos possuem atributos e características que descrevem propriedades, comportamento, propósito, ou outros elementos de processo. Geralmente, atributos de processo são capturados em uma ferramenta a fim de organizar, analisar e gerenciar um portfólio de processos da organização. Dependendo das técnicas e capacidades das ferramentas utilizadas, existem vários atributos que podem ser modelados em fluxo de processo. Capturar essas características permite varias análises do desempenho do processo. Uma amostra de dados úteis que podem ser obtidos nos modelos de processo inclui os seguintes:

Insumos/Resultados	Padrões de chegada/Distribuições
Eventos/Resultados	Custos (indiretos e diretos)
Valor agregado	Regras de entrada
Papéis/Organizações	Regras de saída
Dados/Informações	Regras para decisões
Probabilidades	Regras de junção
Enfileiramento	Tempo de trabalho/Manuseio
Tempo de transmissão	Agrupamento
Tempo de espera	Servidores (número de pessoas disponíveis para desempenhar tarefas)

3.3. Benefícios da Modelagem

Em uma organização orientada a processos, modelos de processos são o principal meio para medir o desempenho versus padrões, determinando oportunidades para mudança e expressando o estado final desejado que preceda o esforço de mudança.

Modelos são, por definição, representações simplificadas que facilitam a compreensão do que está sendo estudada e a tomada de decisões sobre o assunto. Modelagem de processo é um mecanismo essencial para a compreensão, documentação, análise, desenho, automatização e medição de atividade de negócio, bem como medição de recursos que suportam a atividade

e as interações entre a atividade de negócio e seu ambiente. Como tal, tem uma vasta extensão de aplicação e, portanto, pode ser tratado a partir de uma variedade de pontos de lista ou necessidades dentro da organização. A seguir, alguns dos benefícios da modelagem:

- Modelos são relativamente rápidos, fáceis e baratos de completar.
- Modelos são fáceis de entender quando comparados a outras formas de documentação.
- Modelos fornecem uma linha-base para a medição.
- Modelos facilitam o processo de simulação e análise do impacto.
- Modelos nivelam vários padrões e um conjunto comum de técnicas.

3.4. Padrões de Modelagem e Notações

No estudo de caso apresentado nesta dissertação no capítulo 3, foi determinada a notação BPMN – *Business Process Modeling Notation* por melhor se adaptar a plataforma SOA Suíte da Oracle, porém a modelagem de processo de negócios pode ser realizada através de uma série de notações conforme apresentaremos a seguir.

***Business Modeling Notation* – BPMN**

BPMN é um padrão relativamente novo criado pelo *Business Process Management Initiative*, um consórcio de fornecedores de ferramentas no mercado BPM que concluiu fusão com *Object Management Group* (OMG), um grupo de definição de padrões para sistemas de informação. BPMN está rapidamente se tornando a maior e mais amplamente aceita notação de modelagem de processos de negócio no setor.

Fornece uma simbologia simples, mas robusta, para modelar todos os aspectos de processos de negócio.

Exemplos de fluxo de processo de negócio utilizando BPMN:

A figura 4 mostra um diagrama simples de processos de negócio.

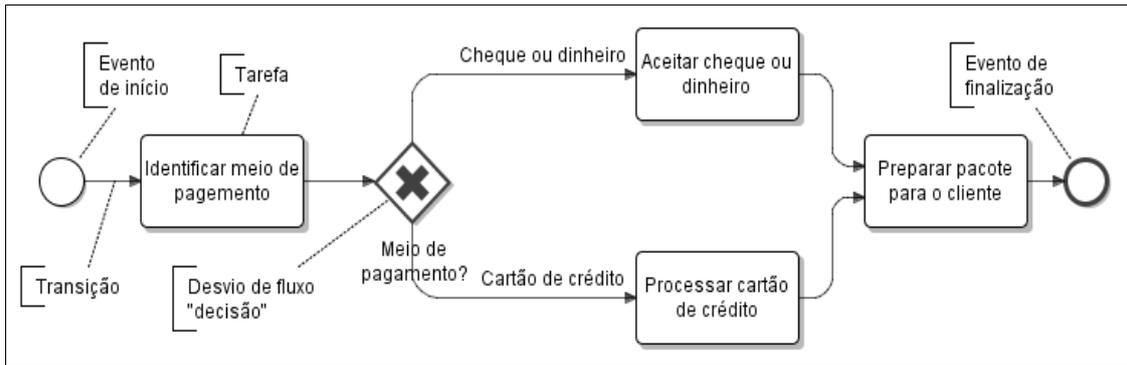


Figura 4 – Exemplo de Diagrama Processo de Negócio Simples. *Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento, ABPMP [1].*

A figura 5 mostra um diagrama mais complexo de um processo de negócio.

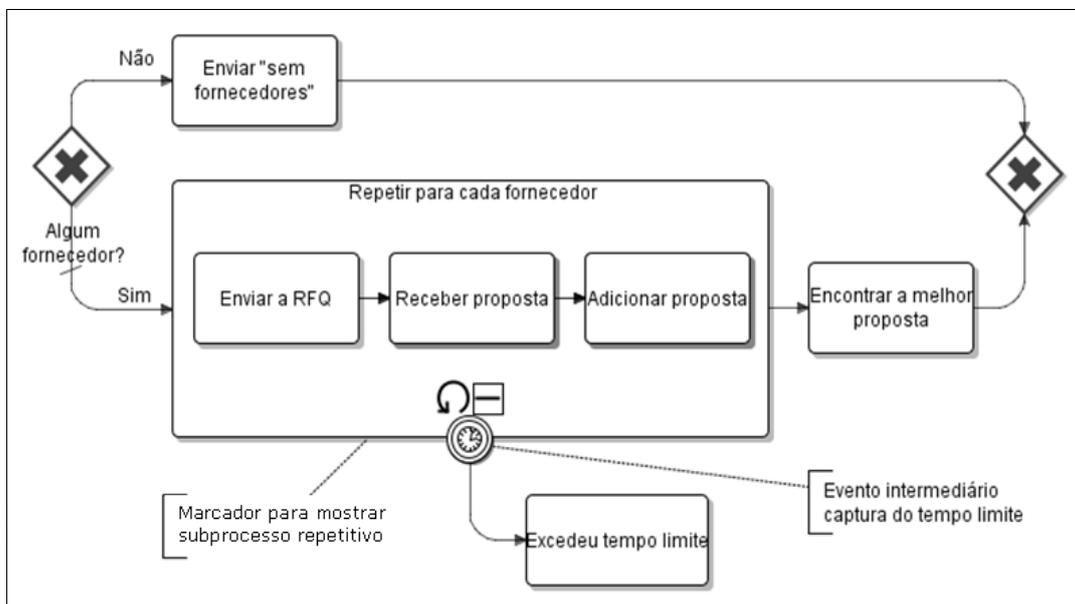


Figura 5 – Exemplo de Diagrama de Processo de Negócio mais detalhado e complexo. *Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento, ABPMP [1].*

A Figura 6 mostra um exemplo de diagrama de processo de negócio tradicional com raias.

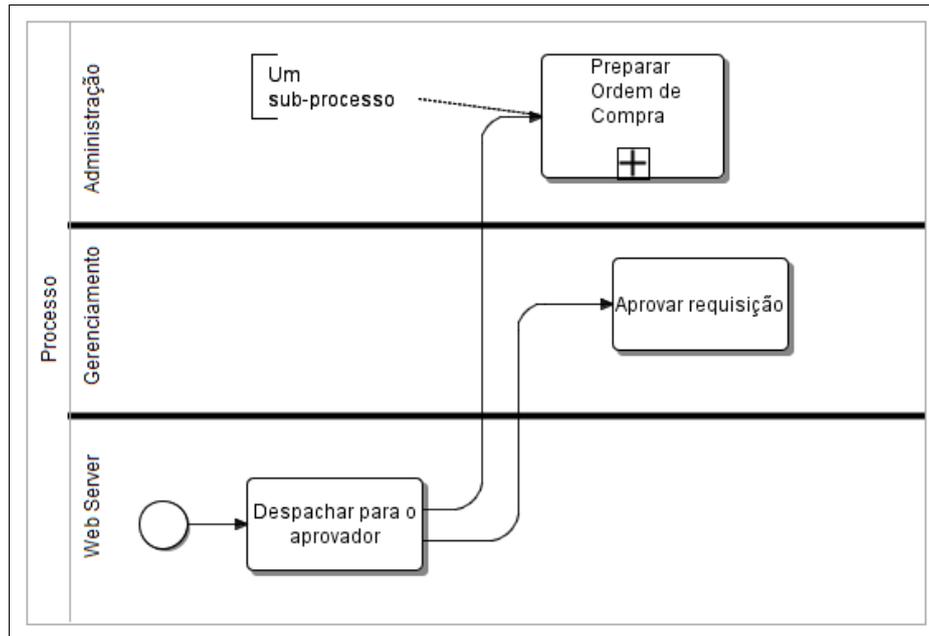


Figura 6 – Exemplo de Diagrama de Processo de Negócio tradicional com raias. *Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento, ABPMP [1].*

A Figura 7 mostra um diagrama de processo de negócio com participantes.

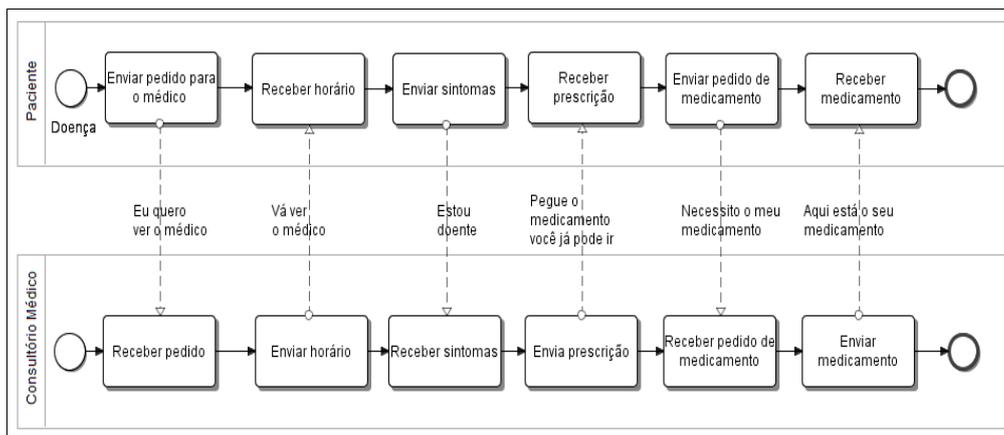


Figura 7 - Exemplo de diagrama de Processo de Negócio com participantes. *Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento, ABPMP [1].*

A figura 8 mostra um exemplo de diagrama de processo de negócio de alto nível.

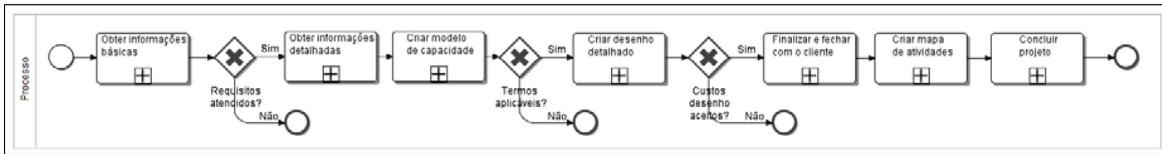


Figura 8 – Exemplo de Diagrama de Processo de Negócio em alto nível. *Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento, ABPMP [1].*

A figura 9 mostra um exemplo de diagrama de processo de negócio de baixo nível.

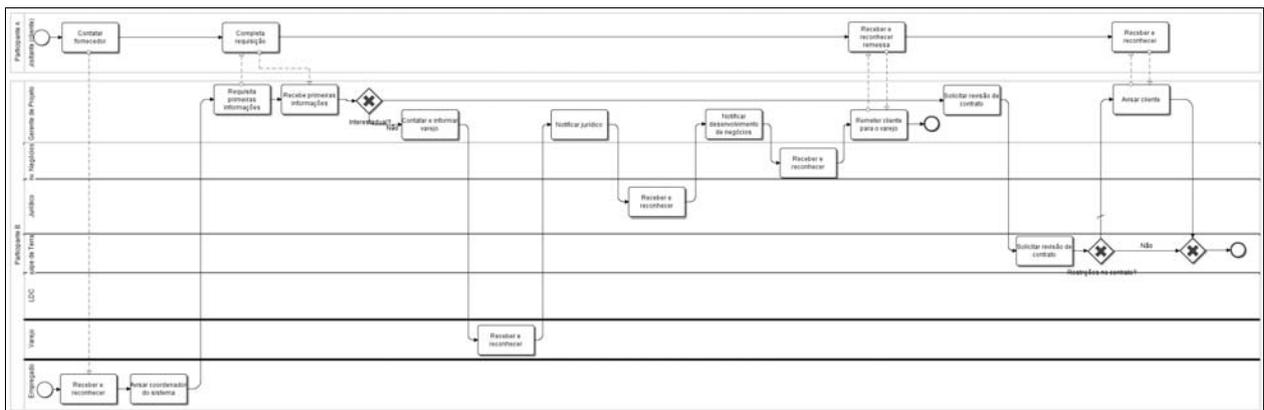


Figura 9 - Exemplo de Diagrama de Processo de Negócio em baixo nível. *Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento, ABPMP [1]*

Fluxogramas

Fluxogramas são amplamente utilizados e se baseiam em um conjunto simples de simbologia para tarefas, decisões e outros elementos primários de processo. Muito associados ao movimento do Gerenciamento da Qualidade Total (TQM) que surgiu no início da década de 1950 e ganhou popularidade ao longo da década de 1970 para a adoção de técnicas de modelagem de fluxo para mapear processos de negócio. A notação mais comum para fluxogramas foi aprovada como um padrão ANSI em 1970 para representar fluxos de sistemas. Outras notações de fluxogramas têm sido utilizadas por engenheiros industriais por décadas que empregam diferentes símbolos e layouts para mapeamentos industriais específicos visando descrever o fluxo de materiais, papéis e trabalho, colocação de máquinas, análise de egressos e ingressos em centros de distribuição etc.

Raias

Introduzida no livro *Managing Organizational Performance de Rummler and Brache*, raias são um complemento a “caixas e setas” do fluxo do processo de um fluxograma que representa como os fluxos de trabalho cruzam unidades organizacionais ou passam responsabilidade de um papel para outro. Realizado através da utilização de linhas dispostas horizontalmente ou verticalmente (raias), representando uma unidade organizacional, papel ou, em alguns casos, organização externa. Essas linhas lembram a faixa de marcação de competições de natação. Ao organizar o fluxo de atividades e tarefas entre essas linhas, é mais fácil visualizar *handoffs* no trabalho, um aspecto-chave da análise de processo de *Rummler-Brache* que se concentra em minimizar e gerenciar *handoffs*.

Event Process Chain (EPC)

Diagramas de *Event Process Chain* (EPC) são muito semelhantes aos diagramas de atividade quanto à adição de eventos ou resultados de tarefas. Um EPC é um gráfico ordenado de eventos e funções. Fornece vários conectores que permitem execução alternativa e paralela de processos. As tarefas (atividades) são seguidas de resultados (eventos) da tarefa, para o desenvolvimento de um modelo de processo bem detalhado. Além disso, é específico pelos usos de operadores lógicos, tais como OR, AND e XOR. Uma das forças do EPC está em sua simplicidade e facilidade de compreensão da notação. Isto torna EPC uma técnica aceita amplamente para desenhar processos de negócio. *Event Process Chain* é tipicamente utilizado para ajudar na transição de processos para automação ou simulação.

O método EPC foi desenvolvido no ambiente ARIS pelo Prof. *Wilhelm-August Scheer* o *Institut für Wirtschaftsinformatik* da *Universität des Saarlandes* no início da década de 1990.

É utilizado por muitas organizações para modelagem, análise e redesenho de processos de negócio.

Nem a sintaxe nem a semântica de EPC são bem definidos e variam de ferramenta para ferramenta. EPC requer uma semântica não local, de forma que o significado de qualquer porção do diagrama pode depender de outras porções arbitrariamente distantes.

Cadeia de valor

A notação de cadeia de valor é utilizada para demonstrar um simples fluxo contínuo da esquerda para direita dos processos que diretamente contribuem para produzir valor para os clientes da organização. Essa notação foi introduzida por Michael Porter em seu trabalho sobre estratégia corporativa e é tipicamente aplicado em planejamento estratégico empresarial. SCOR, o consórcio que definiu o *Supply Chain Reference Model*, utilizou uma notação de cadeia de valor para descrever os fluxos de processos de alto nível que provêm suporte ao gerenciamento da cadeia de valor e seus subprocessos. Mais recentemente, o Modelo de Referência de Cadeia de Valor tem sido proposto por outro grupo, VRM.

Unified Modeling Language (UML)

UML fornece um conjunto de nove ou mais padrões de técnicas de diagramação e notações principalmente para descrever requisitos de sistemas de informação. Embora a UML seja utilizado principalmente para análise e desenho de sistemas de informação, um número limitado de organizações também utiliza diagramas de atividade UML para modelagem de processos de negócio. A UML é mantida pelo *Object Management Group (OMG)*, organismo de estabelecimento de padrões para o campo dos sistemas de informação. Informações complementares sobre UML podem ser encontradas no site www.uml.org.

IDEF-0

O IDEF-0 é um padrão de processamento federal de informação dos EUA (FIPS – *Federal Information Processing Standard*), desenvolvido pela Força Aérea Americana para documentar processos de fabricação. É uma notação e técnica parte de uma metodologia para a definição dos processos de trabalho e sistemas de informação em ambientes de produção. Foi amplamente utilizada e disponível em muitas ferramentas de diagramação por muitos anos e agora é de domínio público.

LOVEM-E

LOVEM-E (*Line of Visibility Engineering Method – Enhanced*) é um conjunto de notação e técnica de modelagem desenvolvida como parte da metodologia da IBM para reengenharia de processos de negócio. O que torna LOVEM-E única é que adiciona

fluxogramas com raias, um conceito de cliente e natureza de trabalho colaborativo entre partes internas e externas, e suporte de sistemas de informação. BPMN também apresenta esses conceitos.

SIPOC

SIPOC (*Supplier, Input, Process, Output, and Customer*). É um estilo de documentação de processo utilizado no em Six Sigma. Não existe qualquer padrão ou conjunto de notação preferida e esta técnica pode ser satisfeita ao completar uma tabela com os elementos SIPOC.

Systems Dynamics

Mais do que apenas uma notação diferente, os modelos Systems Dynamics são diagramas “atividade na flecha” em vez de diagramas “atividade no nó” como a maioria de outras notações listadas. Modelos de sistemas dinâmicos são especialmente úteis no desenvolvimento de modelos de tipo de ciclo de vida dinâmicos que focam no desempenho do sistema geral de negócios e o impacto da mudança de chaves variáveis principais que afetam o desempenho global. Esses são mais freqüentemente utilizados para modelar toda a organização ou linha de negócio em vez de modelos tipo fluxo de trabalho de mais baixo nível. Modelos Systems Dynamics são freqüentemente utilizados para descrever a “arquitetura” de negócio empresarial a partir de uma perspectiva de comportamento dinâmico e não uma perspectiva estrutural estática.

Value Stream Mapping

Value Stream Mapping é uma técnica utilizada no *Lean Manufacturing*. Não devendo ser confundida com notação de cadeia de valor, *Value Stream Mapping* expressa o ambiente físico e fluxo de materiais e produtos em um ambiente de manufatura. Na Toyota, onde a técnica foi originada, é conhecida como “Mapeamento de Fluxo de Informação e Material”.

Há várias abordagens para modelagem de processo: de cima para baixo (*top-down*), meio para cima ou para baixo (*middle-out*), ou de baixo para cima (*bottom-up*). Alguns métodos de desenvolvimento de modelo de processo pedem um enfoque de processo iterativo

onde é esperado que diversas passagens sucessivas para o desenvolvimento do modelo sejam necessárias. A abordagem utilizada varia, dependendo do propósito e escopo do esforço.

Tradicionalmente, modelos de processo foram geralmente criados com a finalidade de melhorar estreitamento de funções dentro de um único departamento ou operação. Muitas vezes, o processo não tem sido documentado e o primeiro passo é tentar descobrir o que realmente está ocorrendo. Abordagens de baixo para cima, centradas em atividade muito detalhada e fluxo de trabalho orientado a tarefa funcionam melhor para esses tipos de projetos.

No estudo de caso da ANVISA apresentado no capítulo três, a abordagem para a modelagem de processo para automação foi a *Top-down*, tendo como escopo o processo de negocio de Registro de Medicamentos.

4. ESTUDO DE CASO - AGÊNCIA NACIONAL DE VIGILÂNCIA SANITÁRIA – ANVISA.

A Universidade de Brasília – UnB, por intermédio do seu Núcleo de Tecnologia da Informação – NTI, desenvolveu projeto técnico e científico, com intuito de colaborar com a Agência Nacional de Vigilância Sanitária na realização de um conjunto articulado de ações de prospecção, identificação, definição e implantação de ações estratégicas para a área de TI, com foco no desenvolvimento de uma arquitetura de desenvolvimento de software orientada a serviço para a automação do processo de Registro de Medicamentos.

A ANVISA estruturou um ambiente de Desenvolvimento de Sistemas aderente a arquitetura orientada a serviços. Este ambiente foi criado e sustentado com base nas definições descritas neste estudo, que viabilizou a implantação da arquitetura em aderência às necessidades do ambiente de Tecnologia da Informação da ANVISA.

Este capítulo fornece uma visão abrangente da arquitetura de referência para a plataforma tecnológica de desenvolvimento de sistemas de informação em SOA da ANVISA, usando diversas visões de arquitetura para representar seus diferentes aspectos. Os estudos orientados para o desenvolvimento desta dissertação provêm uma aplicação compreensiva da arquitetura de referência para a plataforma tecnológica para a ANVISA usando conceitos arquiteturais para compreender seus diferentes aspectos e é voltado para os seguintes públicos do projeto:

- Usuários da Plataforma;
- Patrocinadores do Projeto;
- Integradoras de Sistemas;
- Projetistas de sistemas;
- Desenvolvedores de Sistemas;
- Mantenedores da Plataforma;
- Analistas de Negócio.

4.1. Visão geral da arquitetura

A visão geral da arquitetura tem por objetivo demonstrar como as diversas tecnologias utilizadas em sua construção são integradas. Ela foi projetada para ter como núcleo uma ferramenta de BPMS (*Business Process Management Software*), no qual todos os processos de negócio serão mapeados, e um Barramento de Serviços (*Enterprise Service Bus – ESB*). Este barramento permite a implantação do conceito de Arquitetura Orientada a Serviços, facilitando a integração padronizada com as demais tecnologias presentes na arquitetura e com os sistemas finalísticos instalados em todas as unidades da ANVISA. Essa arquitetura, através do ESB, fornece uma interface comum que possibilita a construção de pontos de integração padronizados entre as diversas aplicações.

A visão estratégica para o desenvolvimento de sistemas de informação elaborada pela GESIS/GGTIN/ANVISA – Gerência de Sistemas da Gerência Geral de Tecnologia da Informação da ANVISA, orienta que a construção dos ativos de software seja desenvolvida com base na disponibilização de serviços (*web services*).

Os serviços, neste caso, podem ser entendidos como softwares que foram construídos para poderem se conectar/integrar a outros softwares, possibilitando a reutilização dos mesmos por todos os processos de negócio da Agência. Neste contexto, podemos definir “*web services*” como a descrição de uma ou mais operações que usam (múltiplas) mensagens, bem-definidas, auto-suficientes e independentes de contexto, para trocar dados entre um fornecedor e um consumidor a partir de um contrato previamente estabelecido.

A arquitetura prevê que todos os softwares desenvolvidos na Agência, possam ser utilizados por qualquer processo de negócio que os necessite, por meio de aplicações que simplesmente possam “invocar” os serviços necessários à composição da mesma.

A arquitetura possui ainda em sua essência uma flexibilidade necessária para que TI responda às necessidades do negócio de forma ágil, utilizando um conjunto de melhores práticas adotadas pelo mercado, e que permitam uma maior governança de todos os ativos de software disponibilizados pela GESIS/GGTIN/ANVISA.

A adoção desta arquitetura visa solucionar dificuldades encontradas no modelo tradicional de desenvolvimento de software, sendo que as principais são:

- Dificuldade de manutenção e controle;
- Aumento de integrações sem o controle adequado proporcionado pela evolução das aplicações;
- Falta de governança;
- Inexistência de reaproveitamento, gerando um elevado índice de re-trabalho;
- Ineficiência;
- Dificuldade de monitorar e tratar erros – impacto direto no usuário final;
- Ciclos de desenvolvimento e homologação complexos e pouco eficazes;

A figura 10 apresenta o cenário atual da estrutura de integração de sistemas no âmbito da ANVISA.

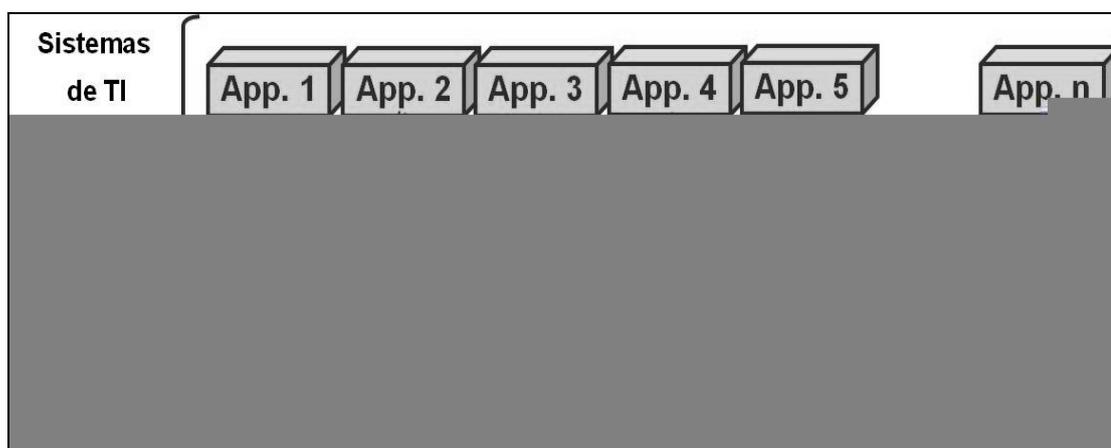


Figura 10 – Cenário atual de Integração de Sistemas na ANVISA

Em contrapartida a essa realidade, a arquitetura apresentará diretrizes aderentes às melhores práticas da construção de um ambiente baseado em serviços. A figura 11 mostra o cenário típico da arquitetura de referência SOA adotada para ANVISA.

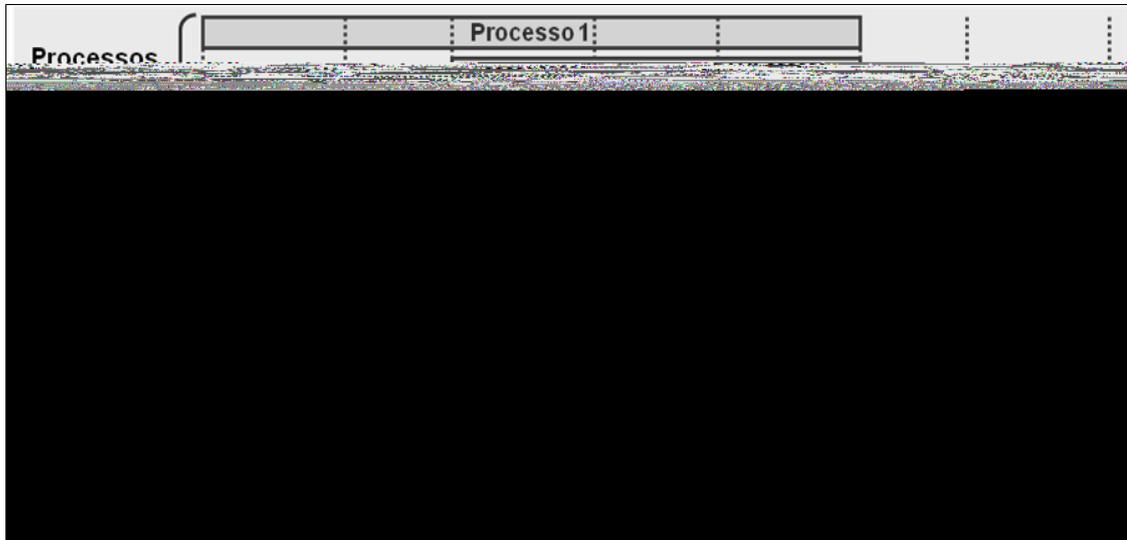


Figura 11 – Cenário Típico de SOA na ANVISA

Os benefícios esperados com a adoção da arquitetura mostrada na Figura 12 são:

- Evolução facilitada
- Redução dos custos de atualização e manutenção;
- Centralização da gestão de aplicações integradas;
- Plataforma expansível e portátil;
- Proteção e potencialização dos investimentos realizados;
- Maior capacidade de integrar novas necessidades de negócio;
- Visão compartilhada dos processos entre áreas de TI e negócios;
- Redução na complexidade na integração de aplicações;
- Facilita a abordagem da complexidade do negócio;

- Desenvolvimento e integração de aplicações;
- Segurança, monitoração e controle de acesso a dados;
- Tecnologia como agente facilitador ao crescimento do negócio;

4.2. Arquitetura para automação de processos – ANVISA

A arquitetura de referência será dividida em camadas com responsabilidades bem definidas, solicitando serviços a sua camada inferior obedecendo às regras pré-existentes. A Figura 12 apresenta graficamente a visão lógica desta arquitetura:

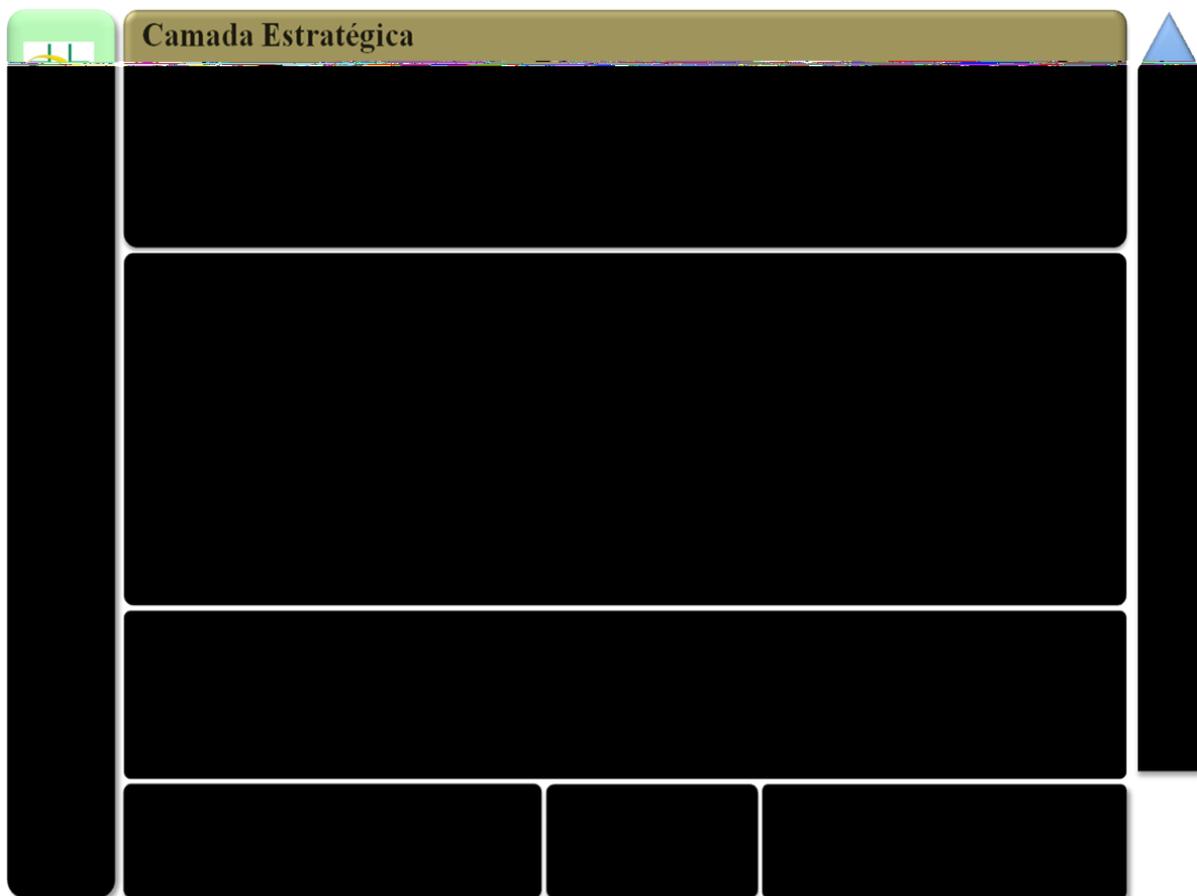


Figura 12 – Visão Lógica da SOA - ANVISA

Esta estruturação tem como objetivo principal organizar os serviços em grupos, ou camadas, que representem similaridade em seus objetivos, níveis de abstração e áreas de atuação.

Camada Estratégica – esta camada é subdivida em três camadas assim definidas:

Camada de Apresentação

A Camada de Apresentação é responsável por receber as requisições dos usuários através de uma interface web, delegar funções às camadas inferiores (negócio e serviços) e responder pela sua requisição. Esta camada comunica-se com a camada de Processos de Negócio e com a Camada de Serviços.

Para o desenvolvimento de interfaces da camada de apresentação serão utilizados os padrões JSF e *Portlets*. A integração entre a camada de apresentação e a camada de serviços deverá ser implementada preferencialmente seguindo os padrões de projeto:

Business Delegate

Padrão de projeto que reduz o acoplamento entre o cliente e a camada de negócios, escondendo a complexidade de acesso aos componentes de negócio.

A figura 13 demonstra um exemplo de fluxo referente a *business delegate*.

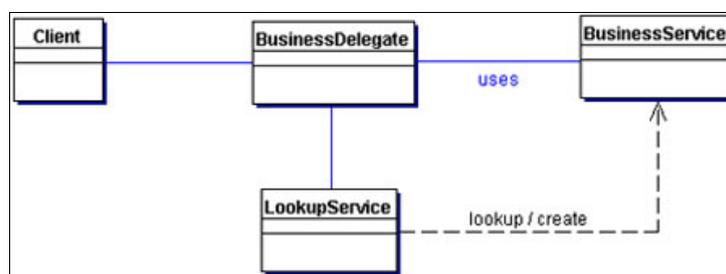


Figura 13 – Padrão de Projetos *Business delegate*.

Service Locator

Padrão de projeto que encapsula a complexidade de localização e criação de serviços de negócio e localiza os objetos de definição dos serviços, provendo um ponto único de controle e, em determinados casos, melhorando a eficiência através de mecanismos de cachê.

A figura 14 mostra um exemplo de padrão de projeto referente à Localização de Serviços.

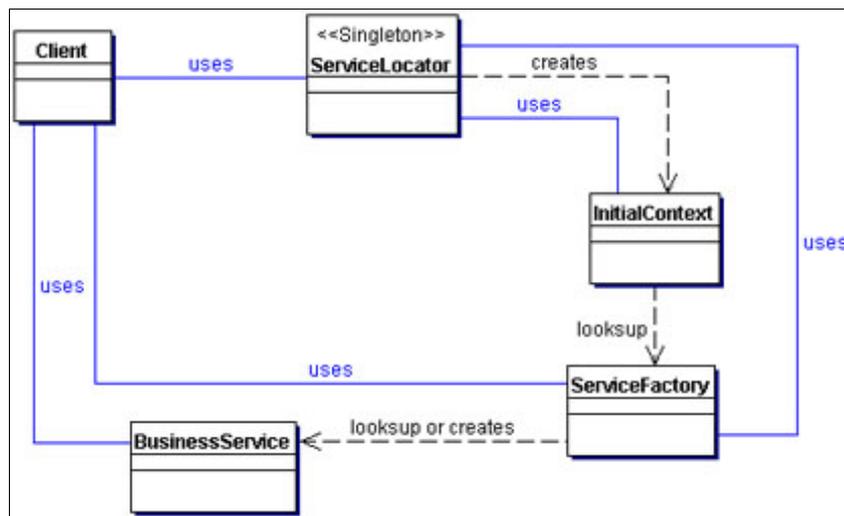


Figura 14 – Padrão de projetos para Localização de Serviços

Painel Gerencial

O Painel Gerencial é responsável por disponibilizar ferramentas de gerenciamento do comportamento das aplicações desenvolvidas utilizando a arquitetura de referência. Neste painel serão apresentados *dashboards*, que mostram através de gráficos em tempo real, os dados resultantes dos indicadores colhidos e processados durante a execução dos processos de negócio.

Processos de Negócio

A Camada de Processos de Negócio é responsável por representar os processos de negócio da organização relacionando os procedimentos necessários para que todas as atividades pertinentes ao negócio sejam executadas de acordo com as regras vigentes.

Neste contexto, os processos de negócio da ANVISA deverão ser modelados na ferramenta Oracle BPA, seguindo a notação de modelagem de processos BPMN e utilizando os filtros e modelos definidos pela Metodologia de Modelagem de Processos da ANVISA.

A comunicação desta camada com as demais camadas da arquitetura ocorre por meio da geração da estrutura de um processo BPEL armazenado na camada de Serviços e que fará a orquestração dos serviços disponibilizados nas camadas inferiores e que permitirão a automação total ou parcial de um processo de negócio.

Camada de Serviços - Esta camada é responsável pela disponibilização de serviços para as aplicações garantindo sua estabilidade em execução. A seguir descrevem cada uma das suas sete subcamadas.

Serviços de Negócio

Nesta camada estão disponíveis os serviços que executam uma função de negócio. Os serviços de negócio podem ter granularidades variadas, indo desde serviços de uso geral discretos aos serviços menos especializados. Os serviços menores tendem a ser mais específicos em sua finalidade e a fornecer um potencial mais elevado para reutilização. Os serviços menos especializados são frequentemente grandes composições de serviços menores. Este tipo de serviço permite que sejam implementados os processos de negócio utilizando o conceito de orquestração de serviços.

Serviços de Decisão

Nesta camada são disponibilizados os serviços que executam regras que fornecem decisões de negócio. Os serviços de decisão fornecem, geralmente, respostas sim/não às perguntas complexas, ou suportam regras exteriorizadas frequentemente em mudança. Os serviços de decisão geralmente são compostos com outros serviços e são de pequenos a médio no tamanho.

Serviços de Dados Canônicos

Nesta camada estão disponibilizados os serviços que normalizam os serviços disponibilizados na Camada de Persistência para prover uma consistente representação de dados e seus relacionamentos baseados nos requisitos definidos para as camadas superiores.

Sua principal responsabilidade é definir um dicionário de dados corporativo que resulta na definição de serviços que conheçam as necessidades e variedades dos consumidores, reduzindo assim a duplicação de serviços.

A Camada de Dados Canônicos estabelece esse formato comum sob uma camada de dados que pode ser usado para orientar a análise e design da arquitetura de serviços, garantido um alinhamento rigoroso das definições dos dados em torno da modelagem de domínios.

Serviços de Persistência

Os serviços de persistência disponibilizados nesta camada são responsáveis por todo acesso direto à camada de dados, utilizando os *drivers* de conexão, para cada instância de banco de dados.

O serviço desta camada representa uma origem de dados específica. Seus esquemas são conduzidos eficazmente pelas estruturas de dados nativas, e devem possuir as seguintes características:

- Não selecione e nem restrinja dados;
- Não transforme e nem normalize dados;
- Não agrupe outros serviços;
- Não agregue dados.
- Estes serviços devem possuir as seguintes orientações para alcançar um nível ótimo de granularidade:
 - Deve ser possível descrever os serviços em termos de função, informação, objetivos e regras;
 - Deve ser auto-suficiente o máximo possível.
- Deve ser atômico, isso significa que a chamada do serviço deve produzir um resultado bem-sucedido ou ser inócua;

- Deve ser consistente, deixando o estado do fornecedor do serviço válido e consistente;
- Deve ser isolado, não sendo influenciado por outras chamadas de serviço que simultaneamente, retornando sempre dados consistentes;
- Deve ser durável, garantido que o efeito após a chamada do serviço seja persistente.

Serviços de Integração

Os serviços disponibilizados nesta camada fornecem o acesso às aplicações e aos dados existentes da Agência, de modo que possam ser utilizados pelo negócio, proporcionando um meio de integração entre sistemas legados e sistemas desenvolvidos seguindo a atual arquitetura. Maiores detalhes sobre a implementação destes serviços estão definidos no documento de Arquitetura de Integração da ANVISA.

Serviços de Infraestrutura

Os serviços disponibilizados nessa na camada de Infraestrutura fornecem os mecanismos que permitam a autenticação, autorização, recuperação da política de segurança, aplicação da política de segurança, descoberta da credencial de segurança, e que serão utilizados por todas as aplicações desenvolvidas seguindo a presente arquitetura.

Camada de Infraestrutura fornece a segurança, a mediação, acompanhamento, gerenciamento de políticas, serviços de governança para a aplicação e camada de mensagens. Ela garante que as políticas apropriadas sejam aplicadas pelos serviços que são invocados, e que as aplicações mandem mensagem em conformidade com as políticas que os serviços devem obedecer.

Adaptadores

Os serviços disponibilizados nesta camada são responsáveis por fazer a comunicação entre a camada de serviços e os diversos tipos de fontes de dados e sistemas legados da organização.

Granularidade

A arquitetura está representada por escalas de representação de serviços, variando de serviços concretos (com maior granularidade) até serviços abstratos com grau de composição mais elevados. A escala de abstração de serviços (mostrado na Figura 6-8) começa em seu nível inferior os serviços mais concretos que indicam a representação real da informação, e em seu nível superior com artefatos que já realizaram transformações e são altamente abstratos.

Os serviços com o nível de abstração mais alto não conhecem detalhes sobre os serviços implementados nas camadas inferiores que os sustentam. Cada camada de abstração conhece sua camada de abstração inferior e este conhecimento deve ser unidirecional. Os componentes das camadas mais baixas possuem uma quantidade grande de detalhes de baixo nível. Por outro lado, os componentes das camadas mais altas sabem somente o que devem fazer, mas têm pouca ou nenhuma informação sobre como será o processamento feito nas camadas inferiores.

A Figura 15 mostra a integração de entre cada uma de suas subcamadas com as demais camadas da arquitetura.

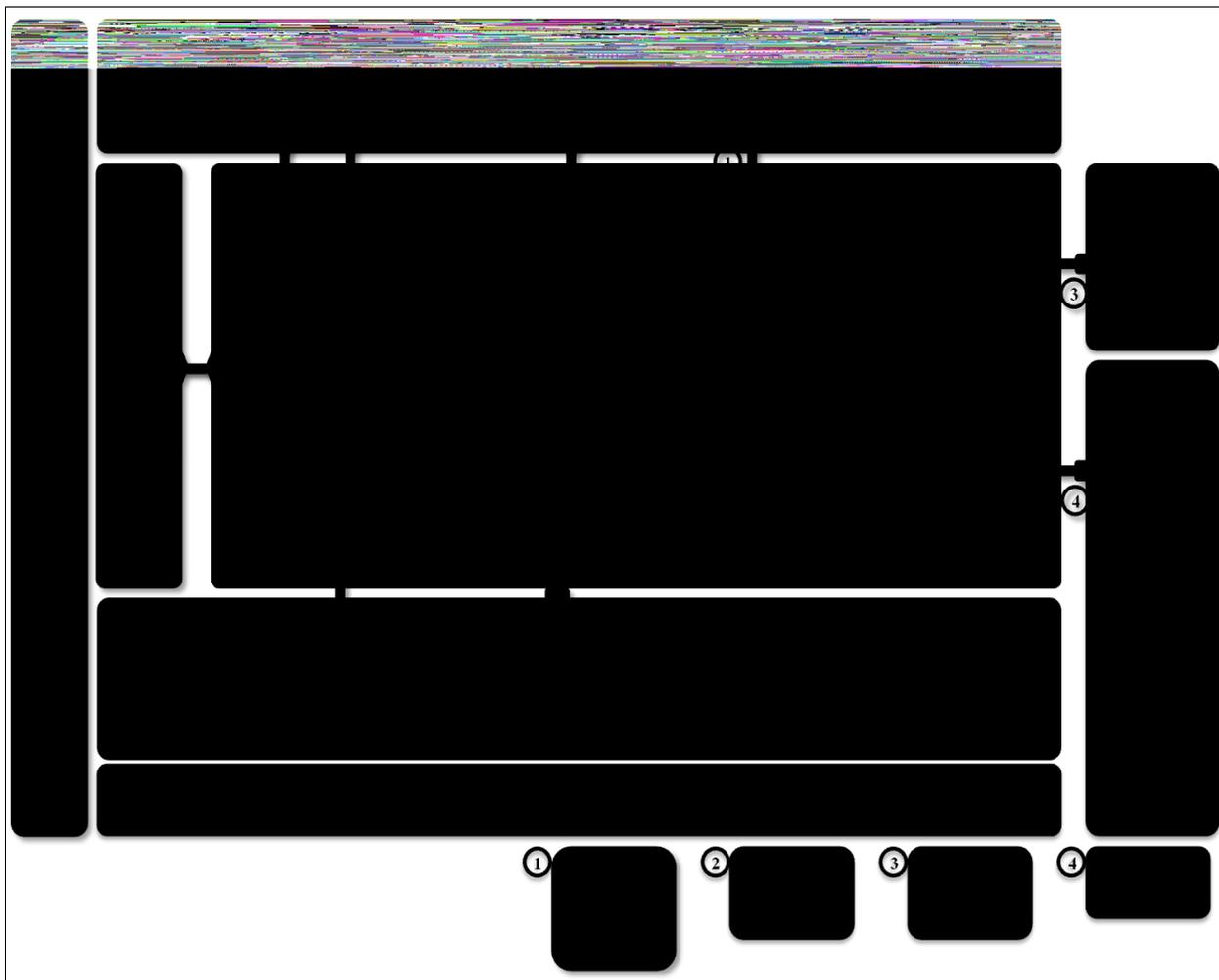


Figura 15 – Integração entre camada e subcamadas da arquitetura

O resultado do emprego desta abordagem é a localização e a densidade dos serviços na arquitetura. Nas camadas mais baixas, existirá naturalmente um número muito grande de serviços quando comparado com as camadas mais altas.

O protocolo padrão para invocação de serviços é o SOAP usando WSDL(1) para definir suas interfaces. Para aplicações JEE deve-se utilizar JCA(2) e JMS(2) para troca de mensagens assíncronas. A comunicação dos adaptadores(3) com os sistemas legados será definida de acordo com o padrão definido pelo fornecedor do adaptador.

A figura acima representa o barramento, chamado de *Enterprise Service Bus* – ESB, que permite a interoperabilidade e acessos a todos os serviços publicados. Este barramento tem a responsabilidade de atender às seguintes tarefas:

- Prover conectividade entre os serviços e as aplicações clientes;
- Transformação de Dados;
- Roteamento inteligente;
- Tolerância a falhas;
- Segurança;
- Confiabilidade;
- Gerenciamento de Serviços, e;
- Monitoramento e log de chamadas.

A figura 16 ilustra a comunicação entre dois serviços utilizando o protocolo SOAP.

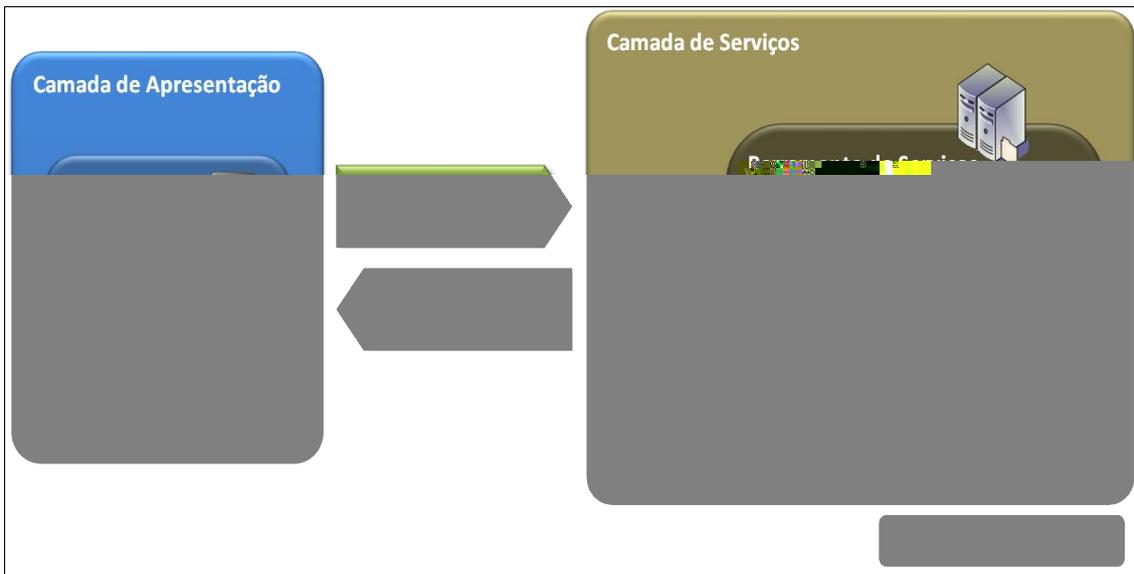


Figura 16 – Comunicação entre dois Serviços

Na Figura 17, fica demonstrado uma chamada de serviço síncrona bem-sucedida que inclui uma mensagem de requisição do consumidor do serviço para o fornecedor e uma mensagem de resposta do fornecedor de volta para o consumidor do serviço. Este diagrama evidencia o protocolo de comunicação SOAP com WSDL para a invocação do serviço e sua respectiva resposta.

O diagrama da Figura 17 detalha a sequência de eventos que deve acontecer.

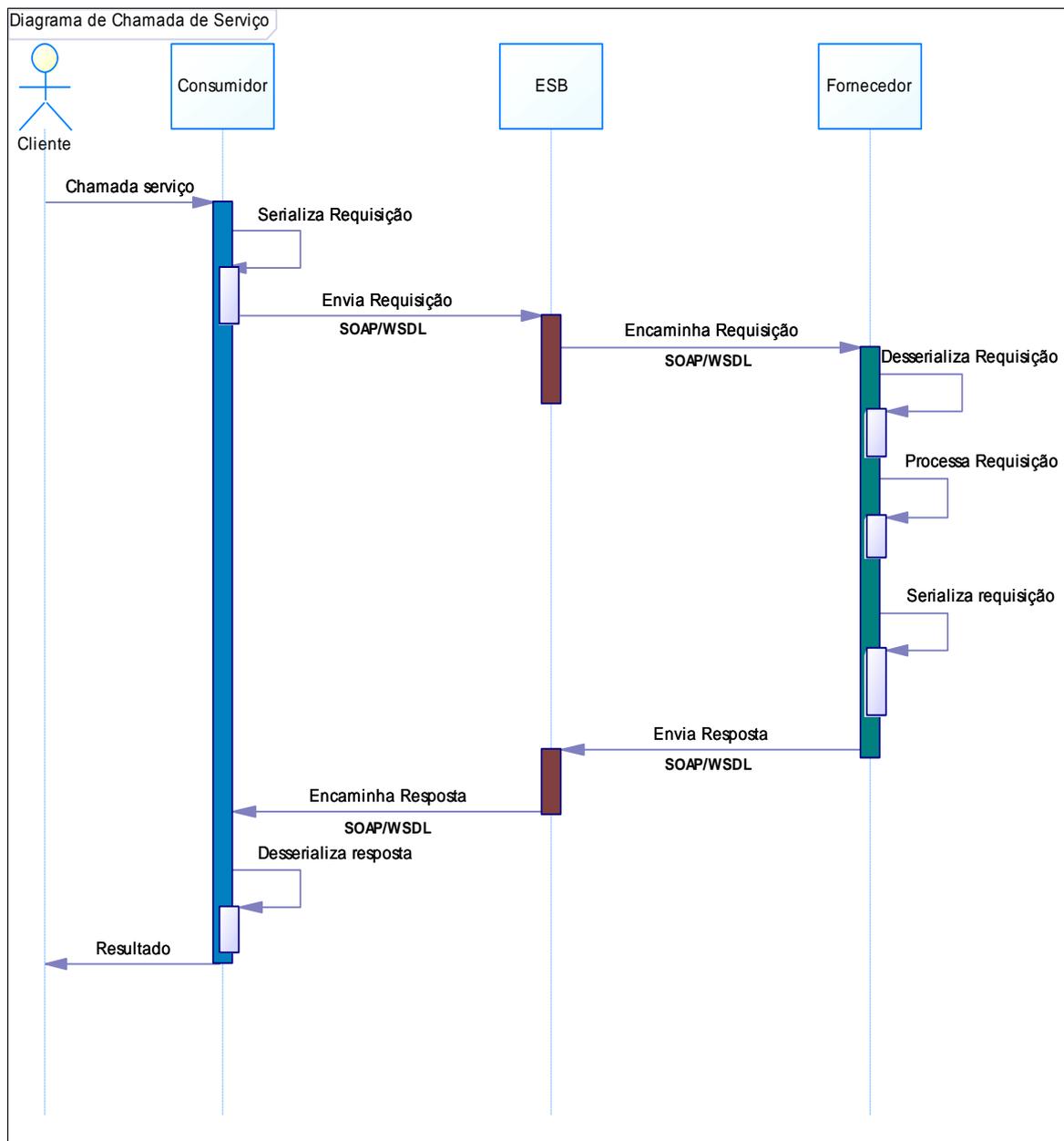


Figura 17 – Diagrama de Sequência de Chamada de Serviço

Infraestrutura - Esta camada é à base dos sistemas transacionais (que suportam a operação) da ANVISA onde estão instaladas as ferramentas que darão suporte a toda arquitetura e é composto pelos componentes descritos nos tópicos a seguir.

Catálogo de Serviços

Responsável por catalogar os serviços criados no ambiente da ANVISA utilizando o protocolo UDDI, que especifica um método de publicar e descobrir serviços. O arquiteto SOA é responsável pela manutenção dos serviços no repositório central de serviços da ANVISA, assim como estabelecer a rastreabilidade entre os mesmos.

Todo *web service* implementado e disponibilizado na arquitetura deve ser catalogado de acordo sua área de negócio e registrado nesse catálogo. Isso permite futuras pesquisas para a reutilização do serviço criado diminuindo o retrabalho na criação de serviços equivalentes.

A definição dos atributos obrigatórios para cada serviço publicado e o fluxo de publicação destes serviços está definida na Metodologia de Desenvolvimento da ANVISA.

Gerência de Regras de Negócio

Possibilita a criação e manutenção de regras de negócio separadamente dos códigos de aplicação. As regras são externalizadas de tal maneira que a alteração das mesmas pode ser feitas por usuários de negócio, em tempo real, não impactando no processo de negócio em si.

Este componente se encaixa na plataforma SOA principalmente na definição de Pontos de Decisão em processos BPEL, os quais podem sofrer alterações na regra de negócio sem causar impacto no fluxo dos processos em produção.

A utilização deste componente apenas não é recomendada em regras de negócio altamente acopladas ao modelo de negócio em questão, que não são passíveis de reaproveitamento por outros serviços, pois isto acarretará em mais uma camada de abstração e aumento da complexidade e processamento do serviço. Estes tipos de regras devem ser implementadas diretamente no processo de execução.

Motor de Processos

Responsável pelo mapeamento entre os processos de negócio mapeados e modelados na ferramenta de modelagem de processos e sua automação. Deverão ser exportados para a linguagem BPEL e implementados/codificados.

O comportamento do processo deverá ser definido previamente e que poderá ser síncrono ou assíncrono, além da definição de todos os atributos de entrada e saída. Estas definições deverão ser implementadas utilizando padrão de codificação XSD (*XML Schema Definition*).

Os serviços implementados devem usar preferencialmente o protocolo SOAP visto que pode haver necessidade de não se utilizar o protocolo HTTP como meio de comunicação, no caso MQ ou JMS, inviabilizando a utilização do protocolo REST.

Os processos são compostos por atividades automatizadas e/ou atividades que requerem intervenção humana. Neste contexto, deverão ser implementados nos processos sensores responsáveis por alimentar os indicadores a serem capturados pela ferramenta de monitoração de processos.

A ferramenta de desenvolvimento será utilizada para criação de cenários de testes para os processos modelados e automatizados.

Gerência de Serviços

Responsável por prover a segurança (autenticação, autorização e criptografia) de todo o *web services* criados na arquitetura e monitorar suas atividades executadas.

Deve-se utilizar o padrão *WS-Security* para autenticação e segurança no nível de mensagem, incluindo criptografia e assinaturas.

As políticas de utilização e controle de acesso dos serviços estão definidas na Metodologia de Desenvolvimento da ANVISA. As políticas de acesso aos serviços devem estar alinhadas e aderentes a política de Segurança de Informações da ANVISA. Esta relação garantirá que apenas pessoas e/ou sistemas autorizados tenham acesso a informações restritas.

A ferramenta que irá gerenciar *web services* criados dentro da plataforma de desenvolvimento da ANVISA é o Oracle *Web Services Manager*.

Business Activity Monitoring - BAM

Responsável por coletar, armazenar e processar os indicadores captados pelos sensores implementados nos processos e bancos de dados. Estes indicadores são processados pelo *Active Data Cache*, armazenados no banco de dados do BAM e enviados em tempo real para geração de gráficos que compõem os *Dashboards*.



Figura 18 – Dashboard do BAM

Cabe um alerta: é importante que os indicadores de desempenho sejam definidos no momento da modelagem do processo de negócio, o que evita retrabalho e eleva a assertividade das informações neles contidas.

A ferramenta que irá os indicadores de atividade nos processos mapeados da ANVISA é o Oracle BAM.

Sistemas Legados

Servidores de sistemas computacionais da ANVISA que utilizam plataforma diferente da descrita neste documento. Por falta de documentação e/ou pela saída do pessoal técnico que participou originalmente no seu desenvolvimento os sistemas legados podem apresentar problemas como: dificuldade de compreensão das regras de negócio neles implementadas, desconhecimento das razões que levaram a determinadas decisões, problemas na estruturação dos módulos de código, miscelânea de estilos de programação, obsolescência das ferramentas de desenvolvimento e impossibilidade de reaproveitamento dos equipamentos nos quais são executados para execução de softwares mais atuais.

Estes sistemas serão integrados com o restante da plataforma com a criação de *web services* e/ou adaptadores publicados no Barramento de Serviços, como descrito no documento de Arquitetura de Integração.

Servidor de Aplicação

O servidor de aplicações provê o repositório corporativo de aplicações e serviços Web baseados em padrões de mercado. Esse conjunto de funcionalidades, que estão presentes na arquitetura da ANVISA, faz parte do Oracle Application Server.

Repositório de Dados

O Repositório de Dados tem o papel de representar em forma de objetos de dados, um conjunto de registros de banco de dados relacional, assim como as dimensões criadas para o *data-warehouse* corporativas. Serão criadas duas visões de dados para a plataforma, sendo elas:

- Dados Operacionais – Repositório onde serão armazenadas todas as informações operacionais da plataforma, incluindo a base de conhecimento, base de autenticação dos usuários, documentos e regras de negócio.
- Dados Analíticos – Repositório onde serão armazenadas todas as informações analíticas geradas a partir dos dados operacionais, incluindo indicadores, *datamarts* e *datawarehouse*.

4.3. Benefícios Alcançados

A compreensão dos benefícios da adoção de SOA é observada através de dois pontos de vista um ponto de vista: de negócio e ponto de vista tecnológico.

Ponto de vista de negócio

Para compreender os benefícios, do ponto de vista de negócio, trazidos pela adoção de SOA, é necessário que sejam definidas métricas relevantes aos objetivos estratégicos de negócio, essas métricas servem para ajudar a avaliar o desempenho, normalmente em termos de fazer progresso em direção ao conjunto de objetivos estratégicos.

A definição de métricas é necessária, pois demonstram as competências e valor agregado para o negócio e, ainda, provê as justificativas para os investimentos dispensados para adoção de SOA.

Os benefícios da adoção de SOA do ponto de vista do negócio são conclusivos desde que haja o alinhamento com os objetivos estratégicos do negócio.

Nesse sentido são propostas as seguintes métricas para avaliação de benefícios no ponto de vista para negócio, conforme apresentadas na tabela 3.

Benefícios	Métricas
Reduzir o tempo de entrega de Sistemas de Informação	Número de Capacidade de Negócio definidas.
	Número de Serviços implementados por capacidade de negócio.
Aumentar a facilidade de acesso	Número de capacidades de negócio disponibilizado pelos serviços através do barramento de serviços.
	Número de Usuários por Serviço disponibilizado.
Melhorar a consistência de dados da informação	Número de Serviços utilizando modelo canônico de dados e bases de dados higienizadas.
Reduzir o custo total de operação e manutenção dos sistemas	Número de Serviços Reutilizáveis.
	Número de Serviços Reutilizados.

Tabela 2 – Métricas da Adoção de SOA

Na prática, esse conjunto de métricas propostas, relacionadas aos benefícios, deverá ser aplicado, em uma análise posterior, em tempo de execução de um projeto de SOA.

Ponto de vista tecnológico

Identificar os benefícios do ponto de vista tecnológico da adoção de SOA significa traduzir os benefícios de negócio em benefícios tecnológicos, neste momento, vale considerar o negócio específico da ANVISA onde se está executando um projeto SOA.

Nesse sentido, podemos definir os benefícios tecnológicos relacionados aos benefícios de negócio conforme ilustrado na figura 19.

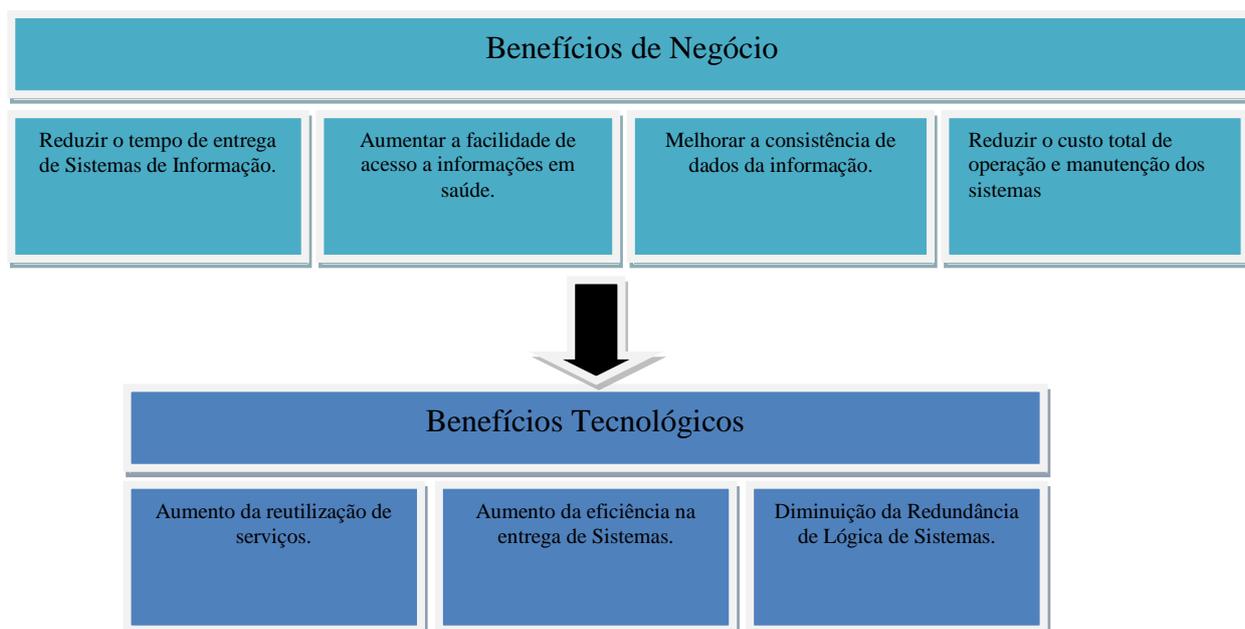


Tabela 19 – Benefícios Tecnológicos da Adoção de SOA.

Com o aumento da Reutilização de Serviços – conforme projetos de SOA vão sendo desenvolvidos – que são executados para o atendimento às demandas pelos sistemas de informação da organização – é iniciado o processo de população do barramento de serviços. Pelas características e princípios técnicos aplicados, os analistas e arquitetos de sistemas terão a disposição, conjuntos de capacidades de serviços acessíveis e padronizados.

Este benefício depende da padronização advinda da aplicação, em uma extensão significativa, dos princípios da orientação a serviço. Esta padronização deve ser seguida para construção dos serviços e sua disponibilização, a partir da definição da metodologia de referencia.

Portanto, caberá, em tempo de execução de projeto de SOA, desenvolver serviços intrinsecamente interoperáveis e agnósticos, capazes de proporcionar sua reutilização em diferentes processos de negócio da ANVISA, que por sua vez solidificará o benefício definido.

A diretriz estratégica de SOA em orientar os projetos ao negócio intrinsecamente relacionado com o estreitamento das áreas de negócio com a tecnologia, nesse sentido, a realidade irá gerar um desafio ainda maior, tanto pela complexidade dos sistemas de informação, quanto pela abrangência aliada à dimensão regional do país. Aumentar a eficiência na Entrega de sistemas é um benefício que permitirá a ANVISA atender seu negócio e ampliar o entendimento dos processos de negócio de forma documentada, para entregas de sistemas de informação cada vez mais aderentes às necessidades.

As técnicas empregadas na análise orientada a serviço, em tempo de execução de projeto de sistemas, em conjunto com a utilização de notações de modelagem de processos de negócio (Ex.: BPMN – *Business Process Management Notation*) e aliados a uma plataforma tecnológica, fazem parte do esforço de ações necessárias para obtenção desse benefício.

Diminuição da Redundância de Lógica de Sistemas – Este benefício é, basicamente, o resultado da reutilização de serviços. Quando o princípio de reutilização é aplicado durante a análise orientada a serviço, permite que analistas e arquitetos de sistemas criem composições de serviços que atendam as capacidades geradas pelo processo de negócio modelado. Essas composições, basicamente, utilizam serviços já estabelecidos no barramento de serviços, evitando, assim, que novas lógicas de sistemas ou serviços sejam desenvolvidas para atender aos requisitos de negócio.

Padrões de projetos de SOA podem ser empregados para que se tenha uma diminuição da redundância de Lógica de Sistemas, como por exemplo, as definições de cápsula de legado, que empregam um padrão de projeto para construção de capacidades de serviço, ampliando a interoperabilidade de um sistema legado, sem a necessidade de redundância de código.

5. CONCLUSÃO

O primeiro passo do trabalho foi identificar através de estudos bibliográficos os princípios e definições a cerca da Arquitetura Orientada a Serviço, as características que podem ser consideradas relevantes na construção de sistemas de informação. Um conjunto de características foi identificado e em seguida, estudado isoladamente. O trabalho buscou também diferenciar o conceito de serviços do conceito de componentes, e concluiu que são conceitos similares. Identificou-se que o termo “serviço” é empregado de formas distintas na literatura acadêmica e na literatura não acadêmica. Os resultados desses estudos deram origem ao capítulo dois deste trabalho.

Paralelamente foi investigada a plataforma de desenvolvimento SOA *Suite* da Oracle, que fornece suporte a construção de sistemas de informações orientadas a serviços. Existem outras plataformas, porém a plataforma Oracle foi objeto de estudo, ma vez que é a plataforma utilizada pela ANVISA no estudo de caso. Encontrados possuem alguma similaridade com um dos três escolhidos para este trabalho.

O estudo serviu principalmente para desmistificar essa nova tecnologia, que vem sendo apresentada, principalmente pelo mercado de desenvolvimento de software, como uma grande inovação tecnológica e que irá alterar a maneira como o software é desenvolvido atualmente. A conclusão que o trabalho chegou é que os *Web Services* não apresentam grande inovação em termos de tecnologia, e sim uma inovação no que diz respeito a padronização de protocolos. Devido a essa padronização, é possível que a adoção dos XML *Web Services* em larga escala se concretize, diferentemente do que ocorreu em propostas anteriores de soluções para o desenvolvimento de sistemas distribuídos.

A última parte do trabalho foi construir uma arquitetura que fornecesse suporte ao desenvolvimento orientado a serviços. Tirando proveito dos estudos feitos nas fases anteriores do trabalho, o objetivo foi criar uma arquitetura que, uma vez implantada, resolvesse grande parte dos problemas de orientação a serviços com simplicidade.

Além disso, a arquitetura aplicada à ANVISA foi construída prover uma nova arquitetura de desenvolvimento de software para a ANVISA padronizada com as bases de

dados e ferramentas Oracle já existente, a fim de manter a compatibilidade e a capacidade técnica da equipe de desenvolvedores.

Atualmente a arquitetura já é utilizada pela ANVISA para definição do Sistema de Registro Eletrônico de Medicamentos da GGMed/ANVISA. Em ambas as aplicações, a escolha do Oracle SOA Suite ocorreu principalmente devido a sua facilidade para construir aplicações que descrevem seus dados em XML e a necessidade de integração com componentes já existentes desenvolvidos para a plataforma J2EE. O uso do Oracle SOA Suite gera grande produtividade, principalmente pela possibilidade de programação declarativa do uso dos serviços através da plataforma Oracle.

5.1. Trabalhos Futuros

Em relação a trabalhos futuros este trabalho fornece algumas opções no que diz respeito à continuidade do desenvolvimento do framework. Abaixo são citadas algumas das possibilidades:

Maturidade da Arquitetura.

Para a Arquitetura atingir sua maturidade e conseqüente aumento da usabilidade, é necessário que haja um estreitamento entre a área de negócio e a área de TI. Durante o desenvolvimento desta dissertação é notório a presença de matérias interdisciplinares, porém seria interessante aumentar a padronização da linguagem da arquitetura de forma que atinja as áreas de negócio.

Arquiteturas baseadas em outras plataformas tecnológicas

Um dos princípios do SOA é que sua implementação seja independente de plataforma tecnológica comercial, sendo assim é bem visto o desenvolvimento de uma arquitetura aplicada baseada e plataforma tecnológica em software livre.

Metodologia detalhadas para o desenvolvimento

O desenvolvimento de uma metodologia detalhada para o desenvolvimento de sistemas de informação apoiado em uma arquitetura específica baseada em conceitos de SOA.

REFERENCIA BIBLIOGRÁFICA

- [1] Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento – BPM CBOOK, ABPMP – *Association of Business Process Management Professional*, Versão 2 – Primeira liberação em português, <http://www.abmpm-br.org>.
- [2] Bieberstein et al. *Service-Oriented Architecture (SOA) Compass: Business Value, Planning, and Enterprise Roadmap*, IBM Press, Pearson Education, 2006. E-book.
- [3] Norbert Bieberstein; Robert G. Laird; Dr. Keith Jones; Tilak Mitra et al. *Executing SOA: A Practical Guide for the Service-Oriented Architect*, eText ISBN-13: 978-0-13-714947-6.
- [4] Paul C. Brown, *Implementing SOA: Total Architecture in Practice*, eText ISBN-10: 0-321-56265-8.
- [5] OASIS - Modelo de Referência para Arquitetura Orientada a Serviço – soa-rm-csbr - <http://www.pcs.usp.br/~pcs5002/oasis/soa-rm-csbr.pdf>
- [6] <http://oracled.wordpress.com/2009/03/11/when-to-use-rest-based-web-services/>
- [7] SOA Suite - <http://www.oracle.com/technologies/soa/soa-suite.html>
- [8] Oracle Architect Center - <http://www.oracle.com/technology/tech/soa/index.html>
- [9] Core JEE Patterns - <http://java.sun.com/blueprints/corej2eepatterns/Patterns/>
- [10] Thomas Erl – *Soa Princípios de Design de Serviços*, 2009, ISBN: 8576051893 ISBN-13: 9788576051893, Editora Prentice Hall Brasil.
- [11] Thomas Erl - *Service-Oriented Architecture Design Patterns*, 2008, ISBN: 0136135161 ISBN-13: 9780136135166, Editora Prentice Hall.
- [12] Thomas Erl - *Service-Oriented Architecture – Soa Concepts, Technology, And Design*, 2008, ISBN: 0131858580 ISBN-13: 9780131858589, Editora Prentice Hall.

[13] University of Waterloo (2006). A Very Brief History of Computer Science.
Página visitada em 2009-11-25.

ANEXO I - PLATAFORMA TECNOLÓGICA

Neste anexo serão descritos as ferramentas padronizadas que darão sustentação à implantação da Arquitetura de Referência SOA – ANVISA.

Como opção tecnológica, a área de Tecnologia da ANVISA optou pelos produtos Oracle.

Server / Componente	Versão
Oracle SOA Suite	10.1.3.4
Oracle BAM Server	10.1.3.4
Database Server	10g
JDeveloper	10.1.3.4
Oracle Application Server	10.1.3
Oracle Service Registry	10.3
Oracle BPA Suite	10.1.3.4
Oracle Adapter Services	10.1.3

Tabela 3 – Requisitos de Softwares da Plataforma Oracle empregada na ANVISA.

1 Ferramenta de Desenvolvimento (IDE)

A ferramenta de desenvolvimento adotada para a construção dos serviços e componentes para a arquitetura é o JDeveloper.

A referida ferramenta possibilita a preparação de todos os recursos necessários ao ambiente de desenvolvimento, além de permitir:

- Criação de *web services* que serão expostos no Barramento de serviços;
- Criação de projetos do Barramento de Serviços (ESB);
- Utilização de adaptadores;

O JDeveloper é a ferramenta de desenvolvimento que será utilizada pelos técnicos e pelo Arquiteto SOA no ambiente da ANVISA, e é integrada para trabalho entre todos os componentes do Oracle SOA Suite.

2 Oracle BPA

Integrações com outros componentes da solução

Oracle BPEL – através de arquivo BPEL exportado a partir de um processo de negócio modelado.

3 Oracle BPEL

Integrações com outros componentes da solução

- Oracle ESB – o processo BPEL invoca componentes ESB disponibilizados pelo Oracle ESB através de *Web Services*.
- Oracle BAM – Sensores são adicionados nos pontos do processo BPEL que foram definidos os KPI's. No momento de execução estes sensores são sensibilizados e atualizam as informações na base de dados do BAM.
- Oracle Business Rules – O processo BPEL pode definir um “Serviço de Decisão” que consulta uma regra de negócio no Oracle Business Rules que determina a continuidade do processo.
- Oracle Adapters – As referências para adaptadores podem ser adicionadas diretamente ao processo BPEL, para acessar componentes externos.

4 Oracle ESB

Integrações com outros componentes da solução

- Oracle BPEL – o componente ESB invoca processos BPEL disponibilizados pelo Oracle BPEL através de *Web Services*.

- Oracle Adapters – As referências para adaptadores podem ser adicionadas diretamente ao processo BPEL, para acessar componentes externos.

5 Oracle BAM

Integrações com outros componentes da solução

- Oracle BAM – o Oracle BAM recebe informações a serem processadas através dos sensores implementados nos processos BPEL e em *triggers* de bancos de dados que devem ser implementadas nas bases de dados a serem monitoradas.

6 Oracle Web Services Manager

Integrações com outros componentes da solução

- O Oracle *Web Services* Manager pode ser associado a qualquer outro componente da plataforma que exponha seus serviços através de *Web Services*, agindo como um encapsulador do serviço, adicionando os quesitos de segurança necessários.

7 Oracle Service Registry

Integrações com outros componentes da solução

- O Oracle Service Registry se integra com todos os outros componentes da solução que exponham seus serviços através de *Web Services*. Os serviços devem ser catalogados e categorizados no diretório central de serviços da ANVISA.

8 Oracle Adapter Services

Integrações com outros componentes da solução

Oracle BPEL – criação de serviços de para acesso a fontes de dados e a sistemas legados.

Oracle ESB – criação de serviços de para acesso a fontes de dados e a sistemas legados.

9 Oracle Business Rules

Integrações com outros componentes da solução

Oracle BPEL – As regras de negócio podem ser expostas como *Web Services*, sendo acessadas pelo processo BPEL como “Pontos de Decisão”.

Aplicações externas – As regras de negócio podem ser acessadas através de API Java disponibilizada.

10 Oracle Application Server

Integrações com outros componentes da solução

O Oracle Application Server se integra com todos os outros componentes da solução sendo plataforma base para a execução de todas as ferramentas utilizadas na arquitetura da ANVISA.

11 Qualidade

A arquitetura deverá respeitar e adequar seus módulos aos padrões estabelecidos neste documento. Havendo a introdução de camadas de integração entre os módulos, estes não podem onerar o desempenho da arquitetura ou utilizarem padrões diferentes dos especificados. Outros fatores de qualidade são escalabilidade, tempo de resposta, robustez e desempenho.

A qualidade da arquitetura será garantida através de inspeção, revisão e teste da arquitetura, como forma de agregar qualidade ao produto, reduzir custos e retrabalho. Maiores detalhes sobre testes estão descritos no documento de Testes da ANVISA.

É de extrema importância que a arquitetura atenda aos requisitos dos usuários para que a solução final possa satisfazê-los.

12 Governança de Serviços

Este capítulo tem como objetivo principal evidenciar os mecanismos de governança que serão utilizados na arquitetura para a garantia da sua integridade.

Controle de Versão das Informações

Possibilitar a delegação de dados no sistema. Neste caso, identificar univocamente o usuário responsável pela digitação das informações. Incluir a data e hora do momento da digitação, conteúdo anterior, e o endereço IP da máquina de onde foi executada tal operação.

A definição das informações que deverão passar por este controle será feita no momento do levantamento de requisitos, considerando seu grau de sigilo e confidencialidade.

Controle do Fluxo de Informação e Integridade de dados

Garantir que o acesso aos dados somente seja possível de ser acessado através do mecanismo de controle de acesso, mesmo para os perfis de administração do sistema ou por outro usuário que porventura venha a ter acesso físico total ao sistema.

Possuir mecanismos de certificação de origem que garantam que somente informações oriundas de servidores acessados através da camada de serviços sejam aceitas, e vice-versa.

Considerar a premissa do caminho não autorizado, a fim de evitar o acesso ao banco de dados por conexões de rede diferente que não a dos servidores de aplicação, evitando-se, desta forma, o acesso direto ao arquivo do banco de dados bem como a visualização ou manipulação do seu conteúdo.

Auditoria

Os sistemas devem possuir registros históricos (logs) para permitir auditorias e provas forenses, assegurando que as informações de auditoria possuam mecanismos que garantam sua integridade e confidencialidade, e garantir que o acesso a essas informações somente seja feito por usuários com perfil de administração do sistema.

A auditoria dos serviços criados e publicados no servidor de processos e no barramento de serviços deve ser realizada da seguinte forma:

Servidor de Processos

Dois níveis de logs de auditoria são suportados:

- Nível de domínio – mantém logs de auditoria entre domínios específicos;
- Nível de sistema – mantém logs de auditoria do servidor de processos.

13 Barramento de Serviços

Os logs de auditoria dos componentes publicados no barramento de serviços ficam registrados no Application Server no qual o barramento está instalado. Para acessar os logs no Oracle Application Server utiliza-se o Oracle Enterprise Manager Application Server Control.

Nesta ferramenta devem-se definir os níveis de log para cada tipo de componente publicado. Os níveis de auditoria devem ser definidos de acordo com o ambiente em uso:

Desenvolvimento: nível DEBUG (provê informações detalhadas de cada ação ocorrida no barramento);

Homologação/Testes: nível INFO (provê informações em nível de aplicação para validação de informações de teste);

Produção: nível ERROR (provê informações em nível de erros de sistema e ambiente).

A padronização e conteúdos dos arquivos de logs gerados pelos processos disponíveis do servidor de processos da ANVISA são definidos em documento de metodologia de desenvolvimento de serviços.

14 Origem dos Certificados Digitais

Incorporar, quando necessário, o uso de certificados digitais no padrão ICP-Brasil em conformidade com o disposto no Decreto 3.996 de 31 de outubro de 2001.

Controle de Autenticação pelo uso de Certificados Digitais

Incorporar mecanismos de assinatura digital pelo Certificado Digital Pessoal para os usuários com acesso ao sistema.

A definição das informações que utilizarão este mecanismo será feita no momento do levantamento de requisitos, considerando seu grau de sigilo e confidencialidade.

15 Comunicação Segura

Estabelecer um canal de comunicação seguro para tráfego das informações nas interações com sistemas. As mensagens SOAP estão seguras usando o WS-Security, que define como a autenticação, codificação e assinaturas digitais devem ser usadas para assegurar as comunicações.

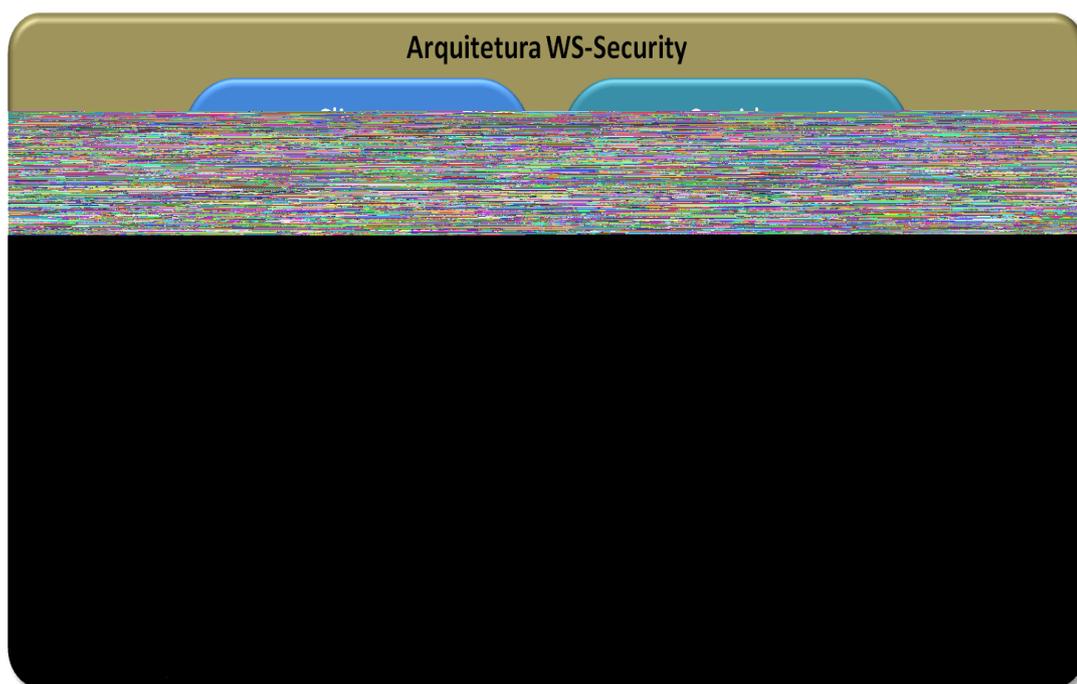


Figura 20 – Canal de Comunicação entre *Web-Services*

16 Cópias de Segurança e Restauração de Dados

A rotina de backup, bem como, as cópias de segurança e a restauração dos dados, deverão seguir as normas estabelecidas pela GESIS/GGTIN/ANVISA.

Autenticação

Suportar autenticação de usuário que expire por tempo absoluto ou inatividade (configurável), com redirecionamento automático para tela de logon.

Possuir interface de aplicação (API) em Java que permita a implementação de extensões no mecanismo de autenticação.

Autorização

Oferecer autorização de acesso ao nível de páginas e *portlets*.

Permitir que privilégios específicos possam ser atribuídos a usuários ou grupo de usuários específicos.

Permitir autorização de acesso a conteúdo específico para usuários ou grupo de usuários.

Possuir interface de aplicação (API) em Java que permita a implementação de extensões no mecanismo de autorização.

Padrões Aplicáveis

Padrões definidos pela GESIS/GGTIN/ANVISA.

Todo o desenvolvimento de sistemas deverá ter como base os documentos listados abaixo. Os padrões que não estiverem contemplados neste documento devem ser definidos em comum acordo junto com a GESIS/GGTIN/ANVISA.

Metodologia de Desenvolvimento de Sistemas baseado em Serviços da GESIS/GGTIN/ANVISA.

Arquitetura Java - Arquitetura de Referência J2EE

Arquitetura de Referência para Portais

Identidade Visual - Manual de Identidade Visual e Camada de Apresentação

Norma de Padronização do Banco de Dados

Normas ISO e NBR para Segurança da Informação

Para a definição dos requisitos de segurança deste documento foram consideradas duas normas ISO: a NBR ISO/IEC 17799 e ISO 15408 por estabelecerem as melhores práticas para garantir a segurança da informação em ambientes computacionais.

Norma NBR ISO/IEC 17799

A NBR ISO/IEC 17799 é a versão brasileira para da norma ISO/IEC 17799:2000 que, por sua vez, é uma transcrição da BS 7799. Publicada inicialmente em 1995, a BS7799 é um ponto de referência para a área de segurança da informação. A norma está dividida em duas partes:

Um código de práticas (melhores práticas) para orientar as organizações quanto à segurança da informação, publicado em 2000, como padrão ISO/IEC 17799; e

Especificação para orientar a criação de Sistemas de Gestão de Segurança da Informação aderentes à norma.

No ano de 2001, a norma foi traduzida e publicada pela Associação Brasileira de Normas Técnicas (ABNT) como NBR 17700 – Código de Prática para a Gestão da Segurança da Informação.

Norma ISO/IEC 15408

A norma ISO/IEC 15408 – *Evaluation criteria for IT Security*, também conhecida por *Common Criteria for Information on Technology Security Evaluation* – auxilia o desenvolvedor ou o integrador de software, em qualquer escala, a incluir, melhorar ou simplesmente avaliar os aspectos de segurança do software em desenvolvimento; e, para o cliente de um software, especificar a segurança do software.

Padrões de Interoperabilidade de Governo Eletrônico (e-PING)

O padrão e-PING define um conjunto mínimo de premissas, políticas e especificações técnicas que regulamentam a utilização da Tecnologia de Informação e Comunicação na interoperabilidade de Serviços de Governo Eletrônico, estabelecendo as condições de

interação com os demais poderes e esferas de governo e com a sociedade em geral. Para referência na construção da arquitetura será adotado o e-PING v4.0 de 16/12/2008.

Modelo de Acessibilidade (eMAG)

O modelo de Acessibilidade definido pelo Departamento de Governo Eletrônico define um conjunto de diretrizes para o desenvolvimento e adaptação de conteúdos do governo na internet que proporcionarão que o processo de acessibilidade dos sítios do Governo Brasileiro seja conduzido de forma padronizada, de fácil implementação, coerente com as necessidades brasileiras, e em conformidade com os padrões internacionais. Para referência na construção dos sistemas da ANVISA será adotada o eMAG v2.0 de 14/12/2005.

Normas NBR 13596 – Qualidade de Software

A norma NBR 13596 permite visualizar mais facilmente a qualidade de um produto de software e, por conseqüência, definir uma forma de medir esta qualidade, possibilitando uma avaliação mais objetiva e uniforme. Além disso, é acessível a qualquer pessoa ou empresa, por se tratar de uma norma brasileira e, portanto, disponível a quem se interessar.

Padrões de Modelagem de Processos

Notação de Modelagem de Processos

Como Notação de Modelagem de Processos foi estabelecida a BPMN 1.0, como definido pela OMG/BPMI (Object Management Group/ Business Process Management Initiative) (www.bpmn.org).

Linguagem para Orquestração de Serviços

Como Linguagem para Execução de Processos foi estabelecida a linguagem BPEL (Business Process Execution Language) nas versões BPEL4WS versão 1.1, como definido pelo OASIS (www.oasis-open.org).

Padrões para Linguagem de Desenvolvimento

Linguagem de Desenvolvimento

Como padrões de Linguagem de Desenvolvimento foram estabelecidos as tecnologias Java SE (Standard Edition) versões 1.4.2 ou 5.0 e Java EE (Enterprise Edition) versões 1.4 ou 5.0, como definido pela Java Community Process (www.jcp.org) e Sun (java.sun.com/javaee, java.sun.com/javase).

Desenvolvimento de *web services*

Como padrões para o desenvolvimento de web services foram estabelecidos a Java 2 Platform, Enterprise Edition 1.4 (J2EE 1.4) Specification (JSR151), Enterprise Web Services 1.1 (JSR 921) e Web Services Metadata (JSR 181), como definido pelo Java Community Process (www.jcp.org).

Construção de Interfaces de Usuário

Como padrão para o construção de interfaces de usuário foi estabelecido Java Server Faces (JSR 127) e Portlet Bridge Specification for JavaServer Faces (JSR 301), como definido pelo Java Community Process (www.jcp.org). Será permitida ainda a utilização da derivação como Oracle ADF.

Como padrões para a construção de Portlets foram estabelecidos Portlet Specification (JSR 168) e WSRP 1.0 (Web services for Remote Portlets), como definido pela Java Community Process (www.jcp.org) e OASIS (www.oasis-open.org).

Linguagem de Definição do Serviço

Como Linguagem de Definição do Serviço foi estabelecida a WSDL 1.1 (WebService Description Language), como definido pelo W3C (www.w3.org).

Serviços de Mensagens

Como padrão para serviços de mensagens Java Message Service (JMS) API (JSR 914), como definido pelo Java Community Process (www.jcp.org).

Protocolo de Troca de Informações

Como Protocolo de Troca de Informações foi estabelecido SOAP 1.2 (Simple Object Access Protocol), como definido pelo W3C (www.w3.org).

Infra-estrutura de Registro

Como Infra-estrutura de Registro foi estabelecida UDDI 3.0.2 (Universal Description, Discovery and Integration), como definida pela OASIS (uddi.org).

Linguagem para Intercâmbio de Dados

Como Linguagem para Intercâmbio de Dados foi estabelecida XML 1.0 (Extensible Markup Language), como definido pelo W3C (www.w3.org).

Formato de Intercâmbio de Hipertexto

Como Formato de Intercâmbio de Hipertexto foram estabelecidos HTML versão 4.01, XHTML versões 1.0 ou 1.1 e XML versões 1.0 ou 1.1, gerados conforme especificações do W3C(www.w3.org).

17 Padrões de Segurança

Como Padrões de Segurança foram estabelecidos JAAS 1.0, x.509 e SSL v.3, conforme estabelecidos por respectivamente, SUN (java.sun.com/products/jaas/) e IETF (Internet Engineering Task Force) (tools.ietf.org/html/rfc3280 e www.ietf.org).

Padrões de Segurança para web services

Como Padrões de Segurança para web services foram estabelecidos WS-Security v1.1, Username Token Profile v1.1, X.509 Token Profile v1.1, SAML Token profile v1.1, Kerberos Token Profile v1.1, Rights Expression Language (REL) Token Profile v1.1 e SOAP with Attachments (SWA) Profile v1.1 conforme estabelecidos pela OASIS (www.oasis-open.org).

Padrão para Gerência Distribuída de Recursos

Como padrão para Gerência distribuída de recursos foi estabelecido uma arquitetura de web services baseada no padrão WSDM (Web Services Distributed Management) v1.1 como definido pela OASIS (www.oasis-open.org).

Padrão de Endereçamento de web services

Como padrão para o endereçamento de web services e Mensagens foi estabelecido WS-Addressing v1.0 conforme definido pela W3C (WWW.w3c.org).

Acesso a Repositório de Conteúdo

Como padrão para acesso a repositório de conteúdo foi estabelecido Content Repository for Java™ technology API (JSR 170) como definido pela Java Community Process (www.jcp.org).