

UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO  
ESCOLA DE INFORMÁTICA APLICADA  
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

Identificação automática de serviços em uma abordagem SOA

Nome do autor:

Henrique Prado Sousa

Nome dos Orientadores:

Flávia Maria Santoro e Leonardo Guerreiro Azevedo

Março/2010

# Identificação automática de serviços em uma abordagem SOA

Projeto de Graduação apresentado à Escola de Informática Aplicada da Universidade Federal do Estado do Rio de Janeiro (UNIRIO) para obtenção do título de Bacharel em Sistemas de Informação

Nome do autor:

Henrique Prado Sousa

Nome dos Orientadores:

Flávia Maria Santoro e Leonardo Guerreiro Azevedo

## **Dedicatórias e agradecimentos**

Ninguém alcança grandes conquistas sem o apoio de pessoas especiais que fazem parte da nossa vida. A cada passo que damos, encontramos pessoas que realmente se importam e vibram com nossas vitórias. A todas essas pessoas devo imenso respeito, carinho e gratidão. A mim, resta apenas corresponder e multiplicar todos esses nobres valores, doando ao próximo toda atenção, paciência e dedicação que recebi e me permitiu voar mais alto.

Dedico e agradeço especialmente por todas essas conquistas realizadas a vocês:

### **Deus**

A Ele devo muito mais do que dedicatórias e agradecimentos. Afinal, foi ele quem planejou tudo isso. Sua presença cada vez mais clara na minha vida, renova minhas forças e com certeza me levará a muitas experiências enobrecedoras. A Ele, minha eterna gratidão.

### **Meus pais**

Pela dedicação exclusiva durante toda minha vida, o que me preparou e permitiu chegar até aqui. Acho que isso foi o mínimo que poderia fazer considerando todas as batalhas que vocês enfrentaram durante esses anos dedicando-se ao meu desenvolvimento. O hoje só foi possível devido a todo amor que recebo de vocês desde quando nasci, por isso, vos dedico esta conquista.

### **Cíntia**

Aquela que esteve ao meu lado e amadureceu junto comigo desde o início durante todos esses anos de namoro. Agradeço por todo o auxílio, paciência e carinho. Você foi fundamental em todos os momentos de altos e baixos que a vida me proporcionou nos últimos 6 anos, me dando força para continuar caminhando, mesmo quando pisava em pedras em alguns momentos desta grande caminhada. Muito obrigado!

### **Orientadores**

Se não fosse por essas duas pessoas incríveis não teria feito metade do que fiz durante os últimos anos. Eles foram fundamentais na minha formação acadêmica e a eles dedico essa conquista e todas as que alcançarei. São minha fonte de inspiração profissional e pessoal que vou levar por toda a vida no meu coração. Muitíssimo obrigado!

### **Corpo docente UNIRIO**

Toda essa dedicação intensiva e amor pelo trabalho que é característico de todo o corpo docente da UNIRIO, ainda vão resultar em conquistas ainda maiores do que as inúmeras que foram alcançadas nos últimos anos. Agradeço por toda a ajuda que recebi de prontidão em todo momento que necessitei e todo o ensinamento, incluindo as diversas noites viradas fazendo trabalhos complexos que me ajudaram a amadurecer meu conhecimento em Sistemas de Informação.

Agradeço também a todos que me apoiaram e me ajudaram nos diversos momentos em que necessitei. Mesmo que a colaboração seja pequena, foram tantos os amigos que dedicaram seu tempo a mim que ao final todos foram de suma importância. Muito obrigado!

Henrique Prado Sousa

*Para o século XXI esperávamos  
supercomputadores, robôs de alta tecnologia e para  
surpresa de muitos conseguimos resultados ainda  
maiores, Somos as próprias máquinas!*

(Paulo Lutzoff)

# SUMÁRIO

---

Capítulo 1:	Introdução .....	1
1.1	Motivação.....	1
1.2	Objetivo.....	5
1.3	Estrutura do texto .....	5
Capítulo 2:	Arquitetura Orientada a Serviços .....	6
2.1	Definição de SOA .....	6
2.2	Serviços .....	7
2.2.1	Interfaces e contratos.....	8
2.2.2	Outras características dos serviços.....	9
2.3	Web Services.....	11
2.4	Ciclo de desenvolvimento .....	12
2.5	BPM e SOA .....	14
2.6	Governança em SOA.....	16
2.7	Resumo.....	18
Capítulo 3:	Modelagem de processos de negócio .....	19
3.1	Definição .....	19
3.2	Objetivos .....	20
3.3	Componentes da modelagem de processos de negócio.....	21
3.3.1	Método .....	22
3.3.2	Metamodelo.....	28
3.3.3	Notação .....	28
3.3.4	Ferramenta.....	30
3.4	Diagramas de processos de negócio.....	31
3.4.1	VAC – <i>Valued Added Chain</i> .....	31
3.4.2	EPC – <i>Event-driven Process Chain</i> .....	32
3.4.3	FAD – <i>Function Allocation Diagram</i> .....	34
3.5	Resumo.....	35
Capítulo 4:	Metodologia para a identificação automática de serviços candidatos a partir de modelos de processos de negócio .....	36
4.1	Identificação de serviços em SOA .....	36
4.2	Método de identificação de serviços candidatos .....	37
4.3	Heurísticas de identificação de serviços candidatos .....	39
4.3.1	Heurística de identificação de serviços a partir de regras de negócio....	39
4.3.2	Heurística de identificação de serviços a partir de requisitos de negócio	40
4.3.3	Heurística de identificação de serviços a partir de <i>cluster</i> .....	42
4.3.4	Heurística de identificação de serviços a partir de padrões de <i>workflow</i>	44
4.4	Heurísticas de consolidação de serviços candidatos .....	58
4.4.1	Heurística de eliminação de serviços candidatos .....	59
4.4.2	Heurística de grau de reuso de serviços candidatos .....	60
4.4.3	Heurística de grau de reuso de serviço candidato identificado a partir de atividade de múltiplas instâncias.....	63
4.4.4	Heurística de associação de serviços candidatos com sistemas .....	63

4.4.5	Heurística de associação de serviços candidatos com requisitos da demanda .....	65
4.4.6	Heurística de associação de serviços com papéis .....	66
4.4.7	Heurística de associação de serviços com atividades .....	67
4.4.8	Heurística de associação de serviço de dados .....	68
4.4.9	Heurística de associação de <i>clusters</i> com modelo conceitual.....	69
4.4.10	Heurística de associação de serviços candidatos a partir de associações entre elementos do modelo.....	70
4.4.11	Heurística de identificação de serviços utilitários.....	71
4.4.12	Heurística de consolidação de serviços identificados a partir de fluxo	
	73	
4.5	Resumo.....	74
Capítulo 5: Automatização do método de identificação de serviços a partir da modelagem de processo de negócio .....		77
5.1	Requisitos do sistema de automação.....	77
5.2	Ferramenta ARIS .....	78
5.3	Modelos de classes .....	80
5.4	Interface.....	80
5.5	Codificação .....	80
5.5.1	Heurística de identificação de serviços a partir de regras de negócio....	81
5.5.2	Heurística de identificação de serviços a partir de requisitos de negócio	
	81	
5.5.3	Heurística de identificação de serviços a partir de <i>cluster</i> .....	82
5.5.4	Heurística de identificação de serviços a partir de padrões de <i>workflow</i>	
	82	
5.5.5	Heurística de consolidação para grau de reuso .....	85
5.6	Produtos da identificação automática de serviços candidatos.....	88
5.1	Resumo.....	93
Capítulo 6: Resultados obtidos em estudo de caso com o método de identificação automatizada de serviços candidatos .....		95
Capítulo 7: Conclusão.....		113
Referências Bibliográficas .....		115
Anexo I – Tabela de notação de símbolos para modelagem de processos.....		120
Anexo II – Processo Analisar Pedido de Crédito.....		124
2.1	Analisar pedido de crédito .....	124
2.2	Verificar cadastro do cliente .....	127
2.3	Cadastrar cliente.....	127
2.4	Atualizar cadastro do cliente.....	128
2.5	Verificar limite de crédito do cliente .....	129
2.6	Cancelar proposta de crédito.....	130
2.7	Comunicar proposta não aprovada.....	130
2.8	Comprometer limite de crédito .....	131
2.9	Calcular alíquota de imposto.....	132
2.10	Determinar taxa de juros a ser cobrada do cliente .....	133
2.11	Gerar proposta de contrato .....	134
2.12	Analisar contrato .....	135
2.13	Verificar condições de contrato com cliente.....	137
2.14	Aprovar contrato .....	137

2.15	Cancelar contrato .....	138
2.16	Comunicar não aprovação de contrato de risco .....	139
2.17	Cancelar contrato de risco.....	140
2.18	Alterar proposta de crédito.....	141
2.19	Elementos de Apoio ao Processo .....	141
2.20	Regras de Negócio .....	141
2.21	Bases de Dados .....	142
2.22	Informações.....	142
2.23	Requisitos de Negócio.....	145
Anexo III	– Modelos de classes .....	149

## LISTA DE FIGURAS

---

Figura 1-1: Caso real de arquitetura de integração [NETO, 2006] .....	2
Figura 1-2: Foco do mercado dedicado a SOA [NETO, 2006].....	3
Figura 2-1: Definições de SOA segundo perspectivas distintas [NETO, 2006].....	7
Figura 2-2 – Ciclo de vida de serviços contendo a fase de identificação de serviços [JOSUTTIS, 2007] .....	13
Figura 2-3- Inclusão da fase de retirada de serviços obsoletos [JOSUTTIS, 2007] ..	14
Figura 3-1 – Objetivos da modelagem de processos de negócio [ARAÚJO, 2008]..	20
Figura 3-2 – Componentes da modelagem de processos de negócio [ARAÚJO, 2008] .....	22
Figura 3-3 – Método para gestão de processos de negócio (Adaptação [ARAÚJO, 2008] .....	23
Figura 3-4 – Exemplo de modelo VAC .....	32
Figura 3-5 – Exemplo de modelo EPC .....	34
Figura 3-6 – Exemplo de modelo FAD.....	35
Figura 4-1 – Etapas do método de identificação de serviços candidatos.....	38
Figura 4-2 – Exemplo de regra de negócio. ....	40
Figura 4-3 – Requisitos de negócio “Aprovar créditos concedidos” e “Consultar créditos concedidos”. ....	41
Figura 4-4 – Exemplo de <i>clusters</i> conectados a portadores de informação.....	43
Figura 4-5 – Exemplo de padrão de <i>workflow AND</i> onde os múltiplos fluxos paralelos convergem em um fluxo de controle simples .....	45
Figura 4-6 – Exemplo de padrão de <i>workflow AND</i> com atividades automatizadas .	46
Figura 4-7 - Exemplo de padrão de <i>workflow AND</i> onde um dos fluxos converge para um evento final e o outro fluxo converge para outro processo .....	46
Figura 4-8 – Exemplo de padrão de <i>workflow OR</i> onde os múltiplos fluxos convergem em um fluxo de controle simples .....	48
Figura 4-9 - Exemplo de padrão de <i>workflow OR</i> onde fluxos convergem para eventos finais.....	48
Figura 4-10 – Exemplo de padrão de <i>workflow XOR</i> onde os múltiplos fluxos convergem em um fluxo de controle simples .....	50
Figura 4-11 - Exemplo de padrão de <i>workflow XOR</i> onde um dos fluxos converge para um evento final.....	51
Figura 4-12 - Exemplo de sequência de atividades.....	52
Figura 4-13 – Exemplo de ciclo/loop de atividades.....	54
Figura 4-14 – Comunicação entre processos utilizando interfaces de processo .....	56
Figura 4-15 – Atividade de múltiplas instâncias identificam três serviços candidatos .....	58
Figura 4-16 – Eliminação de serviços candidatos.....	60
Figura 4-17 – FAD da atividade “Configurar parâmetros para verificação de variação da produção” .....	60
Figura 4-18 – FAD da atividade “Gerar proposta de contrato” .....	61
Figura 4-19 – FAD da atividade “Verificar cadastro do cliente”.....	61
Figura 4-20 – FAD da atividade “Atualizar cadastro do cliente” .....	62
Figura 4-21 – Atividade apoiada por mais de um sistema.....	64
Figura 4-22 – Modelo de requisito de negócio do sistema “CotaWare”.....	65



Figura 4-23 – Atividade “Aprovar contrato” .....	66
Figura 4-24 – Atividade “Gerar proposta de contrato” .....	67
Figura 4-25 – Atividade “Cancelar contrato” .....	68
Figura 4-26 – Atividade “Verificar cadastro do cliente” .....	69
Figura 4-27 – Atividade “Cadastrar cliente” .....	69
Figura 4-28 - Exemplo de associação entre <i>cluster</i> e modelo de dados.....	70
Figura 4-29 – Modelo de dados de “Proposta de contrato” .....	70
Figura 4-30 – Exemplo de serviço utilitário (adaptado de THOM <i>et al.</i> , 2007).....	73
Figura 5-1 – Tela para escolha de heurísticas e relatórios .....	80

## LISTA DE TABELAS

---

Tabela 5-1 – Lista de serviço para regras de negócio .....	90
Tabela 5-2 – Lista de serviços para <i>cluster</i> .....	90
Tabela 5-3 – Lista de serviços para loop .....	90
Tabela 5-4 – Lista de serviços para interfaces de processo .....	91
Tabela 5-5 – Lista de serviços para múltiplas instâncias .....	91
Tabela 5-6 – Consolidação dos serviços candidatos .....	92
Tabela 5-7 – Associação Serviço x Atividade .....	92
Tabela 5-8 – Associação Serviço x Papéis .....	93
Tabela 5-9 – Resumo dos serviços de fluxo .....	93
Tabela 6-1 – Serviços identificados a partir da heurística de regras de negócio .....	96
Tabela 6-2 – Serviços identificados a partir da heurística de requisitos de negócio ..	98
Tabela 6-3 – Serviços identificados a partir da heurística de Loop .....	100
Tabela 6-4 – Serviços identificados a partir da heurística de atividades de múltiplas instâncias .....	100
Tabela 6-5 – Serviços identificados a partir da heurística de interface de processo	100
Tabela 6-6 – Serviços identificados a partir de heurística de atividades sequenciais .....	100
Tabela 6-7 – Serviços identificados a partir da heurística de <i>cluster</i> .....	102
Tabela 6-8 – Relatório de serviços consolidados .....	104
Tabela 6-9 – Relatório de Serviços X Atividades .....	106
Tabela 6-10 – Relatório de Serviços x Papéis .....	111
Tabela 6-11 – Tabela de consolidação de serviços de fluxo .....	112

## LISTA DE ALGORITMOS

---

Algoritmo 5-1– Identificar serviço candidato a partir de regra de negócio .....	81
Algoritmo 5-2 - Identificar serviço candidato a partir de requisito de negócio .....	82
Algoritmo 5-3 - Identificar serviço candidato a partir de <i>cluster</i> .....	82
Algoritmo 5-4 - Identificar serviço candidato a partir de sequência de atividades....	83
Algoritmo 5-5 – Obter atividades sequenciais .....	83
Algoritmo 5-6 – Identificar serviço candidato a partir de loop.....	84
Algoritmo 5-7 - Identificar serviço candidato a partir de interface de processo.....	84
Algoritmo 5-8 - Identificar serviço candidato a partir de atividade de múltiplas instâncias .....	85
Algoritmo 5-9 – Cálculo do grau de reuso para serviços identificados a partir de regras e requisitos de negócio .....	85
Algoritmo 5-10 – Cálculo do grau de reuso para serviços identificados a partir de cluster .....	86
Algoritmo 5-11 – Cálculo do grau de reuso para serviços identificados a partir de atividades sequenciais .....	86
Algoritmo 5-12 – Cálculo do grau de reuso para serviços identificados a partir de loop.....	87
Algoritmo 5-13 – Cálculo do grau de reuso para serviços identificados a partir de interface de processo .....	88

## LISTA DE ABREVIATURAS

---

API - *Application Programming Interface*

BPEL - *Business Process Execution Language*

BPEL4WS - *Business Process Execution Language for Web Services*

BPM - *Business Process Management; Business Process Modeling*

BPMI - *Business Process Management Initiative*

BPMN - *Business Process Modeling Notation*

CORBA - *Common Object Request Broker Architecture*

CRUD - *Create, Retrieve, Update e Delete*

EPC - *Event-driven Process Chain*

ESB - *Enterprise Service Bus*

FAD - *Function Allocation Diagram*

OASIS - *Organization for the Advancement of Structured Information Standards*

OMG - *Object Management Group*

SLA – *Service Level Agreement*

SOA - *Service Oriented Architecture*

TI - *Tecnologia da Informação*

UDDI - *Universal Description, Discovery and Integration*

UML - *Unified Modeling Language*

VAC - *Value Added Chain*

XML - *Extensible Markup Language*

WSDL - *Web Service Description Language*

## RESUMO

---

A modelagem de processos de negócio está sendo utilizada em larga escala nas empresas atualmente. Quando bem elaborados, os modelos englobam informações detalhadas sobre as atividades realizadas no processo: fluxo de execução, sistemas de apoio, regras de negócio, requisitos de negócio, tecnologias de armazenamento, dados de entrada e saída das atividades e outras informações relevantes. Em uma abordagem SOA, essa gama de informações permite uma análise computacional automatizada, baseada em heurísticas que identificam serviços candidatos a partir dos modelos de processo de negócio. Este trabalho descreve como é realizada a identificação automatizada de serviços candidatos, as informações extraídas no processo e sua utilização no apoio aos analistas e desenvolvedores.

**Palavras-chave:** SOA, Modelagem de Processos de Negócio, Identificação de Serviços, Serviços Web.

# Capítulo 1: INTRODUÇÃO

---

O objetivo deste capítulo é contextualizar o assunto abordado neste trabalho, bem como apresentar seu objetivo e a estrutura dos capítulos.

## 1.1 Motivação

Organizações modernas precisam responder de forma efetiva e rápida às oportunidades do mercado [FURTADO *et al.*, 2009]. Os sistemas de informação tornaram-se grandes aliados do negócio, uma vez que apoiam e automatizam atividades dos diversos processos que visam os objetivos da empresa. A busca pelo domínio do mercado por partes das organizações fez com que a procura por sistemas de apoio computacional aumentasse e com isso, o desenvolvimento de softwares crescesse de forma que, hoje, podemos encontrar diversas soluções para o mesmo problema utilizando tecnologias distintas.

As organizações, à medida que crescem geralmente se dividem em departamentos com escopo de atuação definidos, o que resulta em necessidades distintas a cada nova divisão. Então, os sistemas adquiridos para apoiar um departamento raramente suprem as necessidades de outro que, por conseguinte, busca novo software.

Os responsáveis ou proprietários de cada software, resultante das aquisições individuais, são subpartições distintas que compõe uma organização ou ainda empresas parceiras. Cada entidade citada possui planos de gestão e visões que diferem e criam barreiras políticas para iniciativas que dependam de terceiros.

Ao final, a organização encontra-se logicamente segregada pelas *aplicações proprietárias* adquiridas que não se comunicam, em um quadro muitas vezes irreversível, considerando o valor investido. Por conseguinte, o investimento o qual esperava se transformar em lucro torna-se um fator negativo que implicará na perda de competitividade dentro no nicho em que se encontra o negócio. A manutenção da infraestrutura de sistemas se torna insustentável e a necessidade constante de adaptação torna-se inviável, considerando tempo e custo necessário para sua realização [JOSUTTIS, 2007].

Sem uma gerência atuante e capacitada no que tange à aquisição de programas de apoio ao negócio, é possível imaginar que em um dado momento vão existir dezenas de softwares distintos, desenvolvidos com tecnologias diferentes e que não se comunicam, porém, trabalhando por um mesmo objeto: agregar valor à organização.

A partir disto, é possível deduzir problemas significativos que são introduzidos com a ausência de integração entre os sistemas. Este quadro é encontrado em diversas organizações que enfrentam problemas introduzidos por causa da desordem em sua arquitetura de software. Por exemplo, a Figura 1.1 mostra um caso real de um modelo de arquitetura de integração que se encontra ilegível devido ao excesso de componentes que correspondem a diferentes plataformas.

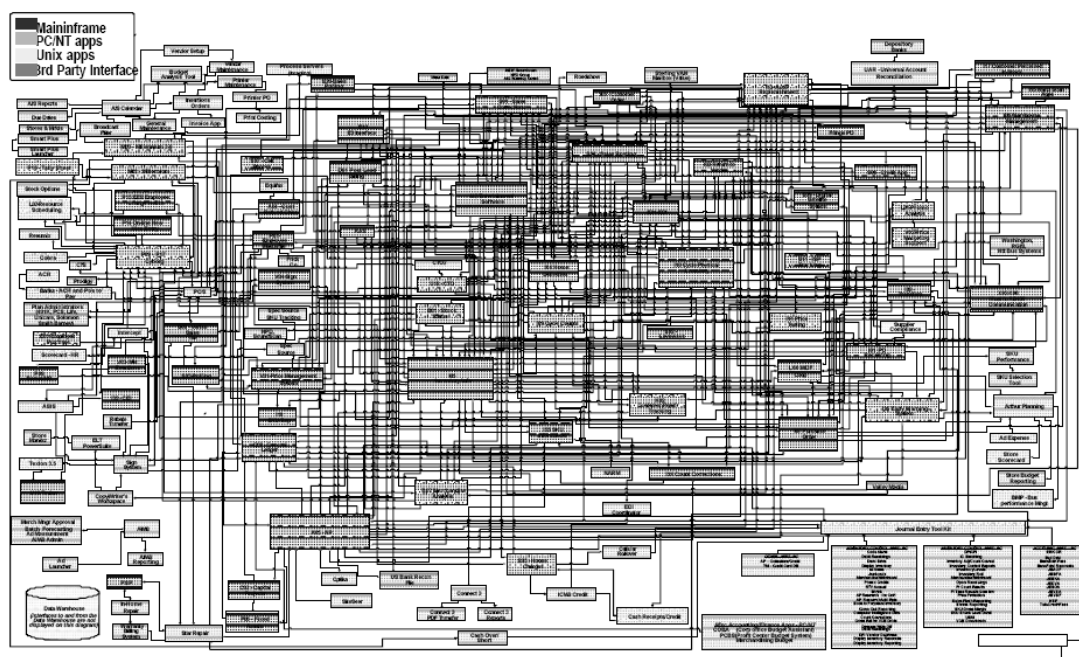


Figura 1: Caso real de arquitetura de integração [NETO, 2006]

Em vista dos problemas de compatibilidade, desenvolvimento e manutenção de softwares, os sistemas de informação, e consequentemente a TI, para garantir o alinhamento às novas necessidades do negócio, passam por reformulações constantes em suas estruturas e paradigmas.

Diversas técnicas foram desenvolvidas para amenizar este quadro, baseado na necessidade de redução de custos, produtividade, reaproveitamento de código e

escalabilidade. Entre as soluções desenvolvidas, a arquitetura orientada a serviços (SOA) destaca-se principalmente pelos seguintes benefícios [MARKS e BELL, 2006]:

- Redução do tempo de desenvolvimento
- Redução de custos
- Redução de riscos
- Melhor alinhamento da TI com o negócio

Como solução para *sistemas distribuídos*, a abordagem *Service-Oriented Architecture* (SOA) ou Arquitetura Orientada a Serviços tem se destacado como solução para *heterogeneidade* entre as aplicações, principalmente motivada pelo *reuso* [JOSUTTIS, 2007]

Os benefícios supracitados e outros que serão apresentados posteriormente chamam a atenção do mercado que, sempre em busca do lucro, investe em tecnologia esperando um retorno. SOA encontra-se como um dos assuntos mais cotados entre os profissionais de TI, conforme é apresentado na Figura 2.

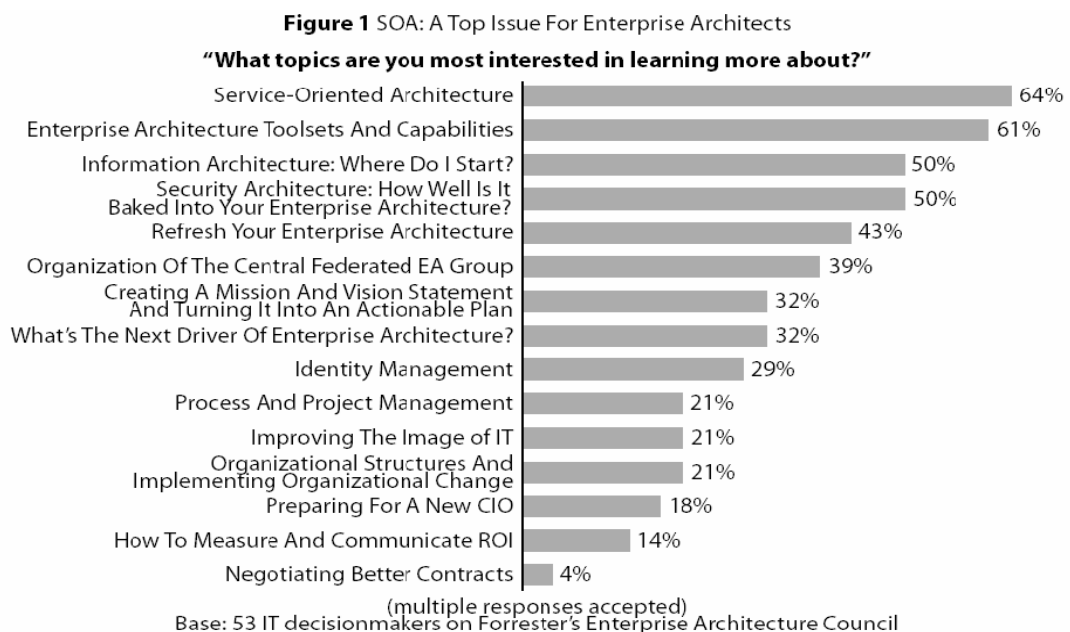


Figura 2: Foco do mercado dedicado a SOA [NETO, 2006]

Entretanto, a adoção de SOA em uma organização deve ser considerada um projeto de longo prazo que demanda muito investimento. A concepção deste novo



paradigma necessitará de esforço coletivo e um ponto fundamental que, segundo pesquisa realizada pelo SOA Fórum - grupo internacionalmente renomado em pesquisa sobre SOA -, é o principal motivo para o comprometimento dos projetos: governança. O estudo realizado com quase 1,3 mil executivos, mostra que a governança de projetos é insuficiente em 88% das organizações. Quase nove entre dez grandes empresas têm práticas insuficientes de governança sobre implantações de arquitetura orientada a serviços [COMPUTERWORLD, 2007 *apud* SOA Fórum].

Apesar da abordagem orientada a serviços requerer mais disciplina e planejamento, o retorno de investimento é elevado [MARKS e BELL, 2006]. Diversas empresas alcançaram êxito na implementação de SOA em sua estrutura e obtiveram os retornos esperados.

Por exemplo, a empresa de energia *Valero Energy* [VALERO ENERGY, 2008] adotou a arquitetura SOA a fim de tratar suas necessidades de mudanças em óleo e gás devido a limitações no acesso a oportunidades de reservas e exploração remota, além dos riscos resultantes da volatilidade do mercado. Como resultado, a empresa conseguiu minimizar custo e separações forçadas, reduzir os ciclos de integração de 6 meses para 2 a 8 semanas e alcançou flexibilidade e velocidade na mudança de processos de negócio [SOA-CONSORTIUM, 2008b].

Embora a arquitetura ofereça muitos benefícios, a complexidade na implantação de SOA é um consenso, pois abrange atividades diversas como a análise de softwares de apoio (barramentos de serviços, servidores, etc.), integração de aplicações com processos de negócio, definição de processos para modelagem, análise, projeto, implementação, manutenção, monitoramento e gestão dos serviços [PAPAZOGLU *et al.*, 2007]. O ciclo de vida da engenharia de software tradicional não se aplica diretamente a SOA. Novos papéis arquiteturais e novas tarefas de desenvolvimento são introduzidos para tratar as características de desenvolvimento orientado a serviços [GU e LAGO, 2007].

Portanto, o desenvolvimento de uma abordagem sistemática para o desenvolvimento de serviços torna-se fundamental [PAPAZOGLU *et al.*, 2007; GU e LAGO, 2007; ERL, 2005]. Os autores [KLÜCKMANN 2007] e [JOSUTTIS, 2007] destacam a necessidade do conhecimento dos processos de negócio, sendo uma

tendência a identificação de serviços a partir de seus modelos. Muitas empresas que já possuem seus processos modelados podem usufruir deste produto e utilizá-lo como ponto de partida para uma iniciativa SOA. Por outro lado, modelar os processos de negócio da organização permitirá aos analistas detalhar seus conhecimentos em relação ao domínio, tornando-os mais preparados a fornecer boas soluções. A identificação de serviços a partir dos modelos de processos de negócio é o principal assunto deste trabalho e visa auxiliar os profissionais envolvidos na implantação de uma abordagem SOA [AZEVEDO, 2009d].

## **1.2 Objetivo**

Este trabalho apresenta a automatização do método de identificação de serviços candidatos a partir de modelos de processos de negócio, proposto em [AZEVEDO *et al.*, 2009a, 2009b, 2009c]. Serviço candidato é uma abstração (não implementada) de serviço em tempo de projeto, que pode vir a ser implementada como um serviço físico (por exemplo, *Web Service*) ou como uma funcionalidade de uma aplicação [ERL, 2005]. A identificação de serviços candidatos resulta em um conjunto de informações que vão auxiliar o arquiteto de serviços a analisar e projetar serviços físicos. A proposta foi implementada utilizando *ARIS Script Report* na ferramenta ARIS da IDS-Scheer. Esta proposta foi avaliada em um estudo de caso fictício, mas com modelos em um nível de detalhes e complexidade que representa a abstração de um domínio próximo ao real.

## **1.3 Estrutura do texto**

Este trabalho está dividido da seguinte forma. O capítulo 1 apresenta a motivação e contextualiza o assunto deste trabalho. Os capítulos 2 e 3 apresentam os conceitos de arquitetura orientada a serviços e modelagem de processos de negócio, respectivamente. Estes correspondem aos principais conceitos deste trabalho. O capítulo 4 apresenta a metodologia para identificação de serviços candidatos, enquanto o capítulo 5 apresenta a automatização desta metodologia. Por fim, o capítulo 6 apresenta os resultados do estudo de caso e o capítulo 7 apresenta a conclusão do trabalho.

## Capítulo 2: ARQUITETURA ORIENTADA A SERVIÇOS

---

O objetivo deste capítulo é apresentar a Arquitetura Orientada a Serviços (SOA). Serão abordados todos os conceitos básicos necessários para a compreensão do paradigma introduzido por esta arquitetura.

### 2.1 Definição de SOA

Existem muitas definições diferentes sobre SOA na literatura, uma vez que os projetos em Sistemas de Informação possuem particularidades únicas, e isso permite que a aplicação prática de SOA resulte em diferentes visões de acordo com as experiências de cada profissional e/ou pesquisador. Entretanto, é visível que todas as definições concordam que SOA é abstrato e basicamente serve como modelo/paradigma para direcionar o desenvolvimento e manutenção de softwares baseados em processos corporativos bem definidos.

Segundo [JOSUTTIS, 2007], “A arquitetura orientada a serviços (SOA) é um paradigma para a realização e a manutenção dos processos corporativos que se encontram em grandes sistemas distribuídos.”

“SOA não é uma arquitetura concreta: é algo que *conduz* a uma arquitetura concreta. Você pode chamá-la de estilo, paradigma, conceito, perspectiva, filosofia ou representação. Ou seja, SOA não é uma ferramenta ou *framework* que se possa comprar. É uma abordagem, uma maneira de pensar, um sistema de valores que leva a certas decisões concretas quando se projeta uma arquitetura concreta de software.”  
[JOSUTTIS, 2007]

Segundo o modelo de referência OASIS,

“A Arquitetura Orientada a Serviço (SOA) é um paradigma para organização e utilização de competências distribuídas que estão sob controle de diferentes domínios proprietários.”  
[OASIS, 2006]

Segundo Norbert Bieberstein,

“Arquitetura orientada a serviços é uma estrutura (*framework*) para integrar processos de negócio e a infraestrutura de TI que o suporta, na forma de componentes seguros e padronizados – serviços – que podem ser

reutilizados e combinados para endereçar as mudanças de prioridade do negócio.” [JUNIOR, 2007 *apud* BIEBERSTEIN *et al.*, 2006]

Segundo Judith Hurwitz,

“Arquitetura orientada a serviços é uma arquitetura de software voltada para a construção de aplicações que implementam processos de negócio ou serviços utilizando um conjunto de componentes “caixa-preta”, fracamente acoplados, e orquestrados para prover um nível de serviço bem definido.” [HURWITZ *et al.*, 2007]

Apesar de existirem diferentes definições escritas por autores conhecidamente especializados no assunto, todas elas estão de acordo com as características de SOA. A Figura 3 é um slide de apresentação da IBM – Arquitetura Orientada a Serviços - Infraestrutura para a Inovação [NETO, 2006], e esclarece que as diferentes definições para o mesmo assunto na verdade são oriundas de diferentes visões e/ou experiências que as pessoas possuem a partir do contato com a arquitetura.

## Afinal, o que é SOA?

- Diferentes interpretações, dependendo do interlocutor...



“SOA é uma tecnologia que cria um ambiente de negócio ágil e provê vantagem competitiva ou maior valor.”



“SOA é conjunto de processo, estrutura e diretrizes de governança que permite alinhar TI às necessidades do negócio.”



“SOA é uma arquitetura de software baseada em padrões abertos que permite integrar aplicações novas e existentes.”



“SOA é um *framework* baseado em *webservices* que permite invocar objetos remotamente utilizando protocolo SOAP, baseado em XML.”

Figura 3: Definições de SOA segundo perspectivas distintas [NETO, 2006]

## 2.2 Serviços

A definição do termo “serviço” é muito discutida, uma vez que podem ser consideradas definições distintas de acordo com o escopo empregado. Seu

significado no ponto de vista do negócio representa as funcionalidades providas pela empresa para seus clientes e parceiros, por exemplo, um serviço de saque, um serviço de abertura de contas [JUNIOR, 2007].

Quando falamos no ponto de vista da TI, trata-se de um componente de aplicação cujas funcionalidades estão disponíveis para outros sistemas ou usuários [JUNIOR, 2007]. No entanto, existem opiniões divergentes sobre a definição de serviço dentro do escopo TI. Para JOSUTTIS [2007], “... pode-se considerar basicamente serviço como sendo uma representação da TI de uma funcionalidade de negócio independente, tal como “criar um cliente”, “obter contratos de um cliente”...”.

Na utilização dos serviços existem duas entidades fundamentais: *provedor* e *consumidor*. O provedor é aquele que oferece e executa o serviço em resposta a uma requisição do consumidor, que por sua vez é aquele que se utiliza do serviço para obter a funcionalidade processada pelo provedor.

Os serviços são a base da implementação de SOA e, consequentemente, possuem diversos atributos que caracterizam as ideias empregadas na arquitetura as quais permitem alcançar os objetivos esperados.

### **2.2.1 Interfaces e contratos**

Serviços são disponibilizados aos clientes através de interfaces. Assim como na programação orientada a objetos busca-se “esconder” a complexidade do código e não permitir acesso à estrutura interna, o objetivo da interface em SOA é o mesmo: esconder e não permitir acesso à implementação do serviço [FURTADO *et al.*, 2009].

Segundo [JOSUTTIS, 2007], “... interface é uma *assinatura*, que descreve parâmetros de entrada, parâmetros de saída e possíveis exceções.”.

A utilização correta da interface necessita do conhecimento do comportamento e semântica do serviço, uma vez que a definição do serviço acessado através da interface pode ter significados diferentes para os clientes que o acessam. Por isso, cada interface possui um contrato, que contém todas as informações relevantes para a correta utilização do serviço.

Um contrato é a especificação completa de um serviço entre um fornecedor específico e um consumidor específico. Do ponto de vista do consumidor, isso define “tudo o que precisa saber quando usar esse serviço”, de tal forma que (preferencialmente) não fique nenhuma dúvida. [JOSUTTIS, 2007]

A utilização de interfaces remete ao conceito do reuso que, segundo [JUNIOR, 2007], é um dos pilares da SOA, pois é ele que possibilita o ganho de velocidade na construção de novas aplicações, a redução dos custos e aumento da qualidade, através do reaproveitamento de componentes prontos, testados e confiáveis.

Considerando a definição do reuso e as características de “encapsulamento” da interface em SOA, chega-se à definição de reuso “caixa-preta”:

“... o reuso *caixa-preta* visa eliminar a necessidade do desenvolvedor de um conhecimento da implementação de algum componente de software que fará parte do processo de reuso. Em vez disso, o reuso caixa-preta se dá através da descrição de interfaces ou contratos bem definidos que devem ser respeitados pela implementação a ser elaborada. O esforço é sempre usado na nova implementação e nunca ocorre um desperdício tentando entender implementações de terceiros.” [MACHADO, 2004]

### **2.2.2 Outras características dos serviços**

Ao modelar um serviço, deve-se buscar um nível de abstração de forma a torná-lo *independente*. Sabe-se, que é quase impossível na prática, definir serviços sem que ao menos alguns deles possuam dependências em tipos de dados fundamentais, entretanto, é necessário cuidado com tipos mais complexos. O objetivo é minimizar as dependências de tal forma que SOA seja apropriada para sistemas distribuídos com diferentes proprietários [JOSUTTIS, 2007].

O nível de abstração exigido para o correto funcionamento da arquitetura de serviços exige um estudo do uso da aplicação para decidir sobre o nível de *granularidade* a ser usado. As chamadas a serviços, quando comparadas, por exemplo, a procedimentos remotos armazenados, pode custar de 5 a 10 vezes mais tempo para resposta [JOSUTTIS, 2007]. Por isso, a transferência imediata de todos os dados úteis para o consumidor em uma única chamada – *granularidade grossa* -

pode oferecer melhor desempenho do que a realização de vários acessos requisitando os mesmos dados.

A aplicação da granularidade visa o melhor desempenho dentro da arquitetura orientada a serviços, porém, é fácil perceber que a implementação equivocada desta técnica resultará em desperdício no processamento de dados que não serão utilizados.

Quando um serviço é acessado, processado e retorna uma resposta imediata, finalizando sua instância, ele é considerado *sem estado*. Por exemplo, a solicitação de saldo da conta bancária é um serviço que retorna um valor imediatamente após a consulta, encerrando sua instância. Para realizar uma nova consulta, será necessário chamar novamente o serviço.

Um serviço *com estado* é aquele que mantém sua “instanciação viva” por um momento maior, aguardando um evento. Por exemplo, um serviço que represente um carrinho de compras em um site de vendas permanece instanciado por um período indeterminado, sofrendo modificações sempre que um cliente inclui ou exclui itens até seu cancelamento ou finalização [JOSUTTIS, 2007].

As organizações costumam ter serviços críticos os quais necessitam de infraestrutura especial para receber milhares de solicitações por dia. Esses serviços possuem uma probabilidade maior de sofrer falhas como, por exemplo, de comunicação. Um serviço é solicitado, processado e respondido, mas por algum motivo a resposta não alcança o solicitante. Isso pode gerar um grave problema, mas pode ser contornado caso este serviço seja *idempotente*. Isso significa que caso ocorra algum problema na chamada de serviço, ela poderá ser reenviada sem que isso gere outros problemas [JOSUTTIS, 2007].

Os serviços idempotentes são aliados na prevenção de falhas, entretanto, nem sempre é possível atingir esse objetivo, já que vão existir casos difíceis de implementar e por conseguinte, tornam-se soluções caras.

Em SOA, os serviços devem ser implementados apenas uma vez de forma que possam ser reutilizáveis e assim todos os sistemas que precisem de certa funcionalidade chamem o mesmo serviço [JOSUTTIS, 2007]. Entretanto, o reuso tem suas limitações quanto à questão de *desempenho* que devem ser consideradas.

Portanto, a reusabilidade não deve ser considerada como regra, mas como um objetivo.

Os serviços também podem ser *compostos*, ou seja, um serviço pode chamar outro, permitindo que uma funcionalidade de negócio maior possa ser composta por outros serviços.

Utilizando-se destas características básicas citadas, a ideia de serviço pode ser considerada a principal responsável pela inovação e melhorias introduzidas pela SOA. Uma das formas de identificação de serviços é através de análises de processo de negócio, o que é tratado neste trabalho. Isso nos leva às definições do gerenciamento de processos de negócio (BPM).

## 2.3 Web Services

Como apresentado anteriormente, serviços podem ser definidos como programas modulares, geralmente independentes e autodescritivos que podem ser localizados e invocados através da internet ou de uma intranet corporativa [FURTADO *et al.*, 2009]. A arquitetura de *Web Services* é a mais utilizada para implementar serviços em uma arquitetura SOA, já que é baseada em um conjunto padrões que permitem a interoperabilidade [JOSUTTIS, 2007].

Os padrões de desenvolvimento utilizados em *Web Services* permitem que funções de negócios possam ser invocadas remotamente. Muitos analistas, fabricantes e autores hoje em dia recomendam apenas uma maneira apropriada para realizar um ambiente SOA: com *Web Services* [JOSUTTIS, 2007].

A plataforma básica dos *Web Services* é HTTP com XML. O HTTP é um protocolo amplamente utilizado na internet. O XML provê uma linguagem que pode ser utilizada em diferentes plataformas e linguagens de programação e mesmo assim disponibilizar mensagens e funções complexas [FURTADO *et al.*, 2009]. Também é o principal responsável pela interoperabilidade entre sistemas, uma vez que ela é a linguagem mais utilizada para troca de mensagens dentro de SOA.

Os elementos padrões dos *Web Services* são SOAP, UDDI e WSDL. O WSDL (*Web Services Description Language* – Linguagem de Descrição de Serviços



Web) é uma linguagem baseada em XML utilizada para descrever *Web Services*. Trata-se de um documento escrito em XML que além de descrever o serviço, especifica como acessá-lo e quais são as operações ou métodos disponíveis [FURTADO *et al.* 2009].

O UDDI (*Universal Description, Discovery and Integration* – Descrição Universal, Descoberta e Integração) é um protocolo baseado em XML que provê um diretório distribuído com listas de negócios na Internet e descoberta destes serviços [FURTADO *et al.*, 2009]. Através da interface do UDDI é possível pesquisar os serviços publicados e obter as informações necessárias para acessá-los.

O SOAP (*Simple Object Access Protocol* – Protocolo Simples de Acesso a Objeto) é independente de plataforma e independente de implementação. Permite baixo acoplamento entre requisitante e provedor e permite comunicação entre serviços de diferentes organizações [FURTADO *et al.*, 2009].

## 2.4 Ciclo de desenvolvimento

Serviços são pedaços de software como qualquer outro software [JOSUTTIS, 2007], logo, o ciclo de vida usual de desenvolvimento de software também é aplicável ao ciclo de vida de desenvolvimento de serviços. No entanto, existem algumas diferenças.

Um serviço é parte de um processo de negócio mais geral. Por isso, qualquer modificação da modelagem ou implementação de um serviço pode ter impacto em outros sistemas. A modelagem inicial dificilmente não passará por modificações, já que a pressão do mercado leva a situações onde não há tempo para realizar uma modelagem sólida. Além disso, até os melhores projetistas cometem erros e, em grandes sistemas, quase nunca existem pessoas que sabem “tudo” a respeito de um assunto. Segundo JOSUTTIS [2007], menos de 50% das interfaces de serviços projetadas mantém-se estável durante a fase de implementação.

Devido às modificações naturais que ocorrem durante o ciclo de desenvolvimento dos serviços, uma abordagem do tipo cascata não funcionaria. Por isso, o desenvolvimento normalmente deveria utilizar um processo iterativo, com os

resultados das fases anteriores ajustados de acordo com as experiências encontradas nos estágios posteriores [JOSUTTIS, 2007].

As características específicas do ciclo de desenvolvimento de serviços estão na fase de identificação de serviços e na fase de manutenção dos serviços. Como os serviços são partes de um processo de negócio, um cenário típico que leva à criação de um novo serviço é a identificação de novos serviços a partir de modelos de processos de negócio. A automatização da identificação de serviço a partir de modelos de negócio é o principal assunto deste trabalho (Figura 4).

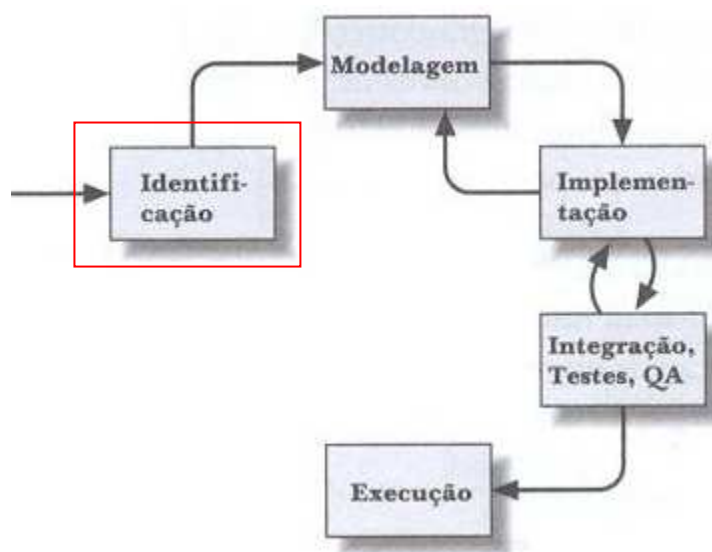


Figura 4 – Ciclo de vida de serviços contendo a fase de identificação de serviços [JOSUTTIS, 2007]

No entanto, esta não é a única abordagem para identificar novos serviços. Por exemplo, outras abordagens podem ser citadas como o levantamento de requisitos tradicional (entrevistas com usuários e analistas do negócio, ou seja, entrevistas com especialistas do negócio), a partir de levantamentos feitos em desenvolvimento de sistemas já orçados, a partir da análise de entidades principais do negócio, a partir de sistemas já implementados e mesmo utilizando serviços já implementados [MARKS e BELL, 2006].

A fase de manutenção de serviços é diferente da fase de manutenção dos softwares tradicionais. Assim que um serviço esteja em produção, ele pode ser usado em cenários de missão crítica e processo de negócio. Isso significa que modificações nos serviços existentes podem levar a modificações nos processos que são apoiados pelos mesmos.

Portanto, a melhor forma para tratar a mudança do comportamento de um serviço em produção é através da implementação de uma nova versão do serviço, que nada mais é do que um novo serviço. A versão anterior pode permanecer disponível para os clientes que ainda se beneficiam deste serviço, reduzindo possíveis impactos ao negócio.

No caso de correções de *bugs* em serviços, significa que sua alteração não acarretará em impactos na modelagem, mas que o serviço não funciona adequadamente. Logo, neste caso, o próprio serviço deve sofrer a manutenção e uma nova versão do serviço não deve ser criada.

O ciclo de vida do serviço finaliza ao eliminar o serviço que está em produção. Geralmente isso ocorre quando um serviço torna-se obsoleto devido às novas versões que foram criadas para corresponder as novas necessidades (Figura 5).

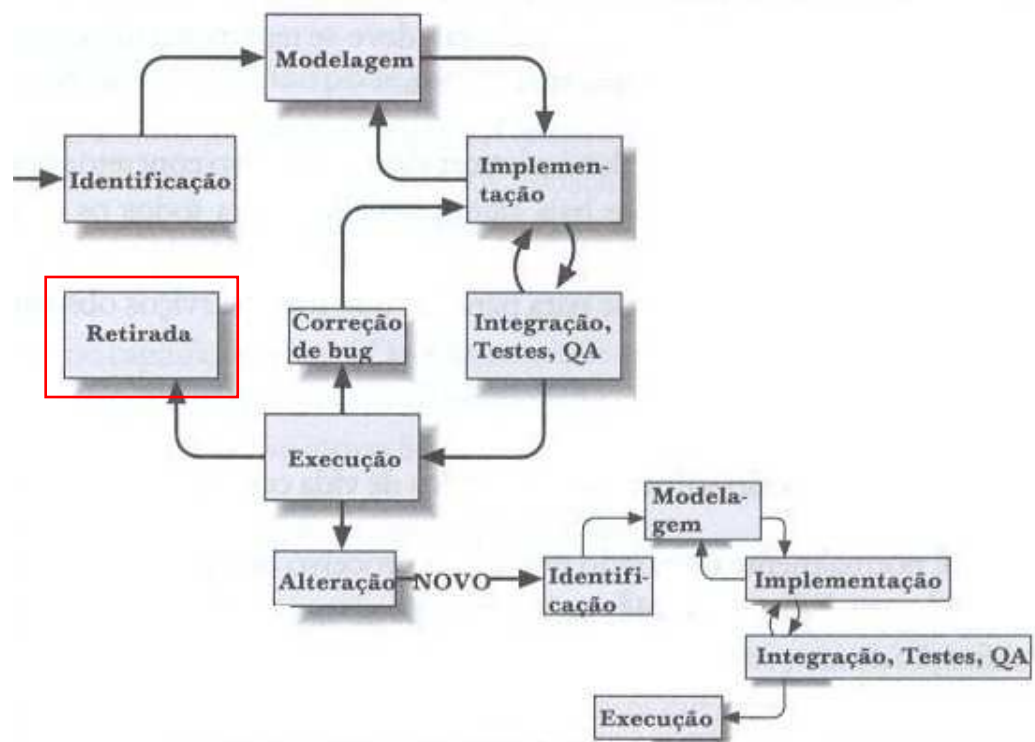


Figura 5- Inclusão da fase de retirada de serviços obsoletos [JOSUTTIS, 2007]

## 2.5 BPM e SOA

Quando se ouve a sigla BPM, é possível criar uma confusão em relação à sua compreensão porque ela é comumente utilizada para definir 2 conceitos: *Business*

*Process Management* - Gerenciamento de Processos de Negócio que, segundo [BLOOMERGSCHMELZER, 2006] significa:

Gerenciamento de processos de negócio normalmente se refere aos meios tecnológicos para a empresa adquirir a visibilidade e o controle sobre os processos de vida longa, processos de múltiplos passos que abrangem um intervalo grande de sistemas e pessoas em uma ou mais organizações. [JOSUTTIS, 2007 *apud* BLOOMERGSCHMELZER, 2006]

Enquanto *Business Process Modeling* - Modelagem de processos de negócio, de acordo com BloomergSchmelzer:

A modelagem de processo de negócio é o conjunto de práticas ou tarefas que as empresas podem executar para descrever visualmente todos os aspectos de um processo de negócio, incluindo o seu curso, controle e pontos de decisão, gatilhos e condições para execução das atividades, o contexto em que uma atividade executa e os recursos associados. [JOSUTTIS, 2007 *apud* BLOOMERGSCHMELZER, 2006]

A sigla BPM, no contexto deste trabalho representa a modelagem de processos de negócio. Ao se decidir por implementar um projeto de SOA em uma organização, podem ser utilizadas diversas abordagens tradicionais para identificar os serviços que serão construídos e disponibilizados para os clientes. Entretanto, para as organizações que já possuem seus processos de negócio modelados, é possível utilizá-los como insumo para o projeto na fase de identificação de serviços. Caso a organização ainda não possua seus processos modelados, a modelagem de processo de negócio pode ser vista como auxílio para a compreensão do negócio, uma vez que seu mapeamento deverá ser bem detalhado a fim de representar fielmente todo o fluxo de trabalho realizado que permite atingir os objetivos do negócio.

Portanto, ao utilizar esse artefato para derivar serviços, é possível alcançar maior alinhamento da TI com o negócio. Considerando ainda que os serviços são tipicamente parte de um ou mais processos de negócio distribuídos [JOSUTTIS, 2007], torna-se mais clara a relação entre as abordagens de modelagem de processos de negócio e SOA.

## 2.6 Governança em SOA

A Governança de SOA é um fator importante para o sucesso de um projeto SOA. Assim como em qualquer projeto de grande porte, as ações administrativas devem ser executadas, tais como as tomadas de decisão e o acompanhamento das atividades do projeto. Mais do que qualquer outro fator em longo prazo, a governança vai fazer a diferença entre o sucesso e o fracasso de SOA, e a proficiência na governança do ambiente SOA vai distinguir os líderes de TI dos mais lentos [WEBMETHODS, 2006].

Ao tornar a governança uma frente dentro da sua estratégia de implementação de SOA, aumentará muito as chances de sucesso e de atingir um duradouro retorno sobre seus investimentos em SOA [WEBMETHODS, 2006]. O grupo Gartner de pesquisa em tecnologia da informação afirma que a falta de mecanismos de governança de trabalho em médios a grandes (mais de 50 serviços) projetos de SOA, após o projeto piloto, será a razão mais comum para o fracasso do projeto. (Probabilidade 0,8) [WEBMETHODS, 2006 *apud* GARTNER, 2006]

A governança SOA atua em aspectos técnicos e não técnicos do projeto, no entanto, não há dúvidas que governança SOA foca primeiramente em aspectos não técnicos. Analisar os detalhes técnicos sobre a infraestrutura (ESB) e ferramentas como repositórios é necessário, mas se torna inútil se não souber por que usá-los, quando usá-los e quem deve usá-los [JOSUTTIS, 2007].

Segundo [JOSUTTIS, 2007], a governança SOA é uma tarefa que lida com as seguintes questões não técnicas: *Visões, objetivos, caso de negócio e modelo financeiro*.

Estabelecer visões, objetivos e caso de negócio é importante para responder a questão do porquê introduzir SOA. É também importante decidir sobre o modelo financeiro que determina como o *overhead* inicial da introdução de SOA e novos serviços irão se pagar [JOSUTTIS, 2007].

[JOSUTTIS, 2007] define os seguintes aspectos não técnicos abordados na governança SOA:

- **Arquitetura de referência**

É necessária uma referência que demonstre suas decisões arquiteturais fundamentais incluindo a sua tecnologia preferida, padrões de troca de mensagens, metamodelo e assim por diante.

- **Papéis e responsabilidades**

Papéis e responsabilidades precisam ser definidos para quem dirige e se preocupa com as questões da arquitetura. Por exemplo, tem que deixar claro quem toma decisões arquiteturais de SOA, onde termina a responsabilidade do ESB e quem é responsável por fazer as modelagens da solução de alto nível que identificam novos serviços.

- **Políticas, padrões e formatos**

Decisões arquiteturais, artefatos e tecnologias levam a políticas e requisitos, os quais devem ser definidos usando padrões e formatos proprietários.

- **Processos e ciclos de vida**

Junto com papéis, responsabilidades e políticas, você tem que definir processos e ciclos de vida para soluções, serviços e outros.

Além dos aspectos não técnicos, existem os aspectos técnicos da governança que, na prática, ajudam a suportar as questões não técnicas. Segundo [JOSUTTIS, 2007], os aspectos técnicos da governança incluem:

- **Documentação**

A documentação desempenha um papel importante para transparência. Ou seja, ela ajuda a promover todas as questões não técnicas da governança (processos, responsabilidades, políticas e outros).

- **Gerenciamento de serviços**

Para o gerenciamento de serviços e contratos de serviços, ferramentas como repositórios e registros podem ajudar.

- **Monitoramento**

Monitoramento é uma tarefa essencial para verificar regras, políticas, contratos, etc. Por exemplo, você pode monitorar violações de SLA ou monitorar o uso para poder retirar os serviços que não estão mais em uso.

- **Gerenciamento de mudança e configuração**

As ferramentas usuais de gerenciamento de configuração podem também ajudar a gerenciar software SOA, artefatos SOA e documentação SOA.

Portanto, governança é primordial para a implantação de uma Arquitetura Orientada a Serviços. Aspectos técnicos e não técnicos devem ser muito bem entendidos para a introdução desta abordagem em uma organização.

## **2.7 Resumo**

Este capítulo apresentou os conceitos básicos sobre a Arquitetura Orientada a Serviços. SOA é um paradigma que oferece um conjunto de benefícios para a arquitetura de sistemas distribuídos de uma organização. Em contrapartida, o sucesso de um projeto de implementação de SOA demanda alto investimento financeiro, dedicação e governança.

O ciclo de vida dos serviços difere dos ciclos de vida de software tradicionais, devido a alta necessidade de mudanças em seus requisitos para adequação imediata do negócio em relação às necessidades do mercado. Isso resulta no desenvolvimento de novas versões de serviços, que são necessárias para corresponder aos novos requisitos. A versão anterior do serviço permanece disponível para o uso por parte dos clientes que não necessitam da nova versão, o que reduz o impacto na mudança de versão. Ao tornar-se obsoleto, o serviço é eliminado.

No próximo capítulo serão abordados os conceitos básicos da modelagem de processos de negócio, incluindo seus objetivos, benefícios e ciclo de vida. As informações apresentadas neste capítulo e no capítulo 3 são necessárias para facilitar compreensão dos capítulos subsequentes.

## Capítulo 3: MODELAGEM DE PROCESSOS DE NEGÓCIO

---

O objetivo deste capítulo é apresentar a modelagem de processos de negócio descrevendo seus conceitos básicos, definições dos objetos representativos do negócio e detalhamento da notação e ferramenta utilizada no desenvolvimento do trabalho proposto neste documento.

### 3.1 Definição

Todo trabalho importante realizado nas empresas faz parte de algum processo [GONÇALVES, 2000 *apud* GRAHAM e LEBARON, 1994]. Não existe um produto ou um serviço oferecido por uma empresa sem um processo empresarial associado. Da mesma forma, não faz sentido existir um processo empresarial que não ofereça um produto ou um serviço [GONÇALVES, 2000].

Um processo de negócio consiste em um conjunto de atividades que são executadas coordenadamente em um ambiente organizacional e técnico. Estas atividades coletivamente realizam um objetivo do negócio. Cada processo de negócio é representado por uma única organização, porém ele pode interagir com processos de negócio que são executados por outras organizações [WESKE, 2007]. Portanto, é através da execução dos processos de negócio que as organizações realizam seus propósitos [THOM *et al.*, 2007].

A modelagem de processos de negócio é o conjunto de práticas ou tarefas que as empresas podem executar para descrever visualmente todos os aspectos de um processo de negócio, incluindo o seu curso, controle e pontos de decisão, gatilhos e condições para execução das atividades, o contexto em que uma atividade executa e os recursos associados [JOSUTTIS, 2007 *apud* BLOOMERGSCHMELZER, 2006].

Um modelo é apenas uma representação do processo que permite às empresas documentar, simular, compartilhar, implementar, avaliar e continuamente melhorar suas operações [JOSUTTIS, 2007 *apud* BLOOMERGSCHMELZER, 2006].



## 3.2 Objetivos

A modelagem de processo de negócio consiste no registro das informações do negócio em modelos, podendo ser armazenados em papel ou em meio computacional. Ao se analisar um modelo de processo de negócio, é possível acessar informações que respondem a questões relacionadas ao negócio (Onde (*Where*)? Quando (*When*)? Quem (*Who*)? Por quê (*Why*)? O que (*What*)? e Como (*How*)? - 5W2H - Figura 6) a partir de um dado contexto.

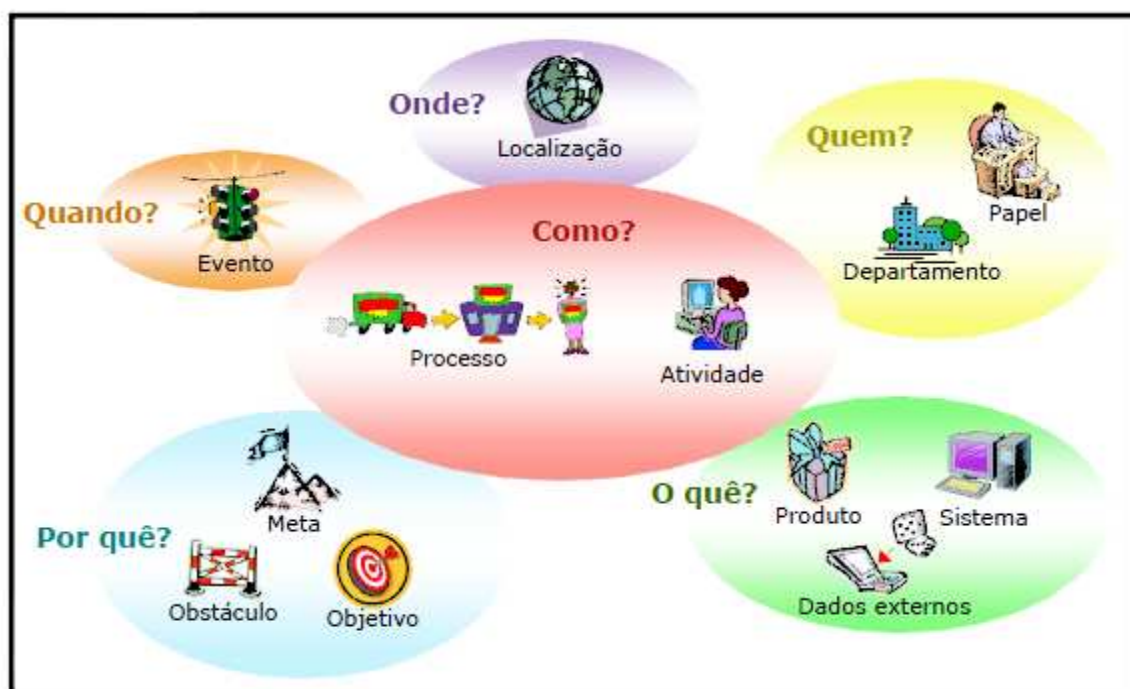


Figura 6 – Objetivos da modelagem de processos de negócio [ARAÚJO, 2008]

Com isso, a modelagem de processos de negócio pode descrever detalhadamente o fluxo de atividades dos processos de uma organização, incluindo informações como papéis executores, sistemas de apoio, informações de entrada e saída, regras de negócio, requisitos de negócio, portadores de informação, e vários outros elementos que agregam algum valor ao negócio [SHARP e MCDERMOTT, 2009].

Outros objetivos podem ser almejados em um projeto de modelagem de processos de negócio, que serão alcançados através dos benefícios gerados pela modelagem. [DAVIS, 2002] enumera alguns dos principais benefícios da modelagem:

- Introduzir rigor e método
- Fornecer um único e consistente registro
- Integrar processos, sistemas, organizações, informações e dados
- Permitir verificar e analisar os relacionamentos
- Prover múltiplos pontos de vista
- Permitir validação, acompanhamento e testes
- Fornecer um meio ideal para avaliar cenários do tipo “E se...”
- Fornecer uma plataforma para rápida engenharia de processo

A falta de conhecimento dos processos da organização também será eliminada com a modelagem do negócio, abrindo oportunidades para melhorias através da gestão de processos. Ainda é possível ser aplicada a transparência dos processos da organização aos seus colaboradores, sejam funcionários ou empresas externas, auxiliando a padronização e comunicação com processos externos.

### **3.3 Componentes da modelagem de processos de negócio**

Um projeto de modelagem de processo de negócio necessita de um conjunto de componentes que deverão ser definidos antes de seu início. Esses componentes viabilizarão a modelagem de processo e irão definir os passos necessários para concluir com sucesso o projeto de modelagem. A Figura 7 ilustra os 4 componentes necessários para um projeto de modelagem de processos de negócio. Esses componentes são: método, metamodelo, notação e ferramenta.

O método define os passos que guiarão o projeto durante sua execução, ou seja, o conjunto de atividades que serão realizadas durante o projeto e que permitirão alcançar o sucesso na modelagem dos processos [ARAÚJO, 2008].

A notação define o conjunto de símbolos que representarão as informações modeladas e as regras de modelagem que orientarão os modeladores ao projetar o negócio para os modelos de forma padronizada [ARAÚJO, 2008].

A ferramenta é o software que fornecerá o suporte computacional durante o projeto de modelagem. Diversas ferramentas com diferentes características estão disponíveis no mercado, sendo que algumas apenas oferecem recursos gráficos enquanto outras possuem recursos de apoio mais sofisticados, como simulação do processo. O escopo do projeto deverá definir os requisitos ferramentais [ARAÚJO, 2008].

O metamodelo define o conjunto de informações do negócio que deverão ser modeladas. Ele insere um limite para os conceitos que devem ser levantados durante o projeto. Esses conceitos referem-se, por exemplo, a papéis, sistemas de apoio, regras de negócio, requisitos de negócio, termos técnicos, unidades organizacionais, ou ainda qualquer outro componente do negócio [ARAÚJO, 2008].

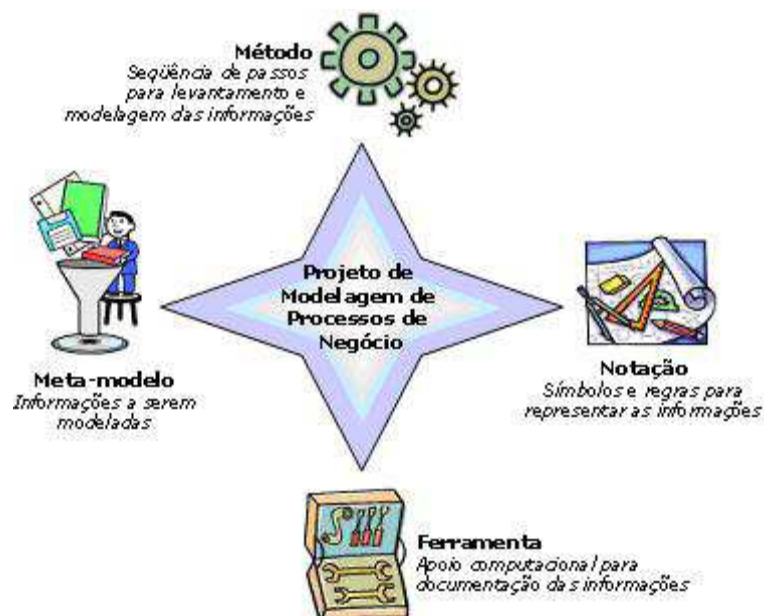


Figura 7 – Componentes da modelagem de processos de negócio [ARAÚJO, 2008]

As subseções seguintes detalham os componentes que devem ser definidos em um projeto de modelagem de processos de negócio.

### 3.3.1 Método

O método de modelagem de processos apresentado neste trabalho é uma adaptação da “metodologia do projeto de três fases”, proposto por Sharp e McDermott no livro *Workflow Modeling* [SHARP e MCDERMOTT, 2009]. As alterações inseridas no método original (fase 4) enfatizam a necessidade de uma

gestão contínua dos processos de negócio, visando melhorias e/ou adaptações de suas atividades para o benefício da organização.

O método é composto por 4 fases, onde cada fase é composta por um conjunto de atividades que produzirão artefatos que servirão para as fases subsequentes. Na primeira fase é desenvolvido o escopo geral do projeto para identificar processos e definir limites. Na segunda fase são modelados os processos atuais da organização, ou seja, a forma exata de como as atividades são executadas no presente. Na terceira fase são realizados estudos para melhoria dos processos e ao final, projetado um novo modelo com as alterações necessárias. Na quarta fase, são realizadas as ações necessárias para implantar as mudanças de melhoria nos processos. Depois de implantado, o novo processo é reavaliado podendo retornar à fase 3 para realizar nova projeção do processo.

Este ciclo permite à organização gerenciar seus processos realizando o estudo e a projeção de melhorias continuamente, o que permitirá respostas rápidas às mudanças do mercado, e aumentará sua competitividade perante as outras organizações do ramo. A Figura 8 ilustra as fases do método.

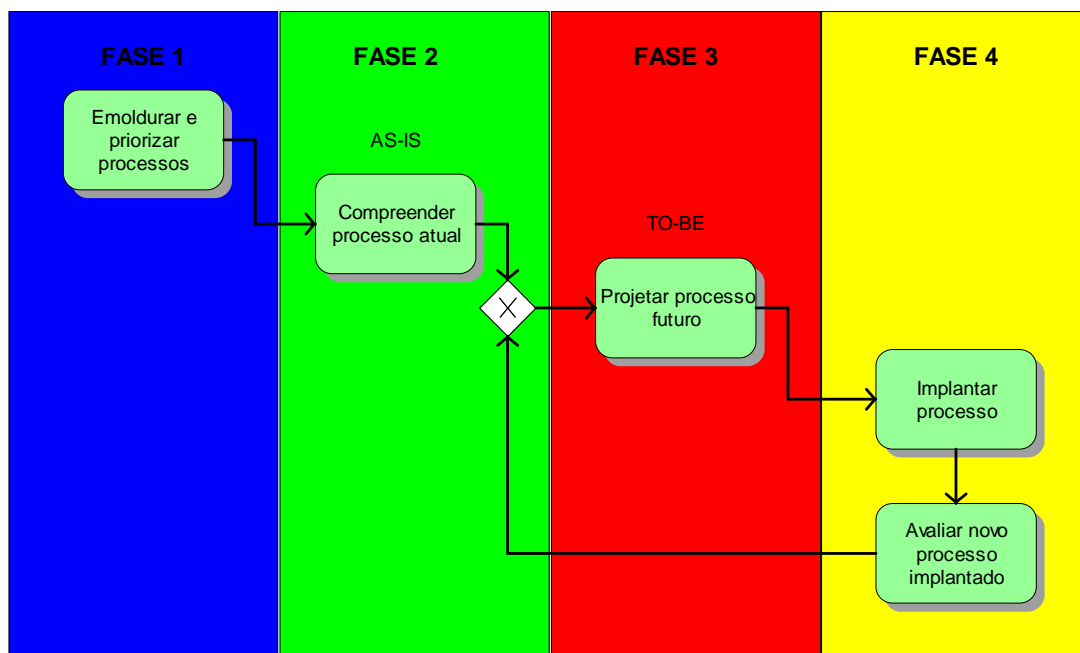


Figura 8 – Método para gestão de processos de negócio (Adaptação [ARAÚJO, 2008])

As subseções seguintes detalham as atividades que devem ser realizadas em cada fase do método.

### 3.3.1.1 Emoldurar processos

A primeira fase do projeto de modelagem de processos de negócio visa adquirir conhecimento sobre os macroprocessos que viabilizam o atendimento aos objetivos estratégicos da organização. Esta é sem dúvida a fase mais importante, porque, embora rápida, evita muitos problemas comuns depois [SHARP e MCDERMOTT, 2009]. A seguir encontra-se o resumo das atividades da fase “Emoldurar processos”, segundo [SHARP e MCDERMOTT, 2009]:

- Identificar e nomear o conjunto de processos relacionados através do desenvolvimento de um mapa geral de processos ou desenho do processo.
- Estabelecer o escopo dos “processos alvo” que serão estudados, utilizando um *framework* para esclarecer o conteúdo e limites do processo.
- Documentar a missão, estratégia, metas e objetivos da organização.
- Desenvolver uma estimativa inicial do processo.
- Determinar objetivos do processo e desempenho almejado.
- Desenvolver glossário de termos e definições.
- Resumir todo o conteúdo, construir e distribuir um pôster ou algo como um resumo.
- Opcionalmente, iniciar a documentação de coisas significantes, observações da cultura organizacional, competências centrais, e sistemas de gerenciamento.

### 3.3.1.2 As-is

Com o escopo do projeto definido e os macroprocessos da organização elucidados, o próximo passo é mapear os processos da forma que estão sendo executados no presente. É importante frisar que pode ocorrer de o processo real implementado na organização estar registrado apenas no conhecimento tácito dos colaboradores, mesmo que haja um processo definido a seguir,

O objetivo aqui não é documentar exaustivamente o processo corrente em seus mínimos detalhes, mas documentar o suficiente para permitir que o modelo seja avaliado e compreendido [SHARP e MCDERMOTT, 2009]. A seguir encontra-se o resumo das atividades da fase “*as-is*”, segundo [SHARP e MCDERMOTT, 2009]:

- Mapear o fluxo de trabalho do processo corrente mostrando os papéis, o que fazem e quando realizam suas atividades.
  - Desenvolver diagramas *swimlane*, focando em um fluxo generalizado.
  - Refinar os diagramas *swimlane*, adicionando alternativas, exceções, erros, e assim por diante.
  - Utilizar níveis de detalhe progressivos, parando quando o comportamento do processo for compreendido – não perder tempo com detalhes!
- Documentar observações importantes sobre todos os habilitadores (usuários corrente da TI, motivações e mensurações, e assim por diante) assim como as observações sobre a cultura, competências essenciais e sistemas de gerenciamento.
- Gravar pensamentos iniciais sobre pontos fortes e pontos fracos do processo em vigor, especialmente pontos de alavancagem onde é possível uma melhoria significativa.

### 3.3.1.3 To-be

Na fase *to-be* serão projetadas melhorias nos processos modelados na fase *as-is*, tais como redução de *handoffs* (linhas que cruzam a linha das raias entre os papéis executores do processo), eliminação de atividades que comprometem o desempenho do processo, automatização de atividades e outras ações que implicarão na redução de custos do processo, redução do tempo de execução, redução e controle de falhas, entre outros. O que vai ser projetado dependerá dos objetivos que se deseja alcançar.

A implementação da fase *to-be* é dividida em dois estágios. O primeiro, “Caracterizar o processo *To-be*”, conduz a uma detalhada avaliação do modelo *as-is*

e utilizá-lo como ajuda para determinar um conjunto de melhorias ou características do desenho que irão trabalhar em conjunto para atingir as metas do processo. Nós fazemos esta fase especialmente porque muitas vezes as equipes saltam para a concepção de um novo fluxo de trabalho que incorpora “melhorias” que na verdade trabalham com objetivos opostos [SHARP e MCDERMOTT, 2009]:

### **Caracterizar o processo To-be**

- Realize uma avaliação final do processo *as-is* baseado nos habilitadores
  - Os habilitadores são: *workflow*, TI, motivações e mensurações, recursos humanos, políticas e regras, facilidades (ou outros).
  - Neste ponto, são coletadas ideias para o processo *to-be*. Dado que esta etapa envolve a avaliação do *as-is* para ajudar a gerar ideias para o *to-be*, esta fase pode ser considerada como uma fase a parte entre a compreensão do *as-is* e a projeção do *to-be*.
- Decidir sobre a direção - abandonar, permanecer com o *as-is*, melhorar, redesenhar ou terceirizar.
- Opcionalmente, avalie cada etapa individual.
- Desenvolva ideias adicionais por características ou recursos do novo processo, para aumentar o número de ideias identificadas durante a avaliação final do “*as-is*”. Esta etapa inclui:
  - Identificar melhorias que podem endereçar pontos de alavancagem.
  - Pressupor desafios às novas etapas do processo subjacente.
  - *Brainstorm* por habilitador (isto pode, de fato, ter sido feito durante a avaliação).
- Rejeitar algumas ideias caso elas sejam impraticáveis, ilegais, conflitantes com a cultura ou direção, e assim por diante.
- Avaliar ideias promissoras no contexto (com respeito aos outros habilitadores) utilizando um formato de matriz (por exemplo, uma mudança no *workflow* pode exigir uma mudança na definição da função

que, por sua vez, exige mudanças no recrutamento e treinamento, como também na remuneração e avaliação de desempenho.

- Baseado na avaliação, selecione as principais características do processo *to-be* e as documente.
- Desenvolva/revise o modelo conceitual de dados.

### **Projetar o fluxo de trabalho do processo To-be**

A segunda parte é realmente simples, se todas as etapas anteriores foram trabalhadas, porque os elementos centrais do novo fluxo de trabalho terão sido identificados durante a avaliação e documentação das principais características do *to-be*, ou eles serão herdados do *as-is* [SHARP e MCDERMOTT, 2009]:

- Desenhe o fluxo de trabalho *to-be*.
- Avance através dos diferentes níveis de detalhe.
- Avalie e verifique a viabilidade em cada nível.
- Revise ou siga para o próximo nível de detalhe.
- Reitere o trabalho.
- Revise com um público maior.

#### **3.3.1.4 Implantação do processo**

Na fase de implantação do processo coloca-se em prática o novo processo desenvolvido na fase 3. Para isso, deverá ser desenvolvido um plano de implantação prevendo treinamentos, implementação de sistemas e gestão de mudanças. Grandes mudanças podem ser realizadas gradativamente, para minimizar impactos e/ou resistências das pessoas que serão afetadas.

Os riscos envolvidos na implantação do novo processo deverão ser previstos no plano de implantação, bem como ações de contingência e mitigação dos riscos. Momentaneamente, o processo pode sofrer alterações de desempenho até a adaptação completa dos envolvidos.



### **3.3.1.5 Avaliação do novo processo implantado**

Os processos de negócio de uma organização devem estar em constante evolução, seja por seu amadurecimento ou por adaptação às novas necessidades do seu mercado de atuação. Por isso, a avaliação e melhoria dos processos tornam-se um ciclo sem fim definido. O processo *to-be* que foi recém implantado transforma-se em uma visão *as-is* que irá amadurecer pela experiência e se adaptar aos avanços tecnológicos, abrindo espaço a um novo processo *to-be* [ARAÚJO, 2008].

A Fase 1 e a Fase 2 não deverão ser executadas a partir deste ponto, uma vez que neste momento o conhecimento do processo tornou-se consolidado e as novas melhorias tenderão a ser pontuais. Portanto, após a avaliação do novo processo retorna-se à Fase 3, reiniciando os estudos do processo para sua evolução.

### **3.3.2 Metamodelo**

O metamodelo definirá quais serão os elementos que deverão ser incluídos na modelagem de processos de negócio. Ele servirá como referência no projeto para as equipes se orientarem em relação aos limites do detalhamento de informações que devem ser levantadas e quais diagramas devem ser modelados [ARAÚJO, 2008].

O metamodelo definirá, por exemplo, os elementos que devem aparecer nos diagramas, tais como papéis, sistemas de apoio, unidades organizacionais, termos técnicos, regras de negócio e outros. Ele deverá ainda definir quais níveis de abstração do negócio devem ser modelados, por exemplo, cadeia de valor, fluxo de trabalho e nível operacional.

Portanto, o metamodelo servirá como um contrato que explicita o produto de modelagem esperado pelo cliente e definirá os limites em que os modeladores devem atuar no dado projeto.

### **3.3.3 Notação**

Cada elemento do negócio que será mapeado nos modelos de processo de negócio deve possuir seu respectivo símbolo representativo. Os símbolos que representam os elementos podem ser diferentes para cada organização, mesmo

quando representam um objeto que possui a mesma definição. A forma de utilização dos símbolos também pode diferir quando as regras de modelagem são diferentes.

O conjunto de boas práticas de modelagem pode ser frisado na notação ou simplesmente impostos como regra na modelagem dos processos. A padronização da notação permitirá a comunicação entre os modelos de processos das diferentes unidades organizacionais da empresa, modelados por diferentes equipes e também entre seus possíveis parceiros.

Atualmente é possível encontrar um padrão de notação que é bastante utilizado. Essa notação foi criada pela *Object Management Group*, Inc. – OMG (<http://www.omg.org/>) e se chama *Business Process Modeling Notation* – BPMN (<http://www.bpmn.org/>).

A iniciativa de gestão de processos de negócio (BPMI) desenvolveu o padrão *Business Process Modeling Notation* (BPMN). O principal objetivo da BPMN é prover uma notação que seja legível e compreensível por todos os usuários do negócio, desde os analistas de negócio que criam as versões iniciais do processo, aos desenvolvedores técnicos responsáveis por implementar a tecnologia que irá executar os processos, e finalmente, às pessoas do negócio que irão gerenciar e monitorar os processos. Assim, a BPMN criou uma ponte para a lacuna entre os modelos de processos de negócio e a implementação do processo [OMG, 2008].

Esta especificação define a notação e a semântica de um Diagrama de Processo de Negócio (BPD) e representa a incorporação das melhores práticas dentro da comunidade de modelagem de negócio. A intenção da BPMN é padronizar a notação de modelagem de processos de negócio em face às várias notações de modelagem e pontos de vista diferentes. Ao fazer isso, a BPMN proverá um meio simples de comunicação com outros usuários do negócio, implementadores do processo, clientes e fornecedores [OMG, 2008].

Outro objetivo não menos importante, é garantir que as linguagens XML, determinadas para a execução do processo de negócio, tais como BPEL4WS (*Business Process Execution Language for Web Services*), possam ser visualizadas com uma notação orientada ao negócio [OMG, 2008].

### 3.3.4 Ferramenta

A ferramenta utilizada na modelagem de processos de negócio é um software que oferece o suporte computacional para o registro das informações mapeadas do negócio para os modelos e objetos virtuais. A ferramenta geralmente oferece um conjunto de objetos que podem ser instanciados com uma interface gráfica distinta que representará os diferentes elementos do negócio. A partir desses objetos, é possível desenvolver normas para a modelagem dos processos que seja compatível com a ferramenta utilizada.

Os objetos que representam os elementos do negócio também possuem atributos que devem ser preenchidos com detalhes específicos do dado elemento o qual o objeto irá representar. As diferentes ferramentas disponibilizam objetos distintos que podem possuir a mesma definição. Por exemplo, um quadrado pode representar uma regra de negócio na ferramenta A e na ferramenta B ser representada por um círculo. Essa diferença de objetos representativos de uma mesma definição apenas afeta a notação utilizada na modelagem.

Algumas ferramentas mais avançadas possuem uma estrutura de banco de dados que armazena de forma centralizada toda a modelagem realizada. Essas ferramentas costumam ser robustas e recomendadas para grandes empresas.

Outras ferramentas apenas salvam seus projetos em arquivos avulsos que podem ser armazenados facilmente em qualquer mídia. Geralmente seus modelos não podem ser manipulados paralelamente, como no caso das ferramentas que armazenam seus dados em um banco de dados. Algumas ferramentas ainda oferecem outras opções que auxiliam na gestão dos processos, por exemplo, a função de simulação do processo.

A diferença de maior significância entre um método de modelagem baseado em papel e utilizando uma ferramenta é que os modelos não são somente diagramas em um pedaço de papel. Os modelos são compostos por objetos, e tanto os objetos quanto os modelos residem em um repositório eletrônico ou banco de dados. A ferramenta pode criticar estes objetos e modelos; seus atributos e relacionamentos podem ser analisados e relatados. Objetos que representam entidades de negócio comuns podem ser reutilizados em vários modelos e carregar seus dados de atributos

consigo. Muitas pessoas podem visualizar e, quando apropriado, atualizar o mesmo modelo, garantindo uma única fonte para atualização da informação no negócio. Direito de acesso e configurações são gerenciadas pela ferramenta, ao invés de ter que controlar vários documentos. Por causa dos modelos estarem no formato eletrônico, estes podem ser analisados, simulados e testados. Através de interfaces apropriadas, os modelos podem ser publicados na *Web* e torná-los disponíveis a todos [DAVIS, 2002].

### **3.4 Diagramas de processos de negócio**

Os processos de negócio são modelados em níveis de visão bem definidas. Cada nível é representado por um tipo de modelo que possui objetos representativos adequados para sua visão. O nível superior é o mais abstrato, sendo seus inferiores mais específicos. No nível mais abstrato encontram-se os processos de negócio da organização. Cada processo é detalhado por um conjunto de atividades coordenadas que, por sua vez, são detalhadas ao nível de execução, onde é possível encontrar informações operacionais da atividade.

Esta sessão apresenta os diagramas que são utilizados na modelagem de processos de negócio.

#### **3.4.1 VAC – *Valued Added Chain***

O diagrama VAC especifica as funções em uma organização as quais influenciam diretamente o real valor agregado da organização. Estas funções podem ser ligadas a outras, de forma a sequenciar as funções e então formar a cadeia de valor agregado [ARIS, 2006].

O diagrama VAC descreve os processos de negócio do ponto de vista mais abstrato. Cada processo contido no modelo possui um ou mais objetivos que agregam valor que garante a existência do negócio. Um modelo VAC pode ser detalhado em outros macroprocessos. A cadeia de valor do nível mais alto representa o processo de negócio da organização.

Em um diagrama VAC, as funções podem ser dispostas em uma hierarquia ou similar a uma árvore de função. A orientação do processo subordinado ou superior é

sempre ilustrada. Assim como a representação da função subordinada ou superior, um diagrama de cadeia de valor representa as ligações entre as funções e as unidades organizacionais e as funções e objetos informativos [ARIS, 2006].

A Figura 9 exemplifica um modelo VAC. Este modelo possui o macroprocesso “Realizar empréstimos para pessoa física” composto por outros três macroprocessos que estão ligados aos seus respectivos objetivos e indicadores que são extraídos durante a realização do processo. A execução coordenada destes três macroprocessos permitirá ao negócio realizar empréstimos para pessoa física.

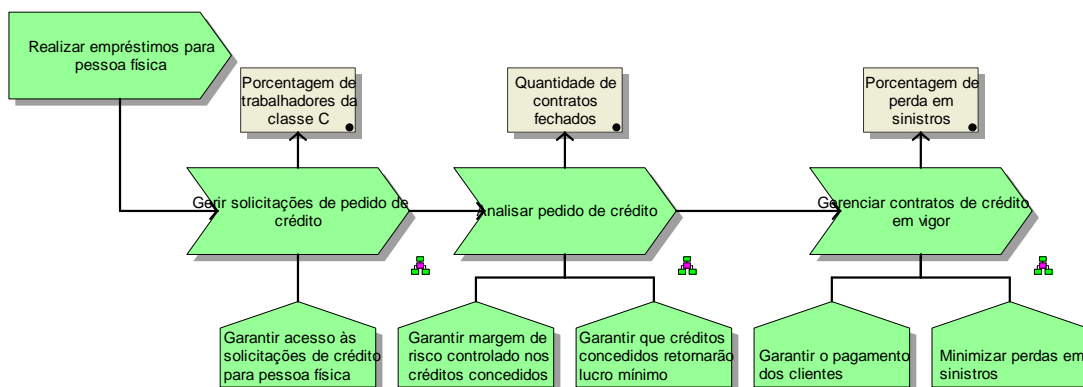


Figura 9 – Exemplo de modelo VAC

A visão de macroprocesso é conhecida como uma visão gerencial devido a sua abstração. Essa visão é moldada durante a primeira fase do ciclo de vida da modelagem de processos de negócio com o propósito de adquirir conhecimento necessário para calcular tempo e custo de um projeto de modelagem de processos do negócio.

### 3.4.2 EPC – *Event-driven Process Chain*

O diagrama EPC é o modelo central para toda a modelagem de negócio. É um modelo dinâmico que traz consigo os recursos estáticos do negócio (sistemas, organizações, dados, etc.) e os organiza para conceber uma sequência de tarefas ou atividades (‘o processo’) que agrega valor ao negócio [DAVIS, 2002]. O modelo EPC pode detalhar tanto um processo previsto na cadeia de valor como uma atividade crítica ou complexa que caracteriza um subprocesso.

O fluxo de trabalho detalhado em um modelo EPC engloba informações como papéis executores e suas unidades organizacionais, raias que organizam as

atividades de acordo com seus papéis executores, elementos de interface com outros processos, eventos que demarcam o início e o fim do processo, eventos intermediários que sinalizam circunstâncias importantes para a continuidade do processo, principalmente nos fluxos de decisão, operadores lógicos para inserir regras no fluxo e as próprias atividades que serão executadas ao longo do processo.

O nível de detalhe em um modelo EPC o difere de um simples modelo de *workflow*, permitindo uma leitura clara das atividades do negócio. Isto auxilia no estudo de melhorias e planejamento de automação das atividades.

A Figura 10 ilustra um exemplo de EPC. Este modelo possui atividades executadas pela área de “Crédito e taxas contratuais” e pelo sistema “Crédito direto”. As atividades de responsabilidade de cada papel estão respectivamente em suas raias. As linhas que cruzam a linha das raias representam *handoffs* entre os papéis executores do processo.

A leitura deste processo pode ser interpretada da seguinte forma: O processo inicia quando a área de “Crédito e taxas contratuais” recebe a proposta de crédito. Então o sistema “Crédito direto” verifica a situação cadastral do cliente e o limite de crédito do cliente. Se o limite de crédito do cliente não for aprovado, o sistema cancela a proposta e a área de “Crédito e taxas contratuais” comunica ao cliente que o limite não foi aprovado, finalizando o processo. Caso o limite de crédito seja aprovado, o sistema “Crédito direto” calcula a alíquota de imposto para cada tipo de imposto e determina a taxa de juros a ser cobrada do cliente. Então a área de “Crédito e taxas contratuais” analisa o contrato. Caso seja identificado que o contrato é de risco, o sistema “Crédito direto” cancela o contrato de risco, finalizando o processo. Caso não haja necessidade de ajustar o contrato, a área de “Crédito e taxas contratuais” aprova o contrato, encerrando o processo. Caso seja identificada a necessidade de ajuste do contrato, a área de “Crédito e taxas contratuais” altera a proposta de crédito, retornando a proposta para uma nova análise. Ao final do processo, o contrato será cancelado ou efetivado.

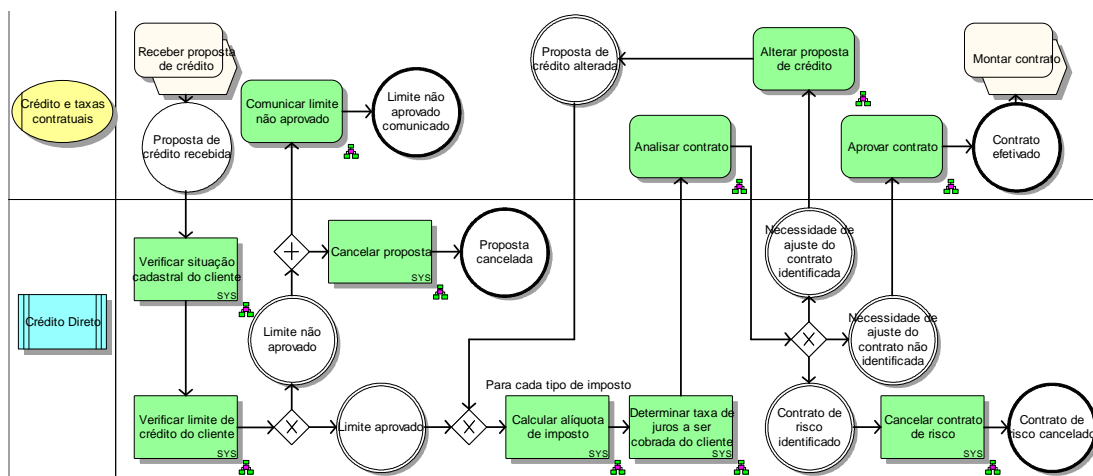


Figura 10 – Exemplo de modelo EPC

O modelo EPC é utilizado para detalhar todos os processos de negócio na cadeia de valor, desde que a modelagem deste nível de detalhamento faça parte do escopo do projeto de modelagem de processos de negócio.

### 3.4.3 FAD – *Function Allocation Diagram*

O diagrama FAD é um modelo que possui informações sobre uma dada atividade do ponto de vista operacional. É utilizado para apresentar uma visão mais detalhada dos recursos disponíveis e necessários, que são relevantes para as atividades. O FAD também é utilizado para reduzir a complexidade dos processos de negócio, representando elementos como: cargos, áreas, transações e sistemas que suportam a atividade, as entradas e saídas de dados, os documentos, os riscos envolvidos nas atividades, entre outras possibilidades que podem ser criadas pelo modelador [BPM-ADVISOR, 2009].

A Figura 11 exemplifica um modelo FAD. Esta atividade é executada pelo sistema “Crédito Direto”, caracterizando uma atividade automatizada. As informações que a atividade consome são “Proposta de crédito”, “Créditos concedidos” e “Cadastro de cliente”, sendo esta última extraída da base de dados “BDCliente”. O resultado da atividade é a informação “Proposta de crédito” atualizada com um status de aprovação que depende da regra de negócio “Verificação de limite de crédito” e que é implementado de acordo com o requisito de negócio “Verificar limite de crédito do cliente”.

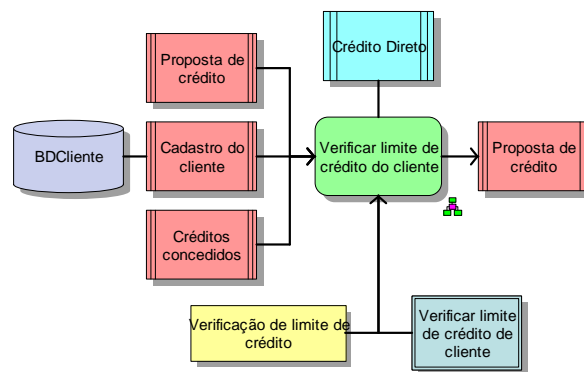


Figura 11 – Exemplo de modelo FAD

O modelo FAD deve ser utilizado para representar todas as atividades de um diagrama EPC, desde que a modelagem deste nível de detalhamento faça parte do escopo do projeto de modelagem de processos de negócio.

### 3.5 Resumo

Este capítulo apresentou os conceitos básicos sobre a modelagem de processos de negócio. Diversos benefícios podem ser alcançados ao modelar os processos de uma organização, uma vez que será possível documentar todos os detalhes do negócio e, a partir disso, realizar uma análise detalhada que permitirá desenvolver melhorias e depois implementá-las.

O ciclo de vida da modelagem de processos de negócio induz ao desenvolvimento de melhorias constantes que permitirá ao negócio adaptar-se às mudanças naturais do mercado com um tempo de resposta menor. Além disso, é possível estabelecer um padrão e a transparência dos processos de negócio da organização.

No próximo capítulo será detalhada a metodologia para a identificação automática de serviços candidatos que utiliza como insumo os modelos de processos de negócio apresentados neste capítulo.



## **Capítulo 4: METODOLOGIA PARA A IDENTIFICAÇÃO AUTOMÁTICA DE SERVIÇOS CANDIDATOS A PARTIR DE MODELOS DE PROCESSOS DE NEGÓCIO**

---

O objetivo deste capítulo é apresentar a metodologia para identificação de serviços candidatos a partir de modelos de processos de negócio. Serão detalhados os conceitos básicos e a metodologia de identificação, incluindo as heurísticas de identificação e consolidação de serviços candidatos.

### **4.1 Identificação de serviços em SOA**

A complexidade da implantação da arquitetura SOA se justifica, pois abrange atividades diversas como a análise de softwares de apoio (barramentos, servidores etc.), integração de aplicações com processos de negócio, definição de processos para modelagem, projeto, monitoramento e gestão dos serviços [PAPAZOGLU *et al.*, 2007]. KLÜCKMANN [2007] e JOSUTTIS [2007] destacam a necessidade do conhecimento dos processos de negócio, sendo uma tendência a identificação de serviços a partir de seus modelos.

A modelagem de processos de negócio descreve detalhadamente as atividades envolvidas em um processo, incluindo a tecnologia empregada, as informações consumidas e geradas e os envolvidos nas execuções das tarefas. Uma organização pode ter seus processos completamente transcritos para modelos, permitindo realizar análises minuciosas em informações que dificilmente poderiam ser percebidas ou compreendidas de outra forma que não fosse a partir da documentação dos processos de negócio. Considerando ainda que os serviços são tipicamente parte de um ou mais processos de negócio distribuídos [JOSUTTIS, 2007], torna-se mais clara a relação entre as abordagens de modelagem de processos de negócio e SOA.

Em um repositório que contenha uma grande quantidade de modelos de processos, a tarefa de identificação de serviços pode se tornar muito complexa de ser realizada manualmente. Assim, métodos que possam ser automatizados para analisar modelos de processos de negócios e identificar serviços são relevantes de forma a

facilitar o processo de análise e direcionar a implementação de serviços. [KOHLBORN *et al.*, 2009] discutem a importância de um método de identificação de serviços a partir de modelos de processos de negócio e avaliam diversas abordagens para identificação de serviços, as quais não são detalhadas e sistematizadas o bastante para serem automatizadas.

## 4.2 Método de identificação de serviços candidatos

A metodologia de identificação de serviços candidatos a partir de modelos de processos de negócio é realizada através de uma análise *top-down* dos modelos de processos. Essa análise identifica os serviços candidatos baseado nos objetos que compõem o modelo, seja por sua definição (por exemplo, regras de negócio, requisitos de negócio, *cluster*, etc.) ou por características padronizadas no fluxo de atividades do processo (por exemplo, ciclo de atividades, sequência de atividades, etc.).

Serviço candidato é uma abstração (não implementada) de serviço em tempo de projeto, que pode vir a ser implementada como um serviço físico (por exemplo, como *Web Service*) ou como uma funcionalidade de uma aplicação [ERL, 2005].

Os serviços candidatos são classificados da seguinte forma:

- Serviço candidato de dados: serviço que apenas executa operações de CRUD (*Create, Retrieve, Update e Delete* – Criar, Recuperar, Atualizar e Apagar);
- Serviço candidato de lógica: serviço que encapsula uma regra de negócio (lógica do negócio), não excluindo a possibilidade de encapsular alguma operação CRUD;
- Serviço candidato utilitário: serviço que implementa um padrão baseado em funções recorrentes em processos de negócio.

O método de identificação de serviços candidatos é subdividido em 3 etapas: (i) seleção de atividades; (ii) identificação e classificação de serviços candidatos; e, (iii) consolidação de serviços candidatos (Figura 12). O insumo do processo de identificação é um modelo de processos de negócio que se encontra na versão “*to-be*”.

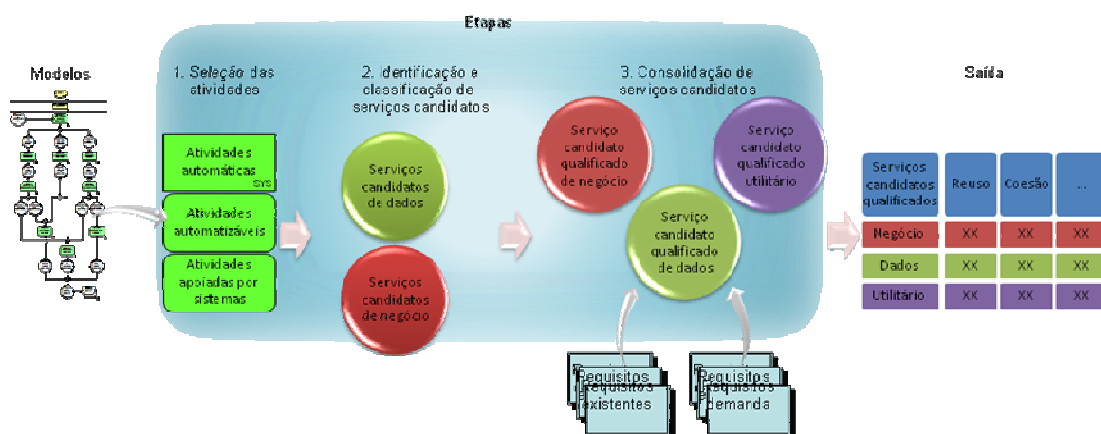


Figura 12 – Etapas do método de identificação de serviços candidatos

Na primeira etapa do método são selecionadas atividades automatizadas (executadas por sistema), apoiadas por sistema (executadas por pessoas, mas utilizando um sistema) e automatizáveis (as que se pretende automatizar). Atividades manuais e que não estão sendo consideradas para automatização são descartadas, já que não há necessidade de desenvolvimento de serviços para elas.

Na segunda etapa, os modelos de processos de negócio são analisados de acordo com um conjunto de heurísticas definidas por [AZEVEDO *et al.*, 2009a, 2009b]. As heurísticas analisam a estrutura e semântica dos modelos de processos:

- As heurísticas para análise da estrutura baseiam-se nos padrões de *workflow* [VAN DER AALST *et al.*, 2000, 2003] e [RUSSELL *et al.*, 2004] e cobrem todas as possibilidades de fluxos de atividades que podem ser representadas por um modelo de processo.
- As heurísticas para análise semântica consideram elementos dos modelos como os requisitos de negócio, informações de entrada e saída e regras de negócio. A semântica destes elementos indica funcionalidades que podem ser implementadas em serviços de apoio ao processo.

O resultado desta etapa é a identificação de serviços candidatos de dados (executam operações CRUD sobre os dados) e/ou serviços candidatos de lógica (executam regras de negócio e eventualmente podem incluir operações CRUD).

Na terceira etapa, os serviços candidatos identificados na etapa anterior são analisados e são geradas informações referentes à granularidade, grau de reuso, dependência entre serviços, associações dos serviços com as atividades de onde foram identificados, papéis que os executam e sistemas que poderão invocá-los etc.

As informações geradas na etapa de consolidação são apresentadas em tabelas, gráficos, grafos de dependência etc. para tornar possível a manipulação dos resultados pela equipe de arquitetos SOA. Estas informações são utilizadas nas fases de análise e projeto de serviços, a fim de definir a melhor forma de implementação do conjunto de serviços candidatos.

### **4.3 Heurísticas de identificação de serviços candidatos**

Nesta seção é enumerada e descrita em detalhes cada heurística utilizada na identificação dos serviços candidatos, propostas por [AZEVEDO *et al.*, 2009a, 2009b, 2009c, 2009d].

#### **4.3.1 Heurística de identificação de serviços a partir de regras de negócio**

Regras de negócio correspondem a declarações de políticas ou condições que devem ser satisfeitas pelos processos da organização (OMG, 2008). As regras de negócio definem ou restringem alguns aspectos do negócio. Sua intenção é garantir a estrutura do negócio ou influenciar o seu comportamento (GUIDE BUSINESS RULE PROJECT, 2008).

A automatização de uma atividade ligada a uma regra de negócio se refere à implementação destas regras em aplicações ou em bancos de dados. Em geral, regras de restrição ou de derivação são implementadas em aplicações, enquanto que regras de definição de termo de negócio e de relacionamento entre termos, as quais correspondem a asserções estruturais, são implementadas em bancos de dados. Entretanto, também existem casos em que regras podem ser implementadas tanto em bancos de dados quanto em aplicações, ou seja, a decisão de onde a regra será implementada é definida pelo projetista. Dessa forma, é importante documentar as

regras de negócio que já estão automatizadas ou para as quais exista demanda de automatização.

Serviços em uma arquitetura orientada a serviços estão diretamente associados à implementação de regras de negócio, segundo o qual é proposta a heurística:

**Heurística de regra de negócio: Um serviço candidato deve ser identificado a partir de uma regra de negócio.**

Considere o exemplo da Figura 13, onde as regras de negócio “Cadastro de cliente desatualizado” e “Cliente novo” dão origem a dois serviços candidatos, seja porque estão automatizados ou existem demandas por sua automatização no sistema “Crédito Direto”.

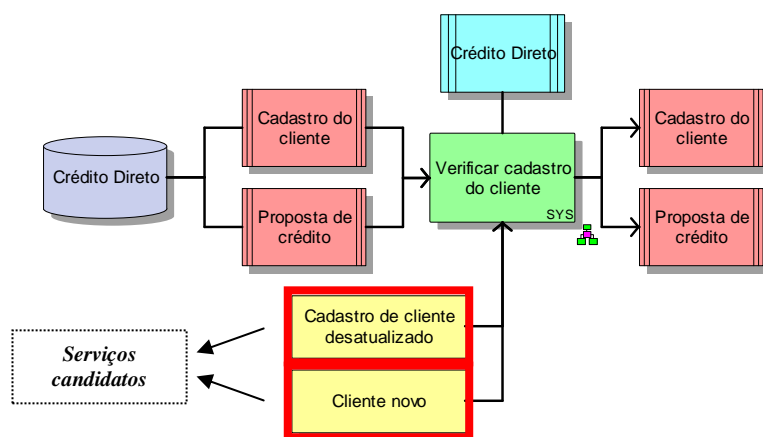


Figura 13 – Exemplo de regra de negócio.

Regra de Negócio	Descrição
Cadastro cliente desatualizado	O cadastro do cliente está desatualizado se o telefone, endereço e renda informados na proposta de crédito forem diferentes das informações do cliente existentes na base de dados. A comparação entre proposta de crédito e cadastro do cliente deve ser feita pelo CPF.
Cliente novo	Um cliente é novo se não existir cadastro de cliente com mesmo CPF que o informado na proposta de crédito.

#### 4.3.2 Heurística de identificação de serviços a partir de requisitos de negócio

Um requisito de negócio especifica uma funcionalidade que está disponibilizada em uma aplicação ou para a qual existe demanda para

implementação, além disso, em geral, um serviço pode implementar um ou mais requisitos de negócio. Dessa forma, é apresentada a seguinte heurística de identificação de serviços a partir de requisitos de negócio:

**Heurística de requisitos de negócio:** Um serviço candidato deve ser identificado a partir de um requisito de negócio.

Considere o exemplo da Figura 14, onde os requisitos de negócio “Aprovar créditos concedidos” e “Consultar créditos concedidos” dão origem a dois serviços candidatos, seja porque estão automatizados ou existem demandas por sua automatização no sistema “Crédito Direto”.

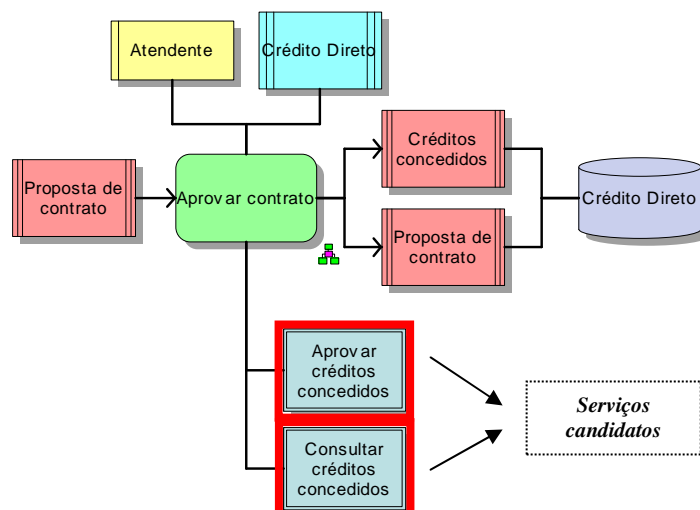


Figura 14 – Requisitos de negócio “Aprovar créditos concedidos” e “Consultar créditos concedidos”.

Requisito de Negócio	Descrição
Aprovar créditos concedidos	O Atendente deve aprovar os créditos concedidos para atendimento da proposta do cliente.
Consultar créditos concedidos	O Atendente deve consultar as seguintes informações de cada concessão de crédito feita para o cliente anteriormente: <ul style="list-style-type: none"> <li>- CPF do cliente,</li> <li>- valor total concedido,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- situação de crédito,</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> </ul> </li> </ul>

	<ul style="list-style-type: none"> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso,</li> <li>- valor pago,</li> <li>- data em que o pagamento foi realizado.</li> </ul>
--	---

### 4.3.3 Heurística de identificação de serviços a partir de *cluster*

Na elaboração dos modelos de detalhamento de atividades, são descritas as informações de entrada e de saída através do objeto denominado *cluster*. *Clusters* de entrada representam informações que a atividade necessita receber para que possa ser executada. De forma análoga, *clusters* de saída representam informações que a atividade produz ao final de sua execução. De forma geral, serviços de dados podem ser identificados a partir de *clusters* de entrada ou saída a fim de representar a leitura das informações necessárias para executar a atividade e a escrita das informações geradas pela atividade.

Em alguns casos, contudo, *clusters* podem representar informações que uma atividade receberá de outra atividade, sem persistência de dados entre o produtor da informação e a atividade recebedora da informação. Portanto, não necessariamente existirá um serviço de dados para ler ou escrever *clusters* de entrada ou saída de atividades. Baseado nisso, o objeto *cluster*, para ser identificado como serviço, deve estar ligado a um portador de informação que representa, por exemplo, uma base de dados que pode ser fonte de leitura ou destino de escrita do *cluster*.

Os serviços identificados a partir de *clusters* de entrada são classificados como candidatos a serviços de dados de leitura, ou seja, serviços que somente possuem a funcionalidade de extrair informações de uma fonte de dados. Serviços candidatos também serão identificados a partir de *clusters* de saída, os quais são classificados como candidatos a serviços de dados de escrita, ou seja, serviços que possuem somente a funcionalidade de armazenar informações em uma fonte de dados. Dessa forma, é apresentada a seguinte heurística de identificação de serviços a partir de *clusters*:

**Heurística de *cluster*:** Um serviço candidato de dado deve ser identificado para cada *cluster* de entrada de uma atividade (o qual também deve ser marcado como serviço de

leitura), assim como serviço candidato de dado deve ser identificado para cada *cluster* de saída de uma atividade (o qual deve ser marcado como de escrita), desde que os *clusters* estejam associados a portadores de informação.

Considere o exemplo da Figura 15, onde existem quatro *clusters* de entrada e um de saída. A partir destes *clusters* serão realizadas as seguintes análises:

- *Cluster* de entrada “Créditos concedidos”: Como o *cluster* não está ligado a nenhum portador de informação, nenhum serviço candidato será identificado.
- *Cluster* de entrada “Taxa de juros”: Como o *cluster* está ligado ao portador de informação (neste caso, “Crédito Direto”), será identificado o serviço candidato “Recuperar taxa de juros”.
- *Cluster* de entrada “Proposta de crédito”: Como o *cluster* não está ligado a nenhum portador de informação, nenhum serviço candidato será identificado.
- *Cluster* de entrada “Alíquota do imposto”: Como o *cluster* não está ligado a nenhum portador de informação, nenhum serviço candidato será identificado.
- *Cluster* de saída “Taxa de juros do cliente”: Como o *cluster* está ligado ao portador de informação (neste caso, “Crédito Direto”), será identificado o serviço candidato “Armazenar taxa de juros do cliente”.

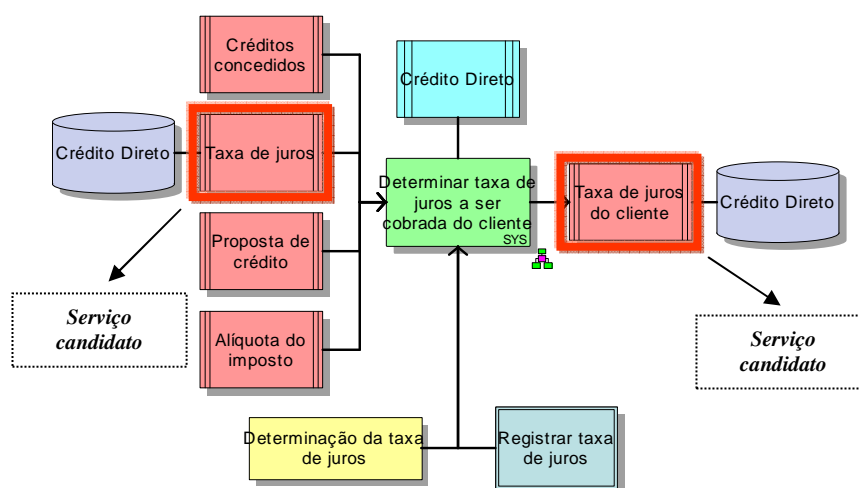


Figura 15 – Exemplo de *clusters* conectados a portadores de informação



#### **4.3.4 Heurística de identificação de serviços a partir de padrões de *workflow***

Um padrão de *workflow* é definido como a abstração de uma forma concreta que se mantém repetitiva em contextos específicos. Estes padrões identificam funcionalidades de *workflow* e oferecem o embasamento necessário para a comparação de sistemas de gerenciamento de *workflow* e avaliação da adequação de linguagens de *workflow*. Estes padrões não descrevem requisitos de negócio, eles são um resumo dos recursos disponíveis em sistemas existentes [AZEVEDO *et al.*, 2009c].

O objetivo desta seção é apresentar as heurísticas de identificação de serviços a partir de padrões de *workflow*. Neste caso, os serviços candidatos identificados serão responsáveis por encapsular regras de negócio que são utilizadas para determinar a dependência entre atividades, explicitando o fluxo dos processos de negócio.

Os serviços identificados a partir destes padrões serão, em geral, irão auxiliar no controle dos fluxos, controle de execução de atividades ou na tomada de decisões.

##### **4.3.4.1 Heurística de identificação de serviços a partir de operadores lógicos *AND***

O fluxo demarcado por um operador lógico *AND*, representa a execução paralela das atividades que são realizadas em todas as ramificações que partem do operador e finaliza em eventos finais ou em outro operador lógico *AND*, responsável pela sincronização das ramificações para a continuidade do processo. O serviço candidato identificado a partir de um *AND* pode ser considerado como um serviço que orquestra a execução paralela das atividades. Além disso, este serviço é responsável pela consolidação das respostas produzidas por cada ramificação e o seu repasse, que permite a continuação do processo. Dessa forma, é apresentada a seguinte heurística de identificação de serviços a partir de operador lógico *AND*:

**Heurística de *AND*:** Um serviço candidato deve ser identificado a partir de uma estrutura iniciada em um ponto no *workflow* onde um fluxo de controle simples

divide-se em fluxos de controle múltiplos, que podem ser executados em paralelo, e finalizados em um ponto no *workflow* onde os múltiplos fluxos paralelos convergem em um fluxo de controle simples, sincronizando-os, ou onde ramificações terminem em evento final.

Na Figura 16, um serviço candidato deve ser identificado a partir de toda a estrutura do *AND*, destacada pelo retângulo vermelho. O serviço candidato será responsável por controlar a execução das atividades.

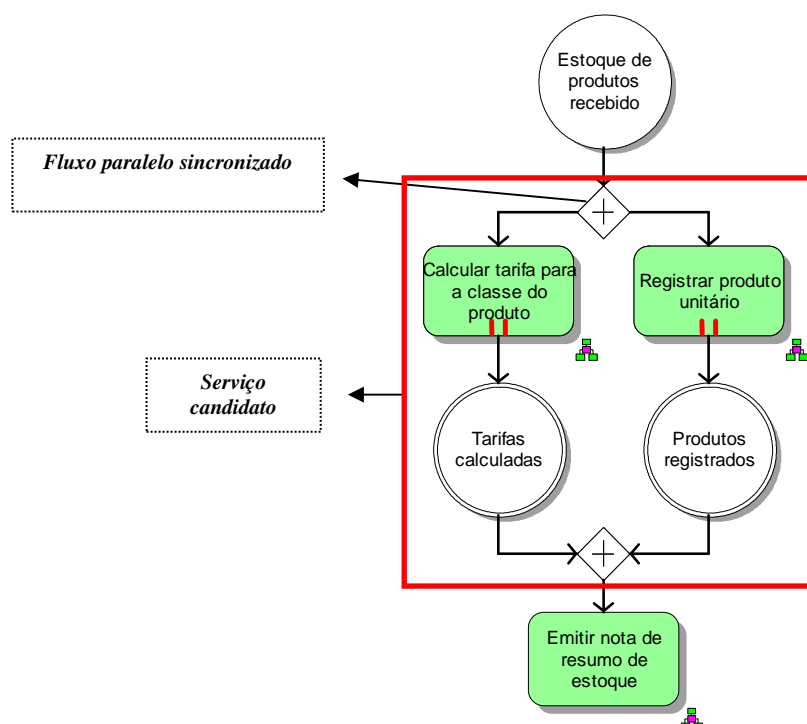


Figura 16 – Exemplo de padrão de *workflow AND* onde os múltiplos fluxos paralelos convergem em um fluxo de controle simples

A Figura 17 apresenta um exemplo de *AND* considerando atividades automatizadas. Segundo a “Heurística de *AND*”, deve ser identificado um serviço candidato responsável pela execução das quatro atividades.

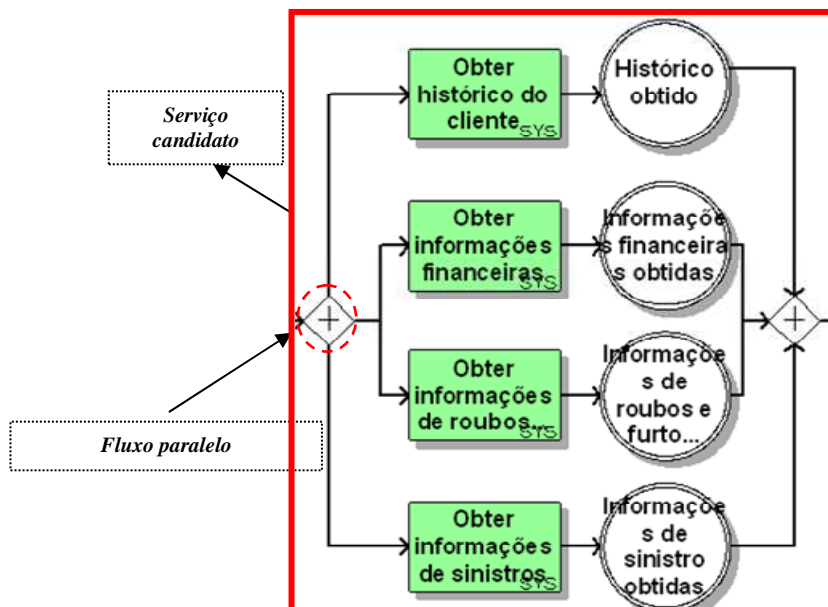


Figura 17 – Exemplo de padrão de *workflow AND* com atividades automatizadas

Na Figura 18, um serviço candidato deve ser identificado a partir da estrutura do AND. Neste exemplo, um dos fluxos converge para um evento final e outro fluxo converge para outro processo.

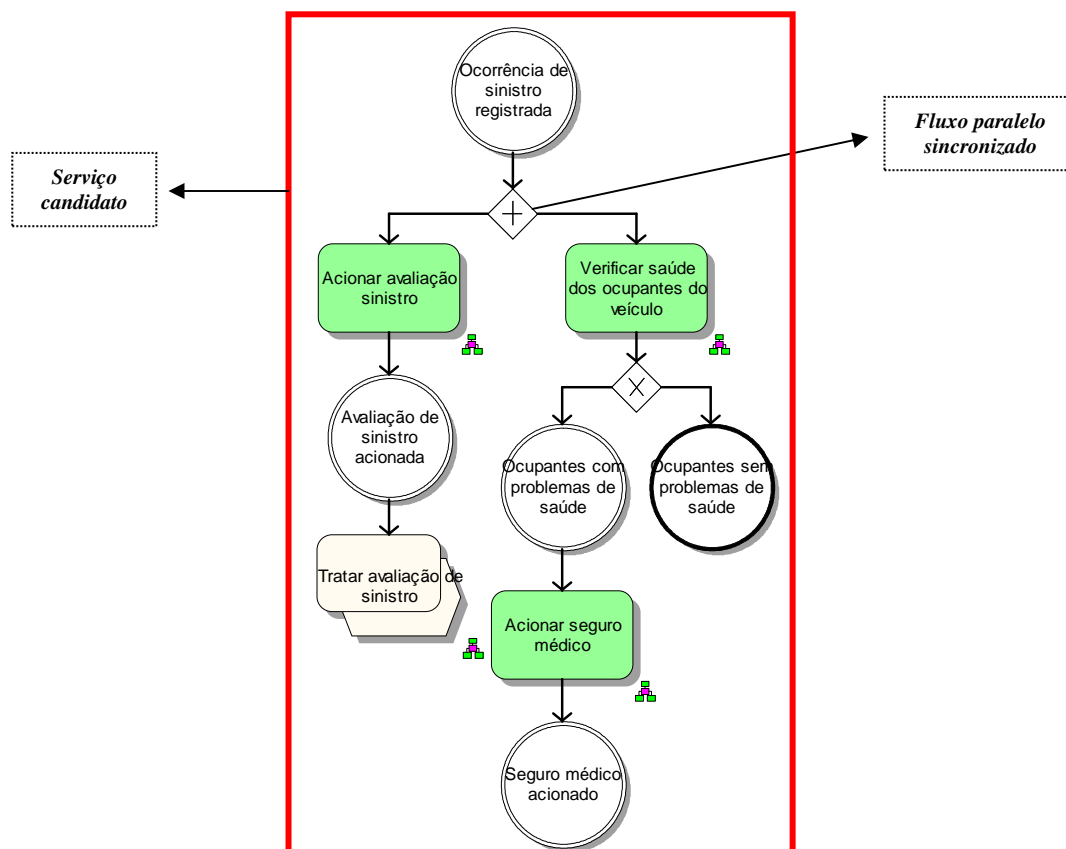


Figura 18 - Exemplo de padrão de *workflow AND* onde um dos fluxos converge para um evento final e o outro fluxo converge para outro processo

#### **4.3.4.2 Heurística de identificação de serviços a partir de operadores lógicos *OR***

O fluxo demarcado por operadores lógicos *OR* representa a execução de pelo menos uma ramificação que parte do operador, permitindo executar qualquer número entre as outras ramificações em paralelo, de forma assíncrona, não criando dependência entre a execução das ramificações para a continuação do processo, no momento em que as ramificações convergem para o operador lógico *OR*. Um serviço pode ser desenvolvido para a implementação desta regra e correta chamada de execução das atividades nas ramificações que, por exemplo, podem ser orquestradas por outro serviço. Dessa forma, é apresentada a seguinte heurística de identificação de serviços a partir de operador lógico *OR*.

**Heurística de *OR*:** Um serviço candidato deve ser identificado a partir de uma estrutura iniciada em um ponto no *workflow* onde, baseado em uma decisão, uma ou mais ramificações do fluxo é escolhida e finalizada em um ponto no *workflow* onde as várias ramificações do fluxo se juntem. Se mais de um dos fluxos de origem for executado, então é necessário sincronizá-los. Ramificações podem também terminar em um evento final.

A Figura 19 mostra as atividades “Verificar dados de contato do candidato”, “Emitir carta de aviso de aprovação no concurso”, “Registrar comprovante de envio da carta de aviso” e “Enviar e-mail de aviso de aprovação no concurso”. Estas quatro atividades serão agrupadas em um único serviço candidato que será responsável pelo controle de execução das atividades.

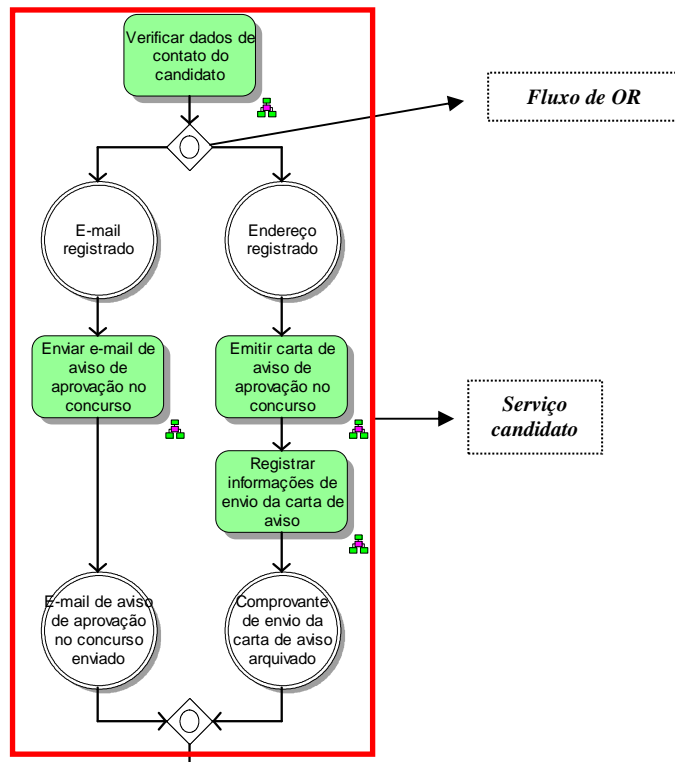


Figura 19 – Exemplo de padrão de *workflow OR* onde os múltiplos fluxos convergem em um fluxo de controle simples

Na Figura 20, todas as atividades devem ser agrupadas em um único serviço candidato. Neste exemplo, fluxos convergem para eventos finais, os quais deverão ser considerados pelo serviço.

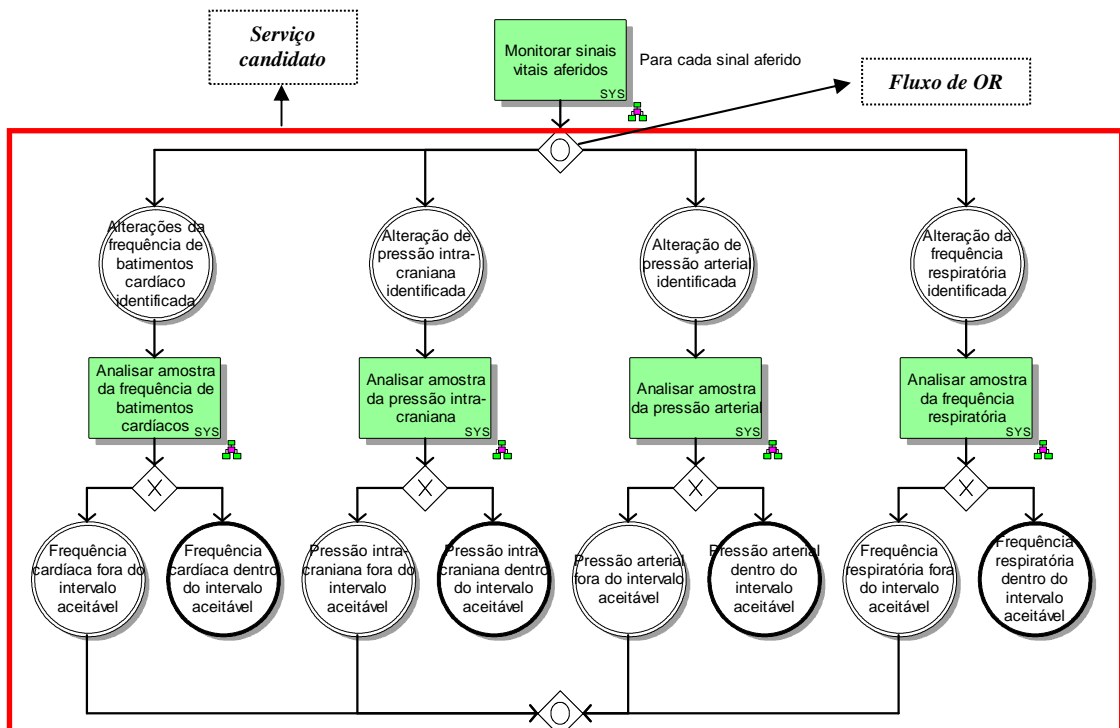


Figura 20 - Exemplo de padrão de *workflow OR* onde fluxos convergem para eventos finais

#### **4.3.4.3 Heurística de identificação de serviços a partir de operadores lógicos XOR**

O fluxo demarcado por um operador lógico XOR, representa uma regra que processa uma decisão sobre a execução de apenas uma das ramificações do fluxo. Um serviço pode ser desenvolvido para executar esta regra e realizar a chamada das atividades na ramificação correta que, por exemplo, podem ser orquestradas por outro serviço. Dessa forma, é apresentada a seguinte heurística de identificação de serviços a partir de operador lógico *XOR*:

**Heurística de XOR:** Um serviço candidato deve ser identificado a partir de uma estrutura iniciada em um ponto no *workflow* onde, baseado em uma decisão, uma e somente uma de várias ramificações do fluxo é escolhida, e finalizada em um ponto no *workflow* onde as ramificações do fluxo se juntem sem sincronização ou quando uma ou mais das ramificações termine em evento final.

A decisão que define qual ramificação do fluxo deverá ser seguida pode ser encapsulada no serviço. No exemplo da Figura 21 as atividades “Verificar cadastro do cliente”, “Atualizar cadastro do cliente” e “Cadastrar cliente” dão origem a um serviço candidato onde será embutida a decisão e a execução representadas pela estrutura. Como o fluxo é composto por atividades automatizadas, então será identificado um serviço candidato para controlar a execução das atividades.

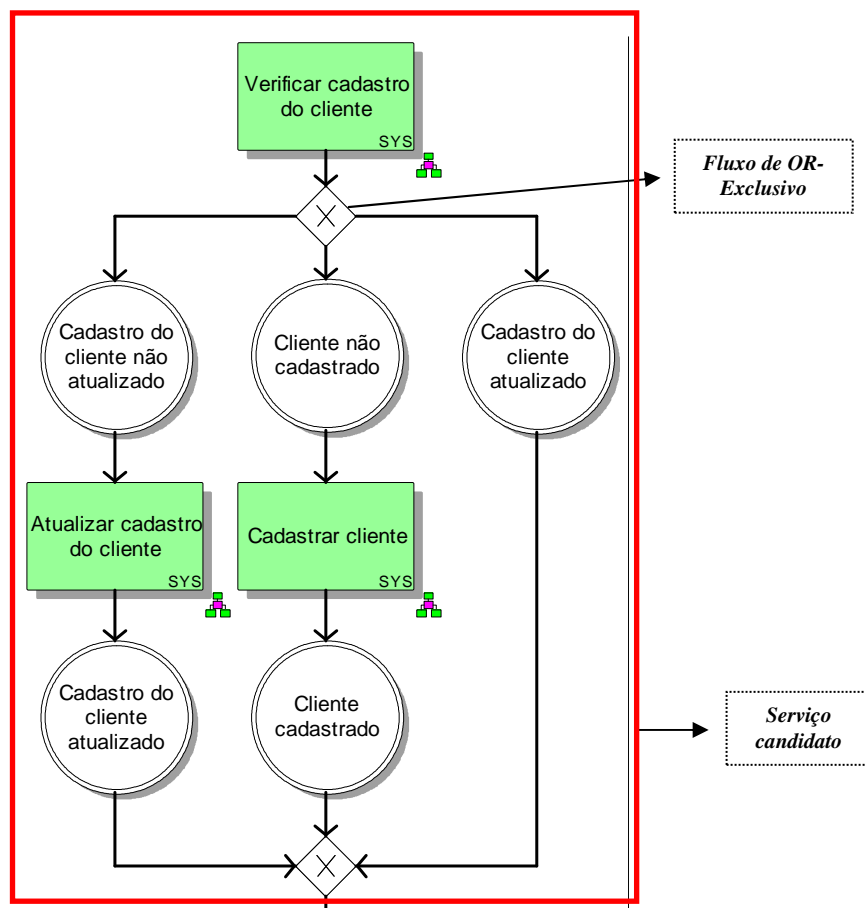


Figura 21 – Exemplo de padrão de *workflow XOR* onde os múltiplos fluxos convergem em um fluxo de controle simples

Na Figura 22, é apresentado um exemplo onde uma das ramificações após o *XOR* segue para um evento final.

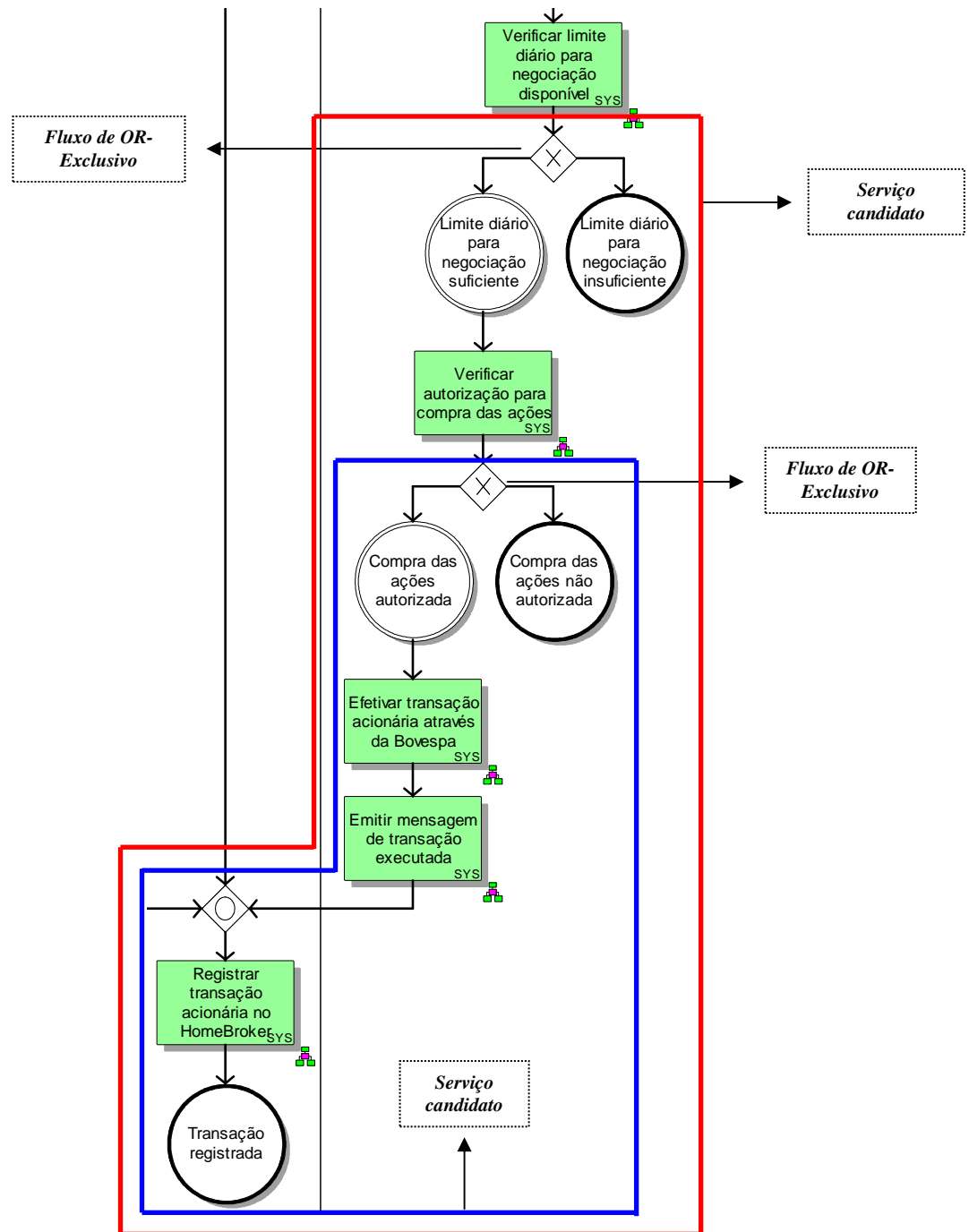


Figura 22 - Exemplo de padrão de *workflow XOR* onde um dos fluxos converge para um evento final

Quando a regra de negócio que expressa a decisão do *XOR* já está definida na atividade anterior ao operador e não existem outras atividades após o *XOR*, então não deve ser identificado serviço a partir do *XOR*.



#### 4.3.4.4 Heurística de identificação de serviços a partir de atividades sequenciais

O fluxo que define um serviço de atividades sequenciais é formado por no mínimo duas atividades que são executadas seguidamente, sem a intervenção de operadores lógicos (*AND*, *OR*, *XOR*). Não há limite para o número de atividades presente na sequência, entretanto, todas as atividades participantes devem ser apoiadas ou executadas por sistema. Dessa forma, é apresentada a seguinte heurística de identificação de serviços a partir de atividades sequenciais:

**Heurística de atividades sequenciais:** Um serviço candidato deve ser identificado para um conjunto de atividades sequenciais.

Considere o exemplo da Figura 23, onde a sequência é composta por cinco atividades, sendo quatro automatizadas e uma apoiada por sistema. O fluxo contínuo formado pelas cinco atividades é demarcado por dois operadores lógicos que iniciam a execução deste pedaço do processo que caracteriza a sequência de atividade. Segundo a “Heurística de atividades sequenciais”, deve ser identificado um serviço candidato responsável pela execução de todas estas as atividades.

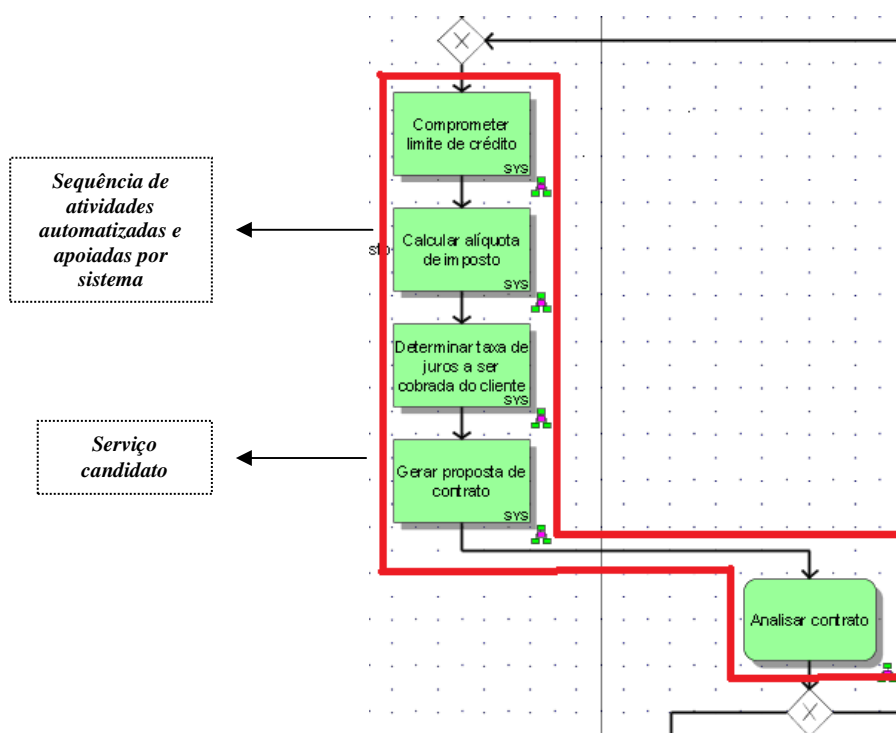


Figura 23 - Exemplo de sequência de atividades

#### 4.3.4.5 Heurística de identificação de serviços a partir de ciclo de atividades

Um ciclo ou *loop* em um modelo de processos é caracterizado pela repetição de uma ou mais atividades até que seja satisfeita uma condição que permitirá a continuação do processo ou o fim de sua instância. Esse processo repetitivo pode definir um serviço com alto grau de reuso, seja através de sua execução em processos distintos e, principalmente, pelo grau de repetição que pode existir devido ao ciclo das atividades. Dessa forma, é apresentada a seguinte heurística de identificação de serviços a partir de ciclo de atividades:

**Heurística de loop:** Um serviço candidato deve ser identificado a partir de uma estrutura do *workflow* onde uma ou mais atividades podem ser executadas repetidamente.

Considere o exemplo da Figura 24, onde o ciclo é composto por quatro atividades automatizadas e duas apoiadas por sistema. A primeira atividade (“Comprometer limite de crédito”) inicia imediatamente após o operador lógico, e o fluxo continua até a atividade “Analisar contrato”, onde é possível seguir por três caminhos dependendo do seu resultado.

Caso a necessidade de ajuste tenha sido identificada, a atividade “Alterar proposta de crédito” será executada e levará novamente a execução da primeira atividade, resultando no retorno do processo para a re-execução das atividades, o que caracteriza o ciclo. Segundo a “Heurística de *loop*”, deve ser identificado um serviço candidato responsável pela execução de todas estas atividades.

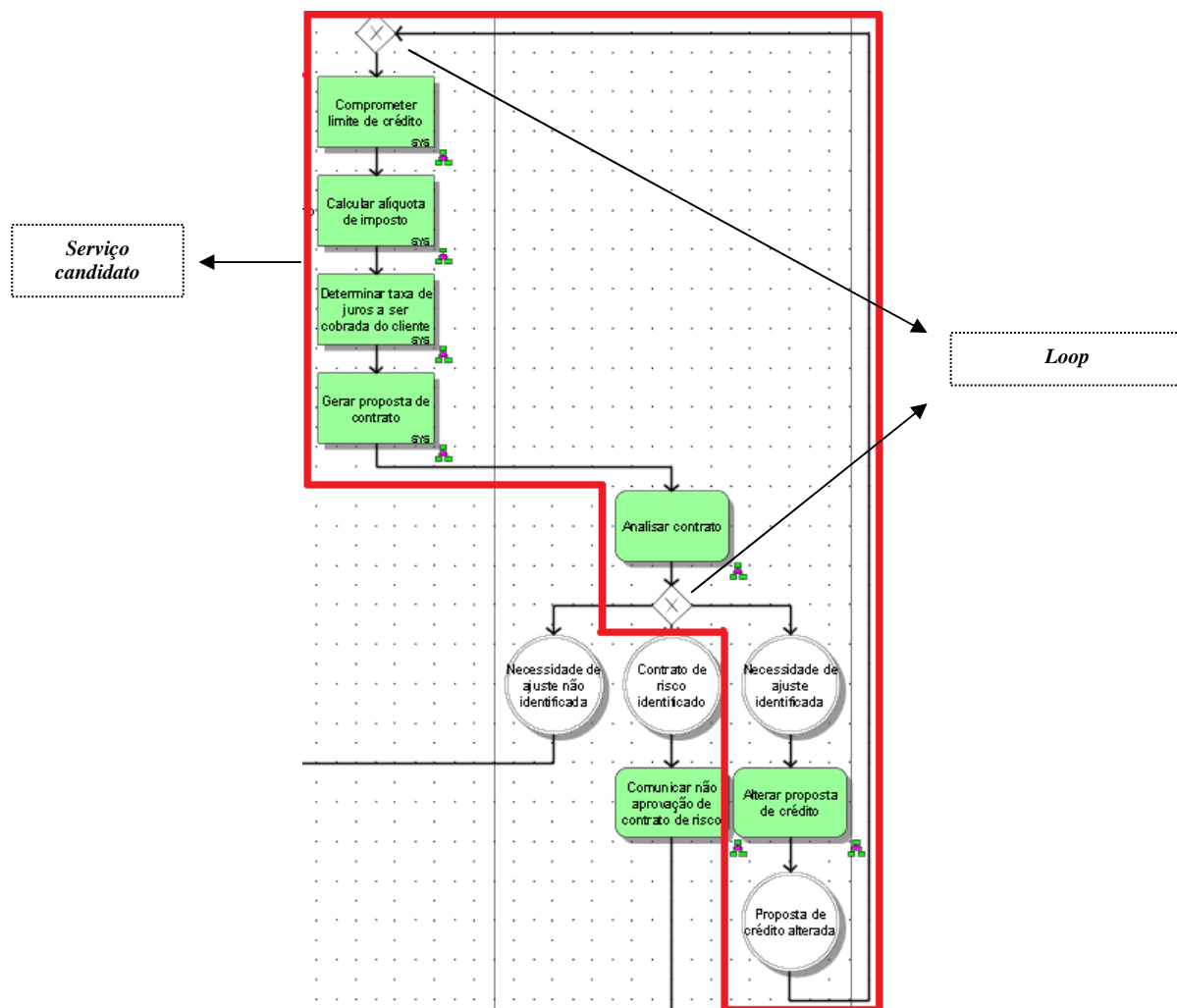


Figura 24 – Exemplo de ciclo/loop de atividades

Este exemplo demonstra quatro atividades automatizadas em uma primeira raia e outras duas que são apoiadas por sistema em uma segunda raia, caracterizando um serviço candidato, que é executado por dois papéis distintos. Ainda é possível observar que os ciclos são demarcados por operadores lógicos que definem o seu início e seu ponto de retorno.

#### 4.3.4.6 Heurística de identificação de serviços a partir de interfaces de processo

Um processo de negócio é definido pela coordenação de macroprocessos que, por sua vez, são compostos por outros macroprocessos ou atividades que visam alcançar os objetivos do negócio. Quando o fim da execução de um macroprocesso

demarca o início de outro, é necessária uma interface que crie essa ligação e permita a comunicação entre estes macroprocessos.

Após a execução da última atividade do processo predecessor A, ele deverá emitir uma mensagem ao seu sucessor B para que ele possa ser inicializado. Neste momento identificam-se duas atividades implícitas de envio e recebimento de mensagens entre os processos, que coordenam a correta sequência de suas execuções através de uma interface de processo.

Se a última atividade do processo A e/ou a primeira atividade do processo B forem automatizadas, considera-se que a passagem da mensagem de A para B possa ser realizada por duas funções de sistema: um para envio e outro para recepção da mensagem. Dessa forma, é apresentada a seguinte heurística de identificação de serviços a partir de interfaces de processo:

**Heurística de interface de processo:** Um serviço candidato deverá ser identificado a partir de uma interface que inicia um processo desde que em pelo menos um dos caminhos a partir da interface a primeira atividade seja automatizada. Este serviço é responsável por receber a mensagem do outro processo e iniciar a atividade automatizada. Da mesma forma, um serviço candidato deverá ser identificado a partir de uma interface que finaliza um processo desde que em pelo menos um dos caminhos que chegam na interface a atividade exatamente anterior seja automatizada. Este serviço é responsável por enviar a mensagem para o outro processo.

Considere o exemplo da Figura 25, que ilustra a comunicação entre os processos “Receber proposta de crédito” (Figura 25.a) e “Analisar pedido de crédito” (Figura 25.b). Neste exemplo, o processo “Receber proposta de crédito” deve enviar uma mensagem para o processo “Analisar pedido de crédito” para que ele possa ser inicializado (Figura 25.a). Os dados registrados da proposta de crédito são enviados nesta mensagem. Como a atividade “Verificar cadastro do cliente” é automatizada

(Figura 25.b), então um serviço candidato responsável por receber a mensagem do processo “Receber proposta de crédito” deve ser identificado.

Analogamente, a interface “Analisar pedido de crédito”, existente no processo “Receber proposta de crédito” (Figura 25.a), dá origem ao serviço candidato responsável por enviar as informações do processo “Receber proposta de crédito” para o processo “Analisar pedido de crédito”, uma vez que existe pelo menos uma atividade automatizada, que neste exemplo é “Registrar proposta de crédito”.

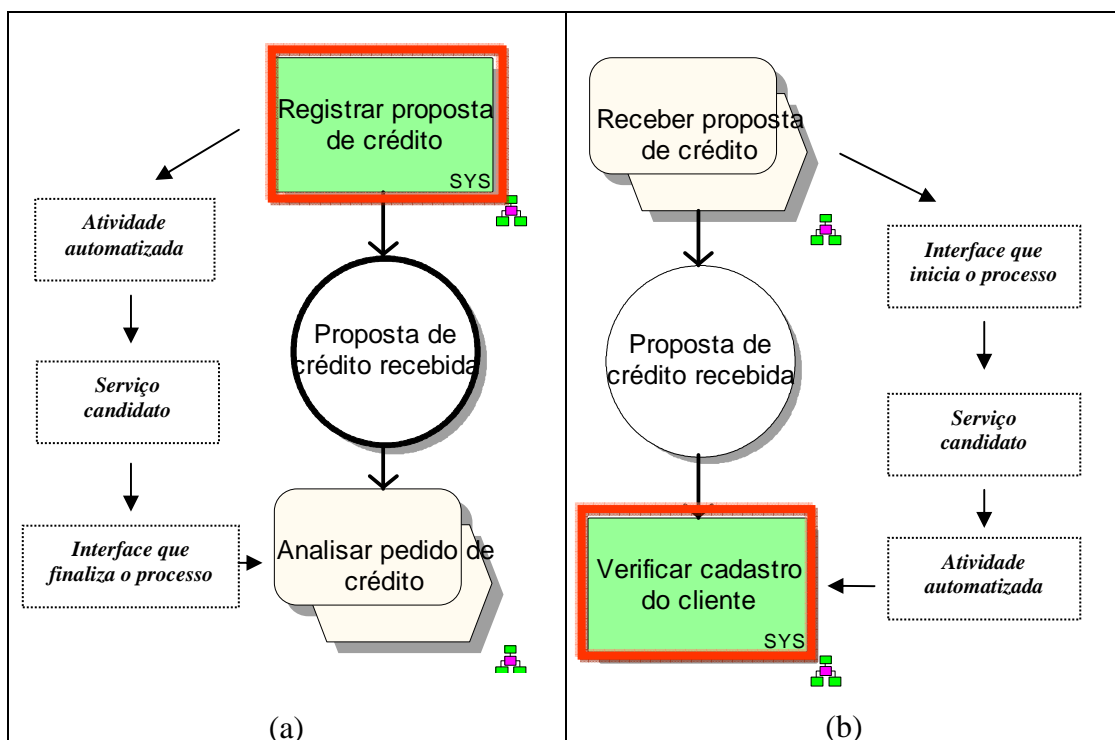


Figura 25 – Comunicação entre processos utilizando interfaces de processo

#### 4.3.4.7 Heurística de identificação de serviços a partir de atividades de múltiplas instâncias

A atividade de múltipla instância representa uma atividade que é executada repetidamente para cada variante específica no seu contexto, a qual requisita uma nova instância da atividade para ser realizada. Por exemplo, na Figura 26, a atividade “Calcular alíquota de imposto” é instanciada para cada tipo de imposto cobrado, onde são calculadas suas alíquotas.

No caso de uma atividade de múltiplas instâncias há um ponto imediatamente anterior responsável por enviar múltiplas mensagens (serviço remetente) para as

instâncias da atividade de múltiplas instâncias e outro ponto posterior responsável por consolidar as respostas de cada instância (serviço destinatário).

O serviço remetente é um serviço candidato (também conhecido como *broker*) e é responsável por receber a mensagem em questão e distribuí-la para todos os serviços destinatários. Cada instância de execução de uma tarefa de instância múltipla executa independentemente das outras instâncias e, dessa forma, tem um requisito de passar os elementos de dados, após sua conclusão, para tarefas subsequentes. Dessa forma, existe um serviço candidato que deve ser identificado para realizar esta consolidação. Este serviço está de acordo com o padrão 12, de interação de dados “De tarefa de instância múltipla”, proposto em (RUSSEL *et al.*, 2004). Além disso, deve ser identificado um serviço candidato para executar as múltiplas instâncias.

Dessa forma, é apresentada a seguinte heurística de identificação de serviços a partir de atividades de múltiplas instâncias:

**Heurística de atividades de múltiplas instâncias:** Os seguintes serviços candidatos devem ser identificados a partir de um ponto do *workflow* onde múltiplas mensagens são enviadas a partir de uma atividade para outras atividades conhecidas:

- Um serviço candidato de dado deve ser identificado para o remetente da mensagem. Este serviço será responsável por transmitir as informações para os serviços destinatários.
- Um serviço candidato deve ser identificado para cada destinatário.
- Um serviço candidato de dado deve ser identificado para consolidar as execuções das diferentes instâncias da atividade de múltiplas instâncias e transferi-la para o destinatário.

Na Figura 26, a atividade “Calcular alíquota de imposto” deve ser executada para cada tipo de imposto. Um serviço candidato deve ser identificado para a atividade “Calcular alíquota de imposto”. No entanto, é necessário que as informações dos impostos sejam enviadas para a atividade de múltiplas instâncias. Logo, um serviço candidato responsável por enviar as mensagens para a atividade de múltiplas instâncias deve ser identificado, por exemplo, com o nome “Enviar mensagens para calcular alíquota de imposto”. Assim como um serviço candidato deve ser identificado para consolidar as informações provenientes da atividade “Calcular alíquota de imposto”, por exemplo, o serviço candidato “Calcular alíquota de imposto”.

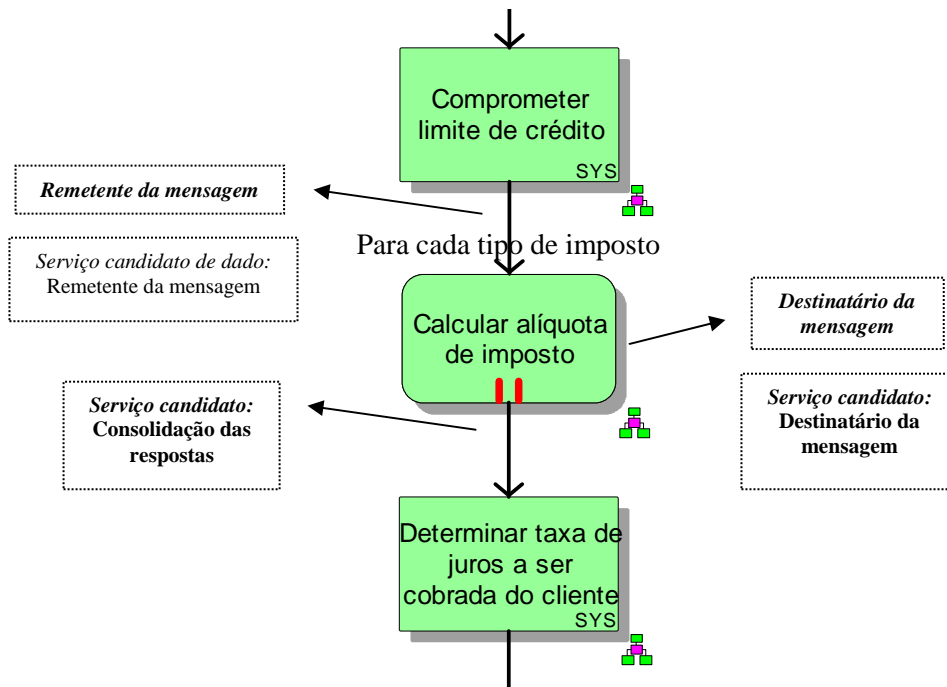


Figura 26 – Atividade de múltiplas instâncias identificam três serviços candidatos

#### 4.4 Heurísticas de consolidação de serviços candidatos

Após a identificação dos serviços candidatos, são realizadas novas análises nos objetos identificados as quais compõem a etapa que realiza a consolidação dos serviços candidatos seguindo heurísticas propostas em [AZEVEDO *et al.*, 2009a, 2009b]. Estas informações têm o objetivo de auxiliar a equipe de arquitetos SOA a definir a melhor forma de implementação do conjunto de serviços físicos.

Nesta seção, são detalhadas as heurísticas utilizadas na consolidação dos serviços candidatos.

#### **4.4.1 Heurística de eliminação de serviços candidatos**

A heurística de eliminação de serviços candidatos consiste em garantir a ausência de serviços iguais representados por objetos diferentes nos modelos e de regras que representam apenas valores padrão. Ou seja, caso existam dois ou mais serviços com nomes diferentes descrevendo exatamente a mesma funcionalidade, apenas um serviço deve ser mantido e regras que explicitam valores padrão (*default*) que devem ser considerados na execução de serviços, não devem ser indicadas como serviços candidatos. O enunciado da heurística segue abaixo:

**Heurística de eliminação de serviços candidatos:** Devem ser eliminados todos os serviços candidatos que apareçam duplicados ou que refiram a regras que explicitam valores padrão.

Por exemplo, na Figura 27, a atividade possui duas regras de negócio: “Determinação da taxa de juros” e “Valor limite para a taxa de juros”. A regra de negócio “Valor limite para a taxa de juros” especifica o valor a ser utilizado quando o serviço for executado e não funcionalidades que um serviço deva executar para o negócio. Dessa forma, o serviço candidato obtido a partir desta regra de negócio deve ser descartado.



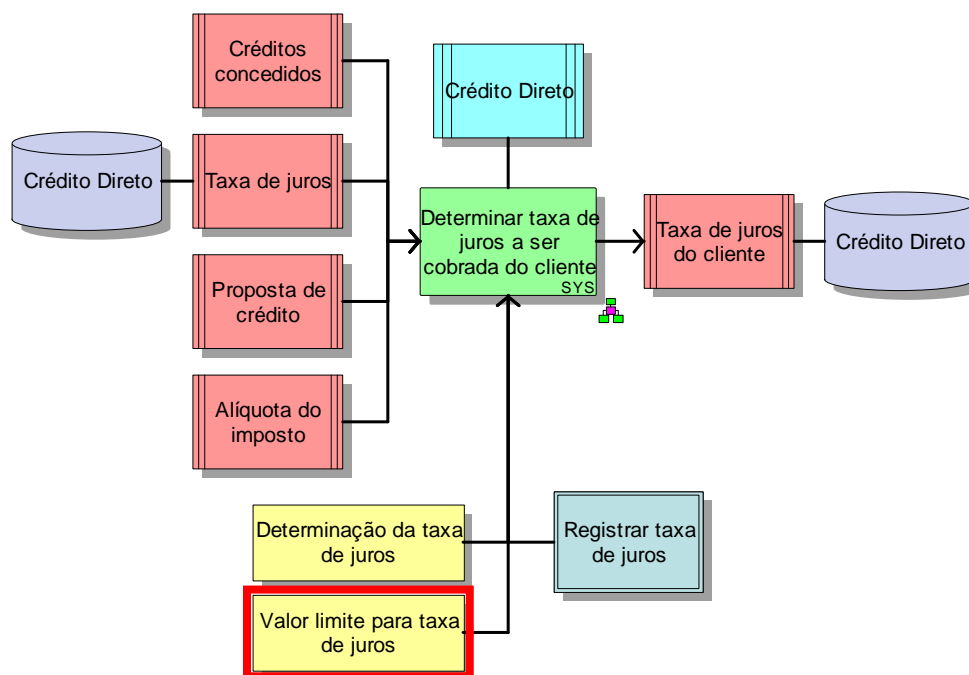


Figura 27 – Eliminação de serviços candidatos

Regra de Negócio	Descrição
Determinação da taxa de juros	A taxa de juros deve ser determinada de acordo com o valor total solicitado para financiamento e o número de parcelas.
Valor limite para taxa de juros	O valor limite para a taxa de juros é de 12%.

Figura 28 – FAD da atividade “Configurar parâmetros para verificação de variação da produção”

#### 4.4.2 Heurística de grau de reuso de serviços candidatos

O cálculo do grau de reuso consiste em verificar o número de ocorrências de um dado serviço nos processos de negócios da organização. Um serviço pode estar replicado em atividades, que podem estar replicadas em um mesmo modelo. O modelo pode estar replicado em subprocessos e os subprocessos replicados em macroprocessos, o que demonstra o potencial de reuso de certos serviços que são comuns aos processos da organização.

Por exemplo, considerando a Figura 29, a partir do requisito de negócio “Consultar informações do cliente” é identificado o serviço candidato “Consultar cadastro do cliente”. Este requisito de negócio aparece nas atividades “Gerar proposta de contrato”, “Verificar cadastro do cliente” e “Atualizar cadastro do cliente”, como apresentado nas Figura 29, Figura 30 e Figura 31. Como estas

atividades não se repetem nos modelos, então o serviço candidato “Consultar cadastro do cliente” tem grau de reuso igual a três, ou seja, o somatório do número de ocorrências do requisito de negócio nos modelos FAD.

Caso houvesse repetição das atividades onde se encontra o requisito de negócio, acarretaria em novas ocorrências do requisito de negócio, o que aumentaria o seu grau de reuso.

Apesar do reuso ser desejável, também existem os serviços candidatos de grau de reuso um, onde não ocorrem repetições do objeto ou conjunto de objetos que caracterizam o serviço candidato.

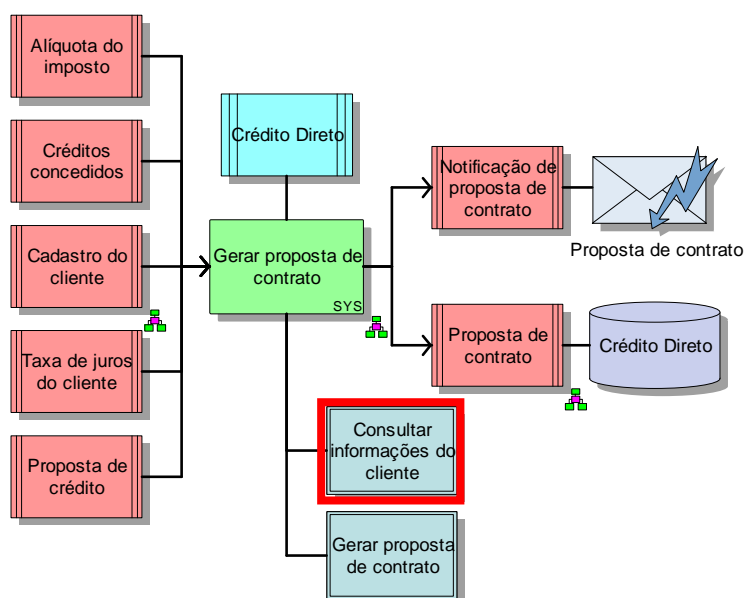


Figura 29 – FAD da atividade “Gerar proposta de contrato”

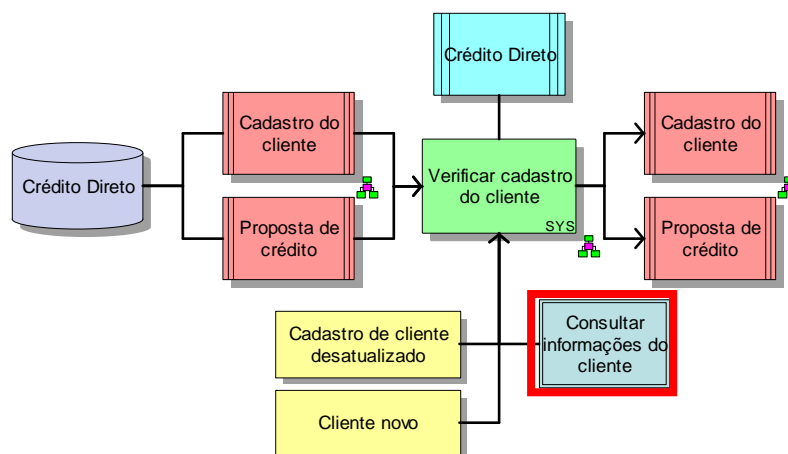


Figura 30 – FAD da atividade “Verificar cadastro do cliente”

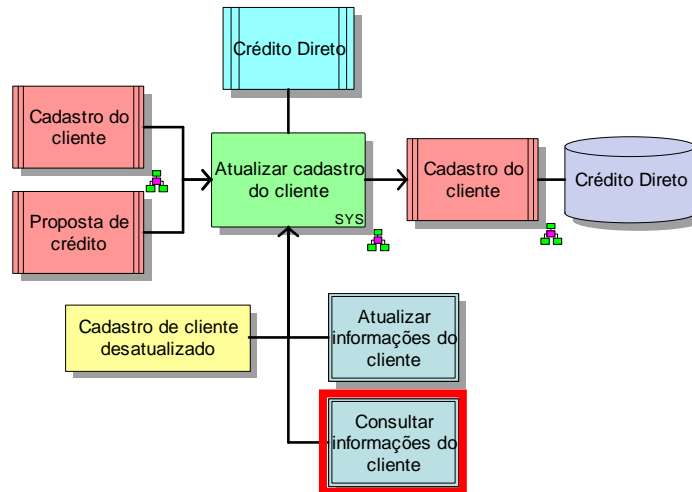


Figura 31 – FAD da atividade “Atualizar cadastro do cliente”

O cálculo do grau de reuso possui um algoritmo distinto para cada tipo de serviço, já que os objetos que compõem cada serviço são diferentes. Abaixo segue a definição da heurística e, em seguida, o detalhamento dos algoritmos que calculam o grau de reuso dos diferentes serviços:

#### Heurística de grau de reuso:

- O grau de reuso de um serviço candidato identificado a partir de uma atividade corresponde à soma do número de ocorrências de cada atividade em que o serviço está relacionado (de acordo com o elemento do qual o serviço foi identificado).
- O grau de reuso de um serviço candidato identificado a partir de um conjunto de atividades (por exemplo, serviço candidato identificado a partir de atividades sequenciais) é igual ao número de ocorrências do conjunto de atividades na mesma ordem que estas foram definidas no serviço.
- Além disso, quando um serviço candidato “A” compõe outro serviço candidato “B”, então se deve somar o grau de reuso do serviço composto “B” ao grau de reuso do serviço “A”.

#### **4.4.3 Heurística de grau de reuso de serviço candidato identificado a partir de atividade de múltiplas instâncias**

A modelagem de processos não identifica a multiplicidade das atividades de múltiplas instâncias com um limite superior predefinido. Desta forma, não é possível definir com precisão qual o grau de reuso de um serviço proveniente de uma atividade de múltiplas instâncias. Contudo, é necessário que essa informação esteja disponível para que o projetista de serviços saiba que essa atividade terá um grau de reuso maior que o indicado na heurística de grau de reuso de serviço candidato.

**Heurística de grau de reuso de serviço candidato identificado a partir de atividade de múltiplas instâncias:** O serviço identificado a partir de atividades de múltiplas instâncias deve conter uma marcação especial a fim de indicar que o seu reuso é maior do que o calculado pela heurística de grau de reuso de serviço candidato, pois ele também é reutilizado por cada instância da atividade.

#### **4.4.4 Heurística de associação de serviços candidatos com sistemas**

Uma atividade pode possuir vários requisitos de negócio em sua execução e, ao mesmo tempo, ser apoiada por mais de um sistema. Visualmente não é possível identificar em um FAD qual requisito de negócio está implementado no respectivo sistema quando a atividade possui mais de um sistema de apoio (Figura 32). Além disso, um requisito de negócio pode ser implementado por um ou mais sistemas.

Para identificar o(s) correto(s) sistema(s) que implementa(m) determinado requisito, é necessário verificar se existe um relacionamento entre o objeto requisito de negócio e o sistema dentro do “modelo de requisito de negócio do sistema” (Figura 33). Se houver, o serviço identificado a partir do requisito deverá ser relacionado aos nomes dos sistemas que o implementam. Isso permitirá ao projetista identificar quais requisitos já estão implementados e que poderiam ser expostos

como serviços mais facilmente e/ou priorizar a implementação de serviços de acordo com a prioridade de atendimento de demandas de determinado sistema.

**Heurística de associação de serviços candidatos com sistemas:** Um serviço candidato identificado a partir de um requisito de negócio deve ser associado aos sistemas que o implementam na tabela de consolidação.

Na Figura 32, o requisito de negócio “Cadastrar produto para cotação” é implementado pelo sistema “CotaWare”, então o serviço candidato identificado a partir deste requisito deve ser marcado. Esta marcação deve aparecer ao lado da informação do reuso dos serviços.

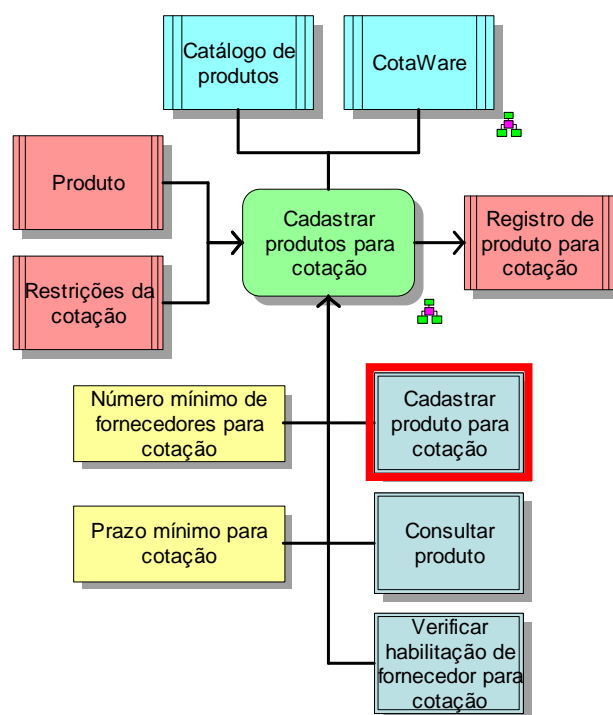


Figura 32 – Atividade apoiada por mais de um sistema

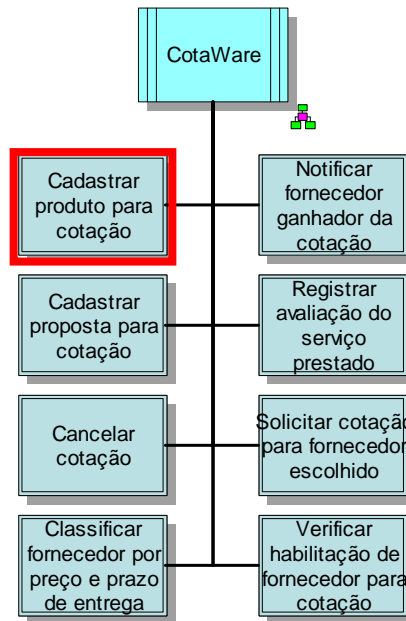


Figura 33 – Modelo de requisito de negócio do sistema “CotaWare”

#### 4.4.5 Heurística de associação de serviços candidatos com requisitos da demanda

Qualquer serviço identificado a partir da demanda deve ser diferenciado dos demais pela associação com o serviço com a demanda. Esta indicação permitirá ao projetista identificar o conjunto mínimo de serviços necessários para atender a demanda inicial.

**Heurística de associação de serviços candidatos com requisito da demanda:** Um serviço candidato identificado a partir de um requisito da demanda deve ser associado ao respectivo requisito da demanda.

#### 4.4.6 Heurística de associação de serviços com papéis

Através da associação dos serviços com os papéis que executam as atividades que deram origem aos serviços, é possível identificar quais os papéis que deverão ter acesso ao serviço candidato.

**Heurística de associação de serviços com papéis:** Um serviço candidato deve ser associado aos papéis que executam a(s) atividade(s) que lhe(s) deram origem.

Na Figura 34, o papel executor “Atendente” será o responsável pela execução dos serviços identificados na atividade “Aprovar contrato”. Com essa associação é possível ter indicativos de regras de segurança para o serviço que poderão ser tomadas na fase de projeto.

Quando o papel executor da atividade for um sistema, como exemplifica a Figura 35, significa que a tarefa a ser executada pelo serviço candidato já está implementada neste sistema.

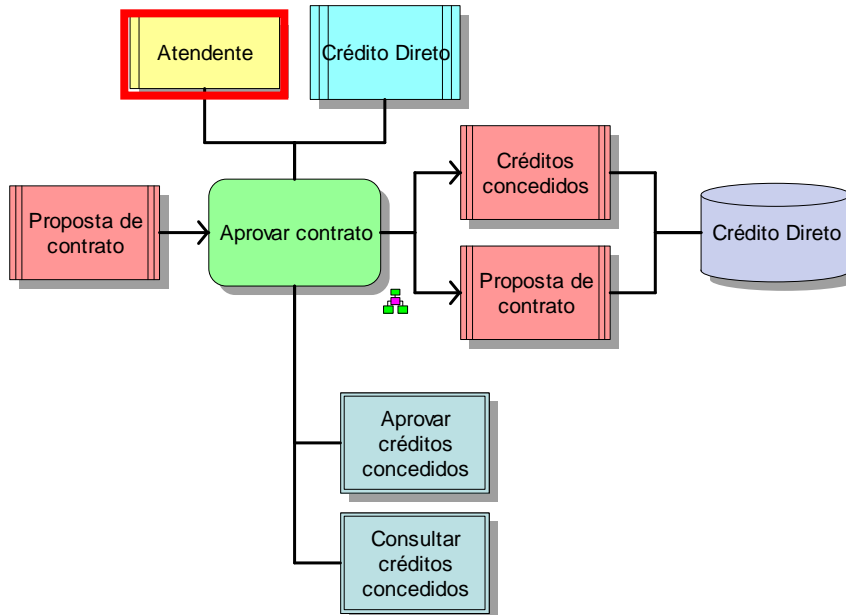


Figura 34 – Atividade “Aprovar contrato”

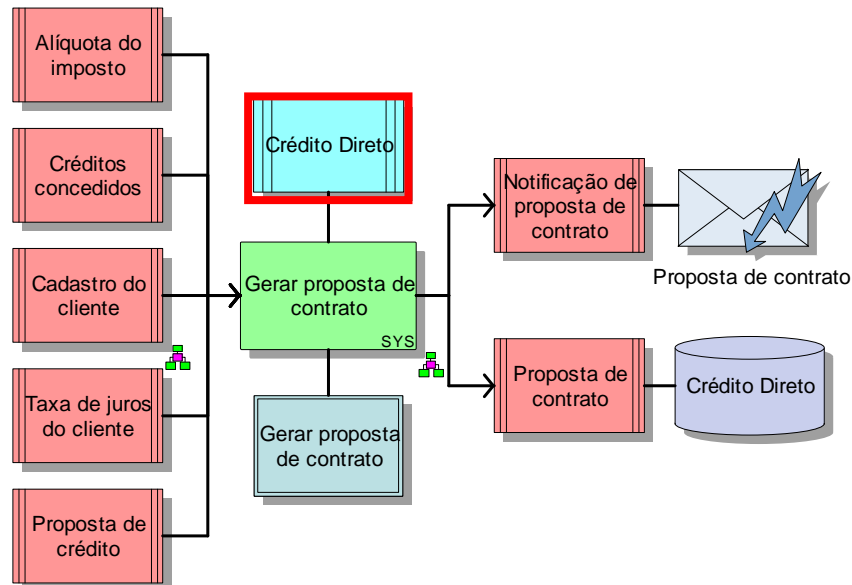


Figura 35 – Atividade “Gerar proposta de contrato”

#### 4.4.7 Heurística de associação de serviços com atividades

A associação de serviços com as atividades que o compõem facilita sua identificação no modelo de processos de negócio, caso seja necessário compreender melhor o contexto do serviço. Além disso, é possível visualizar todos os serviços que estão ligados a uma determinada atividade, permitindo analisar a possibilidade de eventualmente originar um serviço que encapsule todas as funcionalidades tratadas por ela. Esta associação permite também identificar serviços que encapsulem mais de uma atividade, o que é um indicativo da possibilidade de um serviço composto.

**Heurística de associação de serviços com atividades:** Um serviço candidato deve ser associado à(s) atividade(s) que lhe(s) deram origem.

Como exemplo, considere a atividade “Cancelar contrato” da Figura 36. Nesta atividade, temos os serviços candidatos “Escrever créditos concedidos” e “Escrever proposta de crédito”, identificados a partir dos *clusters* associados ao portador de informações “Crédito Direto” e os serviços candidatos “Cancelar créditos concedidos” e “Consultar créditos concedidos”, identificados a partir dos requisitos de negócio. Uma possível decisão do Arquiteto SOA é, depois de analisar estes serviços, optar pela implementação de todos eles um único serviço físico “Cancelar contrato” que engloba os quatro serviços candidatos identificados.



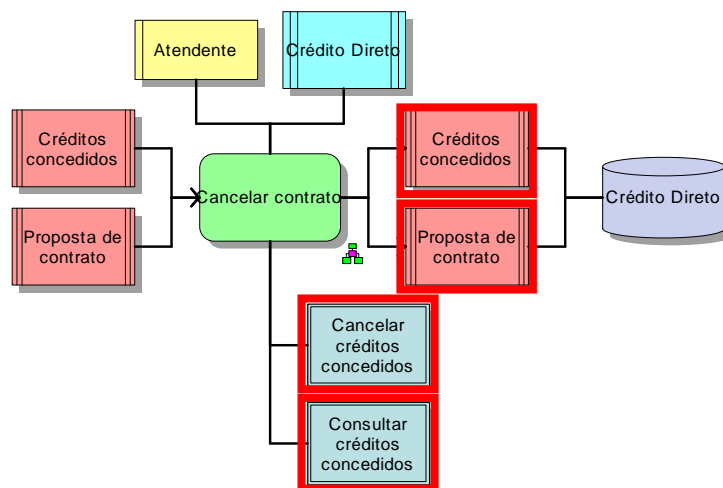


Figura 36 – Atividade “Cancelar contrato”

#### 4.4.8 Heurística de associação de serviço de dados

Serviços candidatos identificados a partir do mesmo *cluster* manipulam as mesmas informações. Assim, essa associação permite identificar serviços que possam ser agrupados em um único serviço com funcionalidades de leitura e escrita da informação que manipula.

**Heurística de associação de serviço de dados:** Serviços de dados de leitura ou escrita identificados a partir do mesmo *cluster* devem ser associados um ao outro.

Na Figura 37, a partir do *cluster* “Cadastro de cliente” seria identificado o serviço candidato “Obter cadastro do cliente”. Na Figura 38, a partir do *cluster* “Cadastro do cliente” seria identificado o serviço candidato “Escrever cadastro do cliente”. Como estes dois serviços mapeiam a mesma informação, eles devem ser associados um ao outro, pois na fase de projeto esta informação pode ajudar a agrupá-los em um único serviço físico.

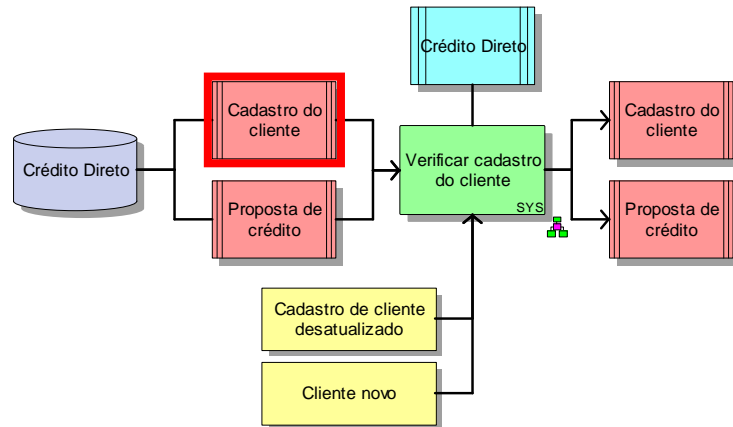


Figura 37 – Atividade “Verificar cadastro do cliente”

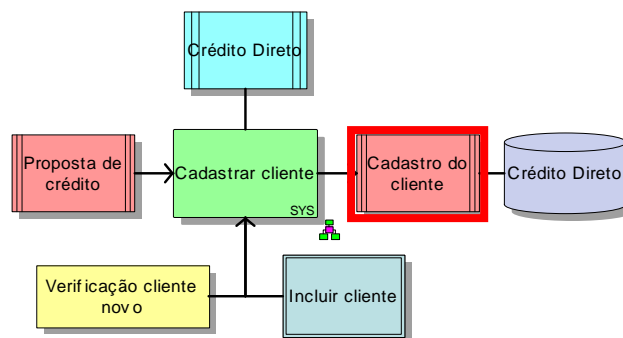


Figura 38 – Atividade “Cadastrar cliente”

#### 4.4.9 Heurística de associação de *clusters* com modelo conceitual

O relacionamento entre *clusters* e modelos de dados permite identificar portadores de informações que estejam definidos com nomes distintos, mas que representam o mesmo conjunto de dados. Assim, essa associação permite identificar serviços candidatos que possam ser agrupados em um único serviço físico de dados.

**Heurística de associação de *clusters* com modelo conceitual:** Serviços de dados identificados a partir de *clusters* que estejam ligados a uma mesma visão do modelo conceitual de dados devem ser associados uns aos outros.

Na Figura 39, os *clusters* “Proposta de contrato” e “Proposta de contrato não aprovada” estão associados à mesma visão do modelo que descreve os atributos da “Proposta de contrato” apresentado na Figura 40. Este fato serve como um indicativo de que estes serviços candidatos (“Ler proposta de contrato” e “Escrever proposta de

contrato”, por exemplo) poderiam ser considerados como métodos de um mesmo serviço físico, ou seja, como métodos que tratam da demanda no serviço “Cadastrar proposta de contrato”, por exemplo.

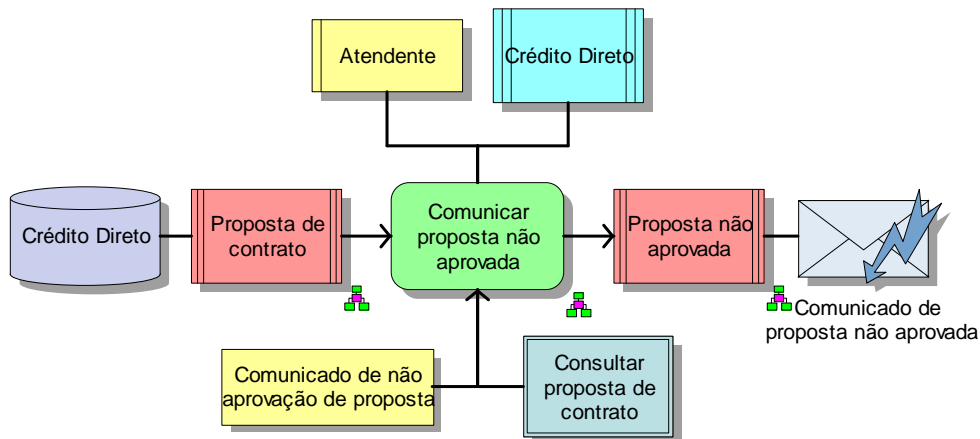


Figura 39 - Exemplo de associação entre *cluster* e modelo de dados

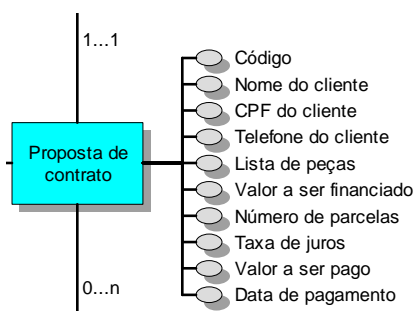


Figura 40 – Modelo de dados de “Proposta de contrato”

#### 4.4.10 Heurística de associação de serviços candidatos a partir de associações entre elementos do modelo

Os relacionamentos entre regra de negócio  $\times$  regra de negócio e de regra de negócio  $\times$  requisito de negócio, explicitados no modelo de negócio, permite a identificação da granularidade dos serviços candidatos, bem como a dependência entre serviços. Os serviços candidatos que não utilizam nenhum outro serviço candidato são considerados como de granularidade fina e são independentes de outros serviços, ou seja, são autocontidos. Os serviços que utilizam outros serviços têm granularidade mais grossa, sendo dependentes dos serviços que eles utilizam. Portanto, o serviço grosso tem maior suscetibilidade de falhas devido a uma manutenção em qualquer serviço de que ele seja dependente.

A associação de regra de negócio  $\times$  *cluster* e requisito de negócio  $\times$  *cluster* auxilia a identificar o relacionamento entre serviços identificados a partir da mesma atividade, auxiliando na fase de projeto tanto na eliminação de serviços como também no reuso de serviço.

**Heurística de associação de serviços candidatos a partir de associações entre elementos do modelo:** Um serviço candidato identificado deve ser associado aos serviços candidatos que o utilizam. Esta informação é obtida a partir da associação entre regra de negócio  $\times$  regra de negócio, regra de negócio  $\times$  requisito de negócio, regra de negócio  $\times$  *cluster* e requisito de negócio  $\times$  *cluster*, quando esta está explícita na modelagem.

#### 4.4.11 Heurística de identificação de serviços utilitários

Processos de negócio e respectivos modelos de *workflow* frequentemente incluem uma variedade de fragmentos, os quais podem ser entendidos como atividades de bloco com semântica específica e bem definida. Em particular, um fragmento (ou função recorrente em processo de negócio) pode ocorrer diversas vezes em uma mesma definição de processo [THOM *et al.*, 2007].

Diversos padrões de workflow são propostos por (THOM *et al.*, 2007), baseado nas recorrências de atividades com objetivos similares. A Figura 41 ilustra um exemplo de padrão de aprovação, que pode ser generalizado, por exemplo, como o serviço candidato “Aprovar documento”, o qual pode ser utilizado para diferentes funcionalidades na organização de acordo com o nível de aprovação do documento.

A identificação deste tipo de recorrência nos modelos permite realizar uma abstração das atividades que possuem objetivos ou aspectos similares, tornando evidente uma funcionalidade abstrata que pode ser traduzida para um serviço candidato genérico, aplicável em todos os momentos onde o padrão for identificado. Este tipo de serviço é classificado como serviço utilitário.

**Heurística de identificação de serviços utilitários:** Um serviço utilitário candidato deve ser identificado a

partir de um ou mais serviços candidatos previamente identificados quando, pela análise dos modelos de processos de negócio, for constatado que este(s) serviço(s) faz(em) parte de um padrão baseado em funções recorrentes de processos de negócio ou de um padrão relacionado a aspectos específicos de estrutura organizacional que podem ser generalizados. Exemplos destes padrões foram propostos por (THOM *et al.*, 2009a).

No modelo de exemplo da Figura 41, se o fluxo compreendido pelas atividades “Obter resultados da revisão na base de dados”, “Registrar aprovação”, “Obter número de papéis organizacionais da base de dados” e “Anular aprovações” puder ser generalizado de acordo com utilizações semelhantes em outros contextos, então a partir deste fluxo têm-se um serviço utilitário.

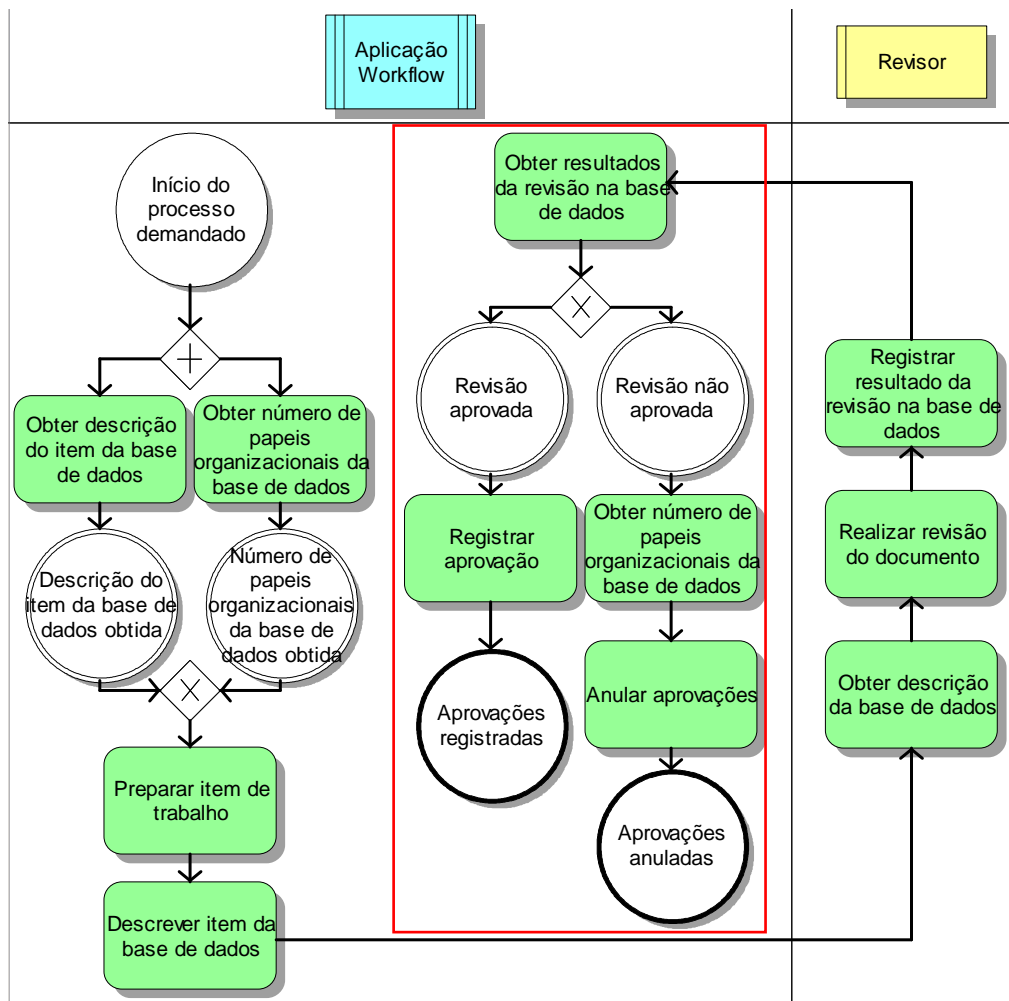


Figura 41 – Exemplo de serviço utilitário (adaptado de THOM *et al.*, 2007)

#### 4.4.12 Heurística de consolidação de serviços identificados a partir de fluxo

A heurística de consolidação de serviços identificados a partir de fluxo define as informações que devem ser extraídas através da análise dos objetos envolvidos nos modelos avaliados durante o processo de consolidação de serviços candidatos.

Abaixo segue a definição da heurística:

**Heurística de consolidação de serviços identificados a partir de fluxo:** Um serviço candidato identificado a partir de fluxo deve ter as seguintes caracterizações:

- Quantidade de atividades que compõem o fluxo (N.o

atividades automatizadas e N.o atividades apoiadas);

- Modelos onde o fluxo aparece: listagem dos modelos onde o fluxo aparece, caso o fluxo apareça em mais de um modelo;
- Número de entidades do fluxo: número de entidades que são utilizadas nas atividades que compõem o fluxo;
- Número de raias envolvidas no fluxo: número de raias diferentes onde existem atividades do fluxo.
- Números de subfluxos: número de subfluxos existentes no fluxo que gerou o serviço, por exemplo, em um fluxo de *AND* do qual o serviço foi identificado, quantos fluxos de *OR*, *XOR* e *AND* existem no subfluxo.

## 4.5 Resumo

Este capítulo apresentou as heurísticas de identificação de serviços candidatos a partir de modelos de processos de negócio e as heurísticas de consolidação de serviços candidatos.

Ao aplicar as heurísticas de identificação de serviços candidatos em um modelo, obtém-se ao final um conjunto de serviços candidatos, ou seja, um conjunto de requisitos para um projeto SOA que são candidatos à implementação.

Ao aplicar as heurísticas de consolidação de serviços candidatos nos serviços identificados, será extraído um conjunto de informações sobre estes serviços que poderão auxiliar os analistas a selecionar e priorizar os que serão implementados como *web services* ou como funcionalidades de aplicações. A decisão sobre os serviços candidatos que serão implementados e como serão implementados fica a cargo do analista, que deverá avaliar suas condições e necessidades de projeto.

As heurísticas apresentadas neste capítulo estão listadas abaixo:

### Heurísticas de identificação de serviços candidatos

- Heurística de identificação de serviços a partir de regras de negócio
- Heurística de identificação de serviços a partir de requisitos de negócio
- Heurística de identificação de serviços a partir de *cluster*
- Heurística de identificação de serviços a partir de padrão de *workflow*
  - Heurística de identificação de serviços a partir de operadores lógicos AND
  - Heurística de identificação de serviços a partir de operadores lógicos OR
  - Heurística de identificação de serviços a partir de operadores lógicos XOR
  - Heurística de identificação de serviços a partir de atividades sequenciais
  - Heurística de identificação de serviços a partir de ciclo de atividades
  - Heurística de identificação de serviços a partir de interfaces de processo
  - Heurística de identificação de serviços a partir de atividades de múltiplas instâncias

#### **Heurísticas de consolidação de serviços candidatos**

- Heurística de eliminação de serviços candidatos
- Heurística de grau de reuso de serviços candidatos
- Heurística de grau de reuso de serviços candidatos identificados a partir de atividade de múltiplas instâncias
- Heurística de associação de serviços candidatos com sistemas
- Heurística de associação de serviços candidatos com requisitos da demanda
- Heurística de associação de serviços com papéis



- Heurística de associação de serviços com atividades
- Heurística de associação de serviço de dados
- Heurística de associação de *clusters* com modelo conceitual
- Heurística de associação de serviços candidatos a partir de associações entre elementos do modelo.
- Heurística de identificação de serviços utilitários
- Heurística de consolidação de serviços identificados a partir de fluxo

O próximo capítulo aborda a automatização do método de identificação de serviços candidatos a partir da modelagem de processo de negócio. Este capítulo detalha o script de identificação automática de serviços candidatos que realiza a análise das heurísticas apresentadas neste capítulo de forma automática, a partir de um dado modelo de processo de negócio.

## **Capítulo 5: AUTOMATIZAÇÃO DO MÉTODO DE IDENTIFICAÇÃO DE SERVIÇOS A PARTIR DA MODELAGEM DE PROCESSO DE NEGÓCIO**

---

O objetivo deste capítulo é apresentar a automatização do método de identificação de serviços a partir da modelagem de processos de negócio. As implementações necessárias para a automatização das heurísticas de identificação de serviços candidatos e das heurísticas de consolidação de serviços candidatos são apresentadas em detalhes, bem como a tecnologia utilizada no desenvolvimento do script, como a ferramenta ARIS e a linguagem Java Script.

### **5.1 Requisitos do sistema de automação**

O método de identificação de serviços candidatos é baseado em heurísticas, que podem ser consideradas como requisitos para a aplicação que automatiza o método. A automatização do método de identificação de serviços foi implementada no módulo *ARIS Report Script* disponível na versão *ARIS SOA Architect* da IDS-Scheer. Este módulo possui uma biblioteca de rotinas que permitem acessar os elementos dos modelos de processos armazenados no repositório de modelos, além de permitir desenvolver novas rotinas em *Java Script* [MOZILLA, 2000].

Os símbolos dos elementos, os tipos de modelo utilizados, o formato das descrições e várias outras definições devem seguir uma diretriz para modelagem de processo. Todas as análises realizadas a partir das heurísticas de identificação de serviços candidatos nos modelos de processos de negócio para o desenvolvimento da lógica dos algoritmos levaram em consideração as Diretrizes para modelagem de processos na E&P - versão 8.1 [PETROBRAS, 2008].

A partir de um modelo EPC, as heurísticas serão executadas no diagrama principal, em seus submodelos e nos modelos de detalhamento de todas as atividades (FAD) e, ao final, é executada a etapa de consolidação. Durante a execução, também são analisados outros modelos de processos que estão relacionados ao modelo em questão. Por exemplo, outros modelos onde as atividades do processo também

aparecem, a fim de extrair as informações da etapa de consolidação. Dessa forma, ter um repositório único onde estão armazenados os modelos dos processos da organização permite uma melhor extração de informações.

## **5.2 Ferramenta ARIS**

Neste trabalho foi utilizada a ferramenta ARIS *Business Architect*, da IDS Scheer. Inicialmente, uma ferramenta CASE, engloba, desde versões mais antigas, outras metodologias e modelos (*Business Process Modeling-BPM*, *Unified Modeling Language-UML*, *Activity Based Cost – ABC*, *Simulação*, *Knowledge Management*, etc.) podendo dessa forma ser utilizada em projetos de modelagem para diferentes objetivos, assim como mais amplos em termos de escopo [BASTOS e CAMAREIRA, 2000].

Ficou conhecida justamente pelo seu potencial em BPM, sendo inclusive parceira da empresa alemã SAP (desenvolvedora do sistema integrado de gestão SAP R3). Também por isso, percebemos seu grande uso em trabalhos de modelagem de negócios, visando a implantação de sistemas integrados de gestão, particularmente o SAP R/3 [BASTOS e CAMAREIRA, 2000].

Esta ferramenta armazena seus objetos em um banco de dados próprio, permitindo também o reuso de objetos e modelos. A padronização dos objetos no ARIS é importante tanto para a gestão dos dados, por impedir que objetos que descrevem o mesmo elemento sejam duplicados, quanto para os modeladores, que através do reuso dos objetos reduzem o tempo de modelagem e ainda consomem menos espaço em disco por não inserir novos objetos.

Muitas vezes, atividades críticas extensas são repetidas em diferentes processos. A reutilização da atividade crítica pode facilitar bastante a modelagem e impedir o retrabalho. Outro ponto positivo do reuso é a possibilidade de verificar em quantos processos o dado elemento se encontra, ou seja, quantas vezes o objeto é utilizado e em que modelos.

A ferramenta ARIS disponibiliza diversos objetos para representar os elementos do negócio, o que auxilia no desenvolvimento de uma notação rica que permite representar informações bem detalhadas nos modelos. Cada objeto possui

um conjunto de parâmetros personalizáveis que receberão o registro de dados complementares ao simples modelo visual, por exemplo, a descrição detalhada de uma atividade ou uma fórmula específica para uma regra.

Ao criar um objeto no ARIS (por exemplo, uma atividade), é registrado em seu banco de dados toda a definição deste objeto, incluindo seu número de identificação, que é único. Também é criada uma ocorrência deste objeto para representar o elemento gráfico no modelo. Ao reutilizar o objeto, o ARIS não duplicará o registro no banco de dados, mas apenas continuará a registrar novas ocorrências do objeto em seus respectivos modelos. Então, uma definição de um objeto pode ter várias ocorrências espalhadas entre os diversos modelos da base.

Como a definição do objeto não se encontra nos modelos, mas no banco de dados, sendo representado por ocorrências gráficas, pode-se concluir que as ocorrências de um objeto possuem alguns atributos diferentes de seu objeto de definição, por exemplo, sua localização no modelo, o modelo onde se encontra o objeto que o representa, entre outros.

Através do módulo *ARIS Report Script*, disponível na versão *ARIS Business Architect*, é possível manipular os objetos contidos na base de dados da ferramenta, e com isso, analisar os modelos e acessar as informações dos objetos. Cada objeto no ARIS possui seus próprios atributos e métodos. Por exemplo, um objeto do tipo EPC, possui o atributo raia (*Lane*) enquanto um objeto do tipo processo possui o atributo símbolo (*Symbol*). Acessando os diversos atributos que os objetos possuem, é possível realizar análises detalhadas dos objetos e modelos e extrair suas informações.

Portanto, é possível manipular livremente qualquer objeto registrado na base de dados do ARIS através das rotinas fornecidas pela própria ferramenta, de acordo com cada tipo de objeto, e ainda manipular facilmente as variáveis através da linguagem *Java Script*, dentro do módulo de desenvolvimento de scripts da ferramenta. Através destas funcionalidades, foi possível implementar o *script* de identificação automática de serviços candidatos a partir dos modelos de processos de negócio registrados na ferramenta.

## 5.3 Modelos de classes

O diagrama de classes ilustrado na Figura 1 e Figura 2 do Anexo III demonstra todas as classes que representam algum conceito que foi utilizado na implementação da automatização do processo de identificação de serviços candidatos. O diagrama de classes ilustrado na Figura 3 do Anexo III foi desenvolvido para representar as classes que possuem funções manipuladoras (*handler*) de objetos da ferramenta ARIS.

## 5.4 Interface

A interface da automatização do método de identificação de serviços candidatos é uma única tela (Figura 42) onde o usuário deve escolher quais das heurísticas de identificação de serviços deverão ser executadas e quais relatórios de saída deverão ser gerados.

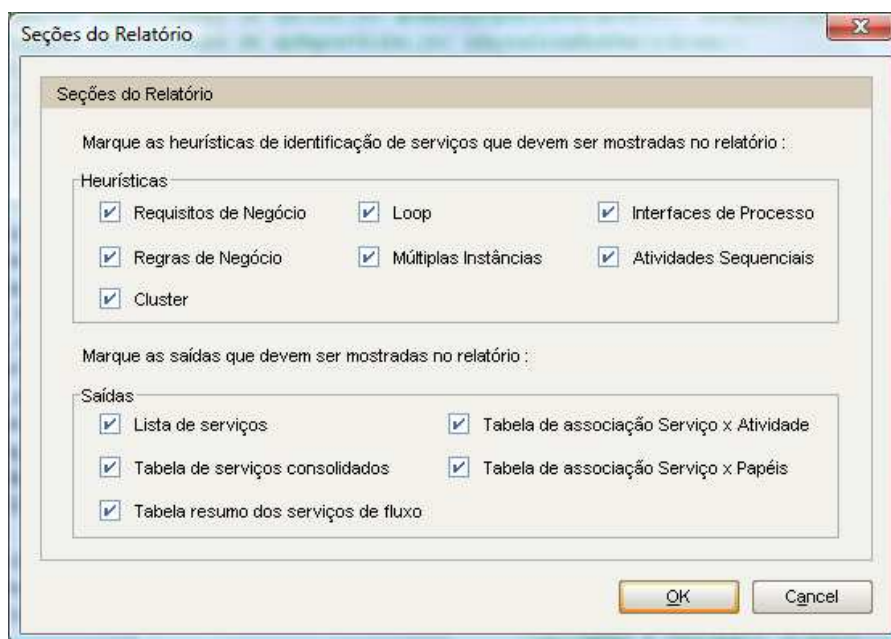


Figura 42 – Tela para escolha de heurísticas e relatórios

## 5.5 Codificação

O objetivo desta seção é detalhar os algoritmos de maior relevância que implementam as heurísticas de identificação e consolidação de serviços candidatos. Cada subseção aborda uma heurística. A descrição das heurísticas foram

apresentadas no capítulo 4. Em alguns casos, são demonstradas funções de apoio que são importantes para o funcionamento de um determinado algoritmo. Alguns nomes de funções foram alterados para facilitar a leitura e compreensão dos algoritmos.

Cabe ressaltar que durante a implementação do método de identificação de serviços houve evoluções em algumas heurísticas para que fosse possível garantir a correta identificação dos objetos nos modelos. Essas alterações foram incorporadas em [AZEVEDO *et al.*, 2009a, 2009b] onde foram atualizados os enunciados das heurísticas para contemplar essas mudanças que foram necessárias na implementação.

### **5.5.1 Heurística de identificação de serviços a partir de regras de negócio**

O algoritmo que implementa essa heurística consiste em analisar os objetos presente nos diagramas FADs das atividades contidas no modelo e buscar os objetos do tipo regra de negócio. O algoritmo que permite a identificação destes objetos é apresentado no Algoritmo 5-1:

```
{* Parâmetros:  
modeloProcesso – é um modelo do tipo EPC*}  
identificarServicoCandidatoRegraNegocio (modeloProcesso)  
  Para cada atividade do modeloProcesso Faça  
    Se a atividade for apoiada por sistema ou automatizada  
      Para cada regra de negócio existente no FAD da atividade Faça  
        Criar serviço candidato para a regra de negócio
```

Algoritmo 5-1– Identificar serviço candidato a partir de regra de negócio

### **5.5.2 Heurística de identificação de serviços a partir de requisitos de negócio**

O algoritmo que implementa essa heurística consiste em analisar os objetos presentes nos diagramas FADs das atividades contidas no modelo e buscar os objetos do tipo requisito de negócio. O algoritmo que permite a identificação destes objetos é apresentado no Algoritmo 5-2.

```

{*Parâmetros:
  modeloProcesso – é um modelo do tipo EPC*}
identificarServicoCandidatoRequisitoNegocio (modeloProcesso)
  Para cada atividade do modeloProcesso Faça
    Se a atividade for apoiada por sistema ou automatizada
      Para cada requisito de negócio existente no FAD da atividade Faça
        Criar serviço candidato para a regra de negócio

```

Algoritmo 5-2 - Identificar serviço candidato a partir de requisito de negócio

### 5.5.3 Heurística de identificação de serviços a partir de *cluster*

O algoritmo que implementa essa heurística consiste em analisar os objetos presente nos diagramas FADs das atividades contidas no modelo e buscar os objetos do tipo *cluster*. O algoritmo que permite a identificação destes objetos é apresentado no Algoritmo 5-3.

```

{*Parâmetros:
  modeloProcesso – é um modelo do tipo EPC*}
identificarServicoCandidatoCluster (modeloProcesso)
  Para cada atividade do modeloProcesso Faça
    Se a atividade for apoiada por sistema ou automatizada
      Obtém todos os clusters e guarda na listaCluster
      Elimina repetições na listaCluster
      Para cada cluster na listaCluster Faça
        Se o cluster estiver ligado a um portador de informação
          Criar serviço candidato para o cluster

```

Algoritmo 5-3 - Identificar serviço candidato a partir de *cluster*

### 5.5.4 Heurística de identificação de serviços a partir de padrões de *workflow*

Esta seção detalha as heurísticas que foram definidas a partir dos padrões de workflow.

#### 5.5.4.1 Heurística de identificação de serviços a partir de atividades sequenciais

O algoritmo que implementa essa heurística consiste em analisar as atividades presentes no modelo (EPC) e identificar as atividades sequenciais

presentes no fluxo. O algoritmo que permite a identificação deste padrão de *workflow* é apresentado no Algoritmo 5-4 e Algoritmo 5-5.

```

{*Parâmetros:
    modeloProcesso – é um modelo do tipo EPC*}
identificarServicoCandidatoSequenciaAtv (modeloProcesso)
    sequenciaAtividades = obterAtividadesSequenciais(modeloProcesso)
    Para cada sequenciaAtividades Faça
        Se todas as atividades de sequenciaAtividades forem ou apoiadas por sistema
                                                ou automatizadas
            Criar serviço candidato para a sequenciaAtividades

```

Algoritmo 5-4 - Identificar serviço candidato a partir de sequência de atividades

```

{*Parâmetros:
    modeloProcesso – É um modelo do tipo EPC*}
obterAtividadesSequenciais(modeloProcesso)
    Para cada operadorLogico do modeloProcesso Faça
        Para cada objeto posterior ligado ao operadorLogico Faça
            Se o objeto for um evento
                Vai para o próximo objeto
            Se o objeto for diferente de um operador lógico
                Para cada atividade ligada ao objeto Faça
                    sequencia.adiciona(atividade)
                Se o próximo objeto for um evento final
                    Sai do loop
                Se o próximo objeto for um evento intermediário
                    Vai para o próximo objeto
                Se o objeto for uma atividade
                    sequencia.adiciona(atividade)

    Se houver pelo menos 2 atividades em sequencia
        listaAtividadesSequenciais.adiciona(sequencia)

retorna listaAtividadesSequenciais

```

Algoritmo 5-5 – Obter atividades sequenciais

#### 5.5.4.2 Heurística de identificação de serviços a partir de ciclo de atividades

O algoritmo que implementa essa heurística consiste em analisar as atividades presentes no modelo (EPC) e identificar as sequências de atividades que formam um ciclo. O algoritmo que permite a identificação deste padrão de *workflow* é apresentado no Algoritmo 5-6.

```

{*Parâmetros:

```



```

modeloProcesso – é um modelo do tipo EPC*
identificarServicoCandidatoLoop (modeloProcesso)
  Se modeloProcesso.PossuiCiclo()
    ciclos = modeloProcesso.Ciclos()
    Para cada ciclo em ciclos Faça
      Se todas atividades do ciclo forem ou apoiadas por sistema ou automatizadas
        Criar serviço candidato para ciclo

```

Algoritmo 5-6 – Identificar serviço candidato a partir de loop

Uma observação importante é que a função *PossuiCiclo()* equivale a função *HasCycles()*, que é uma função nativa da ferramenta ARIS. Ela verifica se no modelo existe algum ciclo, retornando verdadeiro caso encontre. A função *Ciclos()* equivale a função *Cycles()*, que é uma função nativa da ferramenta ARIS. Ela retorna todos os ciclos de um modelo.

#### 5.5.4.3 Heurística de identificação de serviços a partir de interfaces de processo

O algoritmo que implementa essa heurística consiste em avaliar todas as interfaces de processo presentes no modelo (EPC) e identificar quais possuem ligação com atividades automatizadas e então seguir para o outro modelo a partir destas interfaces e verificar se existe atividade automatizada ligada à interface. O algoritmo que permite a identificação deste padrão de *workflow* é apresentado no Algoritmo 5-7.

```

{*Parâmetros:
  modeloProcesso – é um modelo do tipo EPC*}
identificarServicoCandidatoInterfaceProcesso (modeloProcesso)
  Para cada interfaceProcesso do modeloProcesso Faça
    Se interfaceProcesso possui atividade automatizada imediatamente antes
      Criar serviço candidato para interface de processo //Envia mensagem
    Se interfaceProcesso possui atividade automatizada imediatamente depois
      Criar serviço candidato para interface de processo //Recebe mensagem

```

Algoritmo 5-7 - Identificar serviço candidato a partir de interface de processo

#### 5.5.4.4 Heurística de identificação de serviços a partir de atividades de múltiplas instâncias

O algoritmo que implementa essa heurística consiste em identificar todas as atividades de múltiplas instâncias presentes no modelo (EPC) e para cada objeto identificado, criar um serviço candidato responsável pelo envio das mensagens, um

serviço candidato para a atividade de múltiplas instâncias e um serviço candidato de consolidação das mensagens. O algoritmo que permite a identificação deste objeto é apresentado no Algoritmo 5-8:

```

{*Parâmetros:
  modeloProcesso – é um modelo do tipo EPC*}
identificarServicoCandidatoMultiplasInstancia (modeloProcesso)
  Para cada atividade do modeloProcesso Faça
    Se a atividade do tipo de múltiplas instâncias Então
      Criar serviço candidato responsável por enviar as mensagens
      Criar serviço candidato para a atividade de múltiplas instâncias
      Criar serviço candidato responsável por consolidar respostas

```

Algoritmo 5-8 - Identificar serviço candidato a partir de atividade de múltiplas instâncias

### 5.5.5 Heurística de consolidação para grau de reuso

Esta seção detalha os algoritmos que implementam a heurística de consolidação para grau de reuso. Como os serviços possuem algoritmos específicos para sua identificação, o algoritmo para o cálculo do grau de reuso também é diferente para cada tipo de serviço candidato.

#### 5.5.5.1 Cálculo do grau de reuso para serviços identificados a partir de Regras de negócio e Requisitos de negócio

O cálculo do grau de reuso é similar para os serviços candidatos identificados a partir de regras de negócio e requisitos de negócio, bastando alterar o objeto de entrada da função que calcula o grau do reuso:

```

{*Parâmetros:
  objeto – pode assumir o papel do objeto regra de negócio ou requisito de negócio do ARIS *}
calcularGrauDeReusoRegraRequisitoNegocio (objeto)
  Para cada ocorrencia do objeto Faça
    Se o modelo onde a ocorrencia se encontra não for Diagrama de Sistemas
      atividade = ocorrencia.ObterAtividade( )
      grauDeReuso = grauDeReuso + atividade .ObterNumOcorrencias( )

```

Algoritmo 5-9 – Cálculo do grau de reuso para serviços identificados a partir de regras e requisitos de negócio

Uma observação importante é que a função *ObterAtividade( )* retorna o objeto que representa o modelo FAD onde a ocorrência do objeto se encontra. A função *ObterNumOcorrencias()* equivale a função *OccList()* do ARIS que retorna o

conjunto de ocorrências de um objeto. Para obter o número de ocorrências, utiliza-se esta função em conjunto com a função *length*, que é típica da linguagem e, neste caso, retorna a quantidade de objetos que pertencem ao dado conjunto.

#### 5.5.5.2 Cálculo do grau de reuso para serviços identificados a partir de Cluster

```

{*Parâmetros:
  cluster – é o objeto cluster do ARIS *}
calcularGrauDeReusoCluster (cluster)
  Para cada ocorrencia do cluster Faça
    Se o modelo onde a ocorrencia se encontra não for Diagrama de Sistemas
      atividade = ocorrencia.ObterAtividade()
    Se atividade é apoiada por sistema E cluster está ligado ao portador de
    informação
      grauDeReuso = grauDeReuso + atividade.ObterNumOcorrencias()

```

Algoritmo 5-10 – Cálculo do grau de reuso para serviços identificados a partir de cluster

#### 5.5.5.3 Cálculo do grau de reuso para serviços identificados a partir de Atividades Sequenciais

```

{*Parâmetros:
  sequenciaAtividades –é um conjunto de atividades que formam uma sequencia no
  modelo de processos*}
calcularGrauDeReusoAtvsSequenciais (sequenciaAtividades[ ])
  grauDeReuso = 1
  atvDaSequencia = sequenciaAtividades[0]
  modeloSeqOriginal = atvDaSequencia.ObterModelo()
  atvsSeqOriginal = sequenciaAtividades;
  modelosAtvDaSequencia = atvDaSequencia.ObterModelosDasOcorrencias()
  Para cada modelo em modelosAtvDaSequencia Faça
    Se (modelo != modeloSeqOriginal) E (modelo != FAD)
      sequencias = obterAtividadesSequenciais (modelo)
      Para cada sequencia em sequencias
        Se verificarConjuntoDeAtvsIguais(sequencia, atvsSeqOriginal)
          Incrementa grauDeReuso

```

Algoritmo 5-11 – Cálculo do grau de reuso para serviços identificados a partir de atividades sequenciais

Observe que a função *ObterModelo( )* equivale à função *Model( )* do ARIS que retorna o modelo onde se encontra o dado objeto. A função *ObterModelosDasOcorrencias( )* retorna todos os objetos que representam os

modelos, sem repetição, de cada ocorrência do dado objeto. A função *verificarConjuntoDeAtvsIguais(sequencia, atvsSeqOriginal)* retorna verdadeiro caso os dois conjuntos de atividades passados por parâmetro sejam iguais, senão, retorna falso.

#### 5.5.5.4 Cálculo do grau de reuso para serviços identificados a partir de loop

```

{*Parâmetros:
  loop –é um conjunto de objetos que formam um ciclo no modelo de processos*}
calcularGrauDeReusoLoop (loop)
  grauDeReuso = 1
  modeloLoop = loop[0]. ObterModelo( )
  Para cada objeto no loop Faça
    Se objeto for atividade
      modelosObjeto[ ] = objeto. ObterModelosDasOcorrencias( )
      Para cada modelo em modelosObjeto[ ] Faça
        Se (modelo != modeloLoop) E (modelo != FAD)
          Se (modelo.PossuiCiclo())
            ciclos[ ] = modelo.Ciclos()
            Para cada objeto em ciclos[ ] Faça
              proximoObjeto = ciclo[0][0].proximo( ) //Objeto no MODELO
              Se (proximoObjeto != ciclo[1])
                ciclo = inverterArrayCiclo(ciclo)
              Se (ciclo = loop)
                Incrementa grauDeReuso
  Break //Só necessita avaliar uma atividade do loop para calcular grau de reuso

```

Algoritmo 5-12 – Cálculo do grau de reuso para serviços identificados a partir de loop

Observe que a função *inverterArrayCiclo(ciclo)* retorna o *array* passado no parâmetro, invertido. A função *proximo( )* retorna o próximo objeto no modelo. Neste caso, no modelo onde o ciclo se encontra, obtém-se a segunda atividade do ciclo. Caso ela seja diferente do objeto no *array*, ou a função *Ciclos( )* retornou o *array* invertido, então não é um ciclo igual.

#### 5.5.5.5 Cálculo do grau de reuso para serviços identificados a partir de interface de processo

```

{*Parâmetros:

```

```

modelo - é o objeto modelo do ARIS
interfaceProcesso - é o objeto interface de processo do ARIS
conector - é o tipo de serviço de interface de processo que foi identificado de
acordo com a direção de busca (Envia/Recebe) utilizada como parâmetro no
algoritmo da respectiva heurística de identificação de serviços
atividadesAutomatizadas - é o conjunto de atividades automatizadas identificadas
a partir da interface de processo no algoritmo da respectiva heurística*}
calcularGrauDeReusoInterfaceProcesso (modelo, interfaceProcesso, conector,
atividadesAutomatizadas[ ])
    grauReuso = 0
    Para cada ocorrencia da interfaceProcesso Faça
        Se (ocorrencia.ObterModelo( )= "EPC") E
            (ocorrencia.ObterModelo( ) = modelo)
            Se (conector = "Envia")
                atvAutomatizadas[ ] = buscarAtivsAutomatizadas(ocorrencia, "entrada")
            Senão
                atvAutomatizadas[ ] = buscarAtivsAutomatizadas(ocorrencia, "saida")
            Se (atividadesAutomatizadas [ ] = atvAutomatizadas[ ])
                Incrementa grauDeReuso

```

Algoritmo 5-13 – Cálculo do grau de reuso para serviços identificados a partir de interface de processo

Observe que a função *buscarAtivsAutomatizadas (ocorrencia, direcao)* retorna o conjunto de atividades automatizadas que estão ligadas imediatamente antes da interface de processo, quando o parâmetro "direcao" está definido como "saida" ou o conjunto de atividades automatizadas que estão ligadas imediatamente depois da interface de processo, quando o parâmetro "direcao" está definido como "entrada".

As heurísticas de identificação de serviços candidatos a partir de padrões de *workflow* "AND", "OR" e "XOR", a heurística de associação de serviços candidatos a partir de associações entre elementos do modelo e a heurística de identificação de serviços utilitários não foram implementadas neste projeto. Os algoritmos que implementam essas heurísticas são complexos e demandariam mais tempo para a projeção e realização de testes do que o tempo disponibilizado pelo projeto neste momento. Portanto, estas implementações constituem-se como trabalhos futuros.

## 5.6 Produtos da identificação automática de serviços candidatos

O produto final da automatização da identificação de serviços candidatos é um relatório que pode ser gerado no formato .XLS (planilha do MS Excel) ou .DOC

(documento do MS Word). No formato .DOC, o conteúdo é disponibilizado com descrição e formatação específica. No caso de .XLS, a formatação foi definida juntamente com os desenvolvedores dos serviços e arquitetos SOA a fim de dar a maior flexibilidade possível para manipulação do seu conteúdo. Dessa forma, o arquiteto SOA pode ordenar e agrupar linhas/colunas, por exemplo, para priorização de acordo com a coluna “grau de reuso”.

Estas tabelas correspondem ao artefato produzido na fase de identificação de serviços do ciclo de vida de serviço contendo os serviços candidatos para o projeto. Esse artefato servirá de insumo para a fase de projeto. Os conteúdos das tabelas descrevem os serviços candidatos, que têm a função de requisitos de software para o projeto de SOA. A representação dos dados em tabelas permite a fácil manipulação das informações e o acesso às informações por softwares de terceiros, por exemplo, bancos de dados que suportam a importação de dados armazenados em planilhas.

A partir destas informações o analista avaliará quais serviços candidatos devem ser implementados, decidindo aspectos como granularidade dos serviços, desempenho, reuso e qualquer outra variável relativa à implementação, de acordo com os interesses do projeto que está em andamento.

Ao todo são disponibilizadas 5 tipos de tabelas que são organizadas a partir das heurísticas de consolidação que as definem. As tabelas são detalhadas a seguir.

- Tabela lista de serviços

Para cada heurística de identificação de serviços selecionada para execução na interface do *script* (Requisito de negócio, Regra de negócio, Cluster, Loop, Múltiplas Instâncias, Interfaces de Processo, Atividades Sequenciais), é gerada uma tabela com a lista de serviços contendo as seguintes informações: Índice (número autoincremental que identifica o serviço); Nome do serviço; Tipo (se é serviço de dados ou serviço de negócio); Atividades (a(s) atividade(s) da(s) qual(is) o serviço candidato foi identificado); Papéis Executores (nome dos papéis executores da(s) atividade(s) da qual o serviço candidato foi identificado); Sistema de apoio (os sistemas que suportam a(s) atividade(s) da(s) qual(is) o serviço candidato foi identificado); Descrição do serviço. A Tabela 5-1 exemplifica uma lista de serviços para regras de negócio.

Tabela 5-1 – Lista de serviço para regras de negócio

Índice	Nome	Tipo	Atividade(s)	Executores	Sistemas de apoio	Descrição
1	Cliente novo		Cadastrar cliente;	Crédito Direto;	Crédito Direto;	Um cliente é novo se não existir cadastro de cliente com mesmo CPF que o informado na proposta de
2	Cadastro de cliente desatualizado		Atualizar cadastro do cliente;	Crédito Direto;	Crédito Direto;	O cadastro do cliente está desatualizado se o telefone, endereço e renda informados na proposta de crédito
3	Comprometimento do limite de crédito		Comprometer limite de crédito;	Crédito Direto;	Crédito Direto;	No comprometimento do crédito para o cliente a situação de crédito concedido deve ser igual a
4	Cálculo das alíquotas de impostos		Calcular alíquota de imposto;	Crédito Direto;	Crédito Direto;	Deverá ser calculada a alíquota de imposto para cada tipo de imposto existente.

Para os serviços candidatos identificados a partir da heurística de cluster, ciclo, atividades sequenciais, interfaces de processo e múltiplas instâncias existem detalhes específicos que são detalhados a seguir:

Para os serviços candidatos identificados a partir de *cluster*, foram incluídas as seguintes informações na tabela de consolidação: Tipo (“Serviço de dados”); Clusters Entrada (número de vezes em que o *cluster* aparece como entrada de uma atividade); Clusters Saída (número de vezes em que o *cluster* aparece como saída de uma atividade); Modelo conceitual (nome do modelo conceitual que representa os conceitos tratados pelo *cluster*). A Tabela 5-2 ilustra um exemplo da lista de serviços para *cluster*.

Tabela 5-2 – Lista de serviços para *cluster*

Índice	Nome	Tipo	Atividade(s)	Papéis Executores	Sistemas de apoio	Modelo Conceitual	Entidades	Entrada atividade	Saída atividade	Descrição
39	Proposta de crédito	Dado	Alterar proposta de crédito; Verificar cadastro do cliente; Cancelar proposta de crédito;	Analista de crédito; Crédito Direto;	Crédito Direto;	-	-	10	3	Representa as informações do solicitante e dos produtos a serem financiados necessárias para solicitação de crédito, contendo: - código da proposta de
40	Cadastro do cliente	Dado	Verificar cadastro do cliente; Cadastrar cliente; Atualizar cadastro do cliente;	Crédito Direto;	Crédito Direto;	Cadastro do cliente	Cliente	4	3	Representa as informações do cadastro do cliente, contendo: - nome, - CPF, -
41	Créditos concedidos	Dado	Cancelar contrato; Comprometer limite de crédito; Cancelar contrato de risco; Aprovar contrato;	Atendente; Crédito Direto;	Crédito Direto;	-	-	8	5	Representa as informações dos créditos concedidos aos clientes, contendo para cada concessão de crédito: - código da proposta de crédito, - CPF do cliente, - valor total concedido, - número de parcelas, - taxa de juros, - situação de crédito (comprometido, cancelado pelo analista de crédito, aprovado pelo analista de crédito, rejeitado pelo cliente

No caso de serviço identificado a partir de ciclos, o nome do serviço é preenchido com a palavra “Ciclo -” + um número correspondente à ordem em que o ciclo foi identificado. Por exemplo, “Ciclo – 1”, “Ciclo – 2”, e assim sucessivamente. O mesmo ocorre para atividades sequenciais, onde o nome do serviço é preenchido com “Sequencia– 1”, “Sequencia– 2”, e assim sucessivamente. A Tabela 5-3 ilustra um exemplo da lista de serviços para loop.

Tabela 5-3 – Lista de serviços para loop

Índice	Nome	Tipo	Atividade(s)	Papéis Executores	Sistemas de apoio	Descrição
34	Loop - 1		Comprometer limite de crédito; Calcular alíquota de imposto; Determinar taxa de juros a ser cobrada do cliente; Gerar proposta de contrato; Analisar contrato; Alterar proposta de crédito;	Crédito Direto; Analista de crédito;	Crédito Direto;	-

No caso dos serviços identificados a partir de interfaces de processo a composição do seu nome e a análise das atividades que compõem o serviço variam de acordo com a posição da interface de processo. Para as interfaces que estão

iniciando o processo, o nome é preenchido com o nome do modelo + “(RECEBE)” + o nome da interface. O campo Atividades é preenchido com os nomes das atividades automatizadas imediatamente posteriores à interface em cada caminho saindo da interface. Para as interfaces que estão saindo do processo, o nome é preenchido com o nome do modelo + “(ENVIA)” + o nome da interface. O campo Atividades é preenchido com os nomes das atividades automatizadas imediatamente anteriores à interface em cada caminho chegando na interface. A Tabela 5-4 ilustra um exemplo da lista de serviços para interfaces de processo.

Tabela 5-4 – Lista de serviços para interfaces de processo

Índice	Nome	Tipo	Atividade(s)	Papéis Executores	Sistemas de apoio	Descrição
35	Analisar pedido de crédito (Recebe) Receber proposta de crédito		Verificar cadastro do cliente;	Crédito Direto;	Crédito Direto;	-

No caso de serviço identificado a partir de atividades de múltiplas instâncias, são gerados três serviços e o campo nome é preenchido da seguinte forma: para o serviço remetente, o nome é preenchido com “Enviar mensagem para” + nome da atividade de múltiplas instâncias; para o serviço correspondente às instâncias, o do serviço é igual ao nome da atividade; para o serviço de consolidação, o nome do serviço é preenchido com “Consolidar mensagens de” + nome da atividade. A Tabela 5-5 ilustra um exemplo da lista de serviços para interfaces de processo.

Tabela 5-5 – Lista de serviços para múltiplas instâncias

Índice	Nome	Tipo	Atividade(s)	Papéis Executores	Sistemas de apoio	Descrição
35	Calcular alíquota de imposto		Calcular alíquota de imposto;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto calcula a alíquota do imposto. As informações necessárias são: créditos concedidos e impostos sobre crédito. A informação gerada é: alíquota do imposto.
36	Enviar mensagem para Calcular alíquota de imposto	Dado		-	-	Este serviço emite mensagem para atividade de múltipla instância Calcular alíquota de imposto
37	Consolidar mensagens de Calcular alíquota de imposto	Dado		-	-	Este serviço consolida o produto da atividade de múltipla instância Calcular alíquota de imposto

- Tabela de serviços consolidados

A tabela de serviços consolidados corresponde às informações geradas na fase de consolidação do método de identificação de serviços. Os procedimentos para a consolidação das informações são aplicados a todos os serviços identificados na fase anterior.

A tabela de consolidação de serviços candidatos possui as seguintes informações: Índice (número autoincremental que identifica o serviço); Nome do serviço; Grau de reuso (número de repetições do serviço nos modelos); Múltiplas instâncias (indica se o serviço é de múltiplas instâncias); Sistema de apoio (os



sistemas que suportam a(s) atividade(s) da qual o serviço candidato foi identificado); Papéis Executores (nome dos papéis executores da(s) atividade(s) da qual o serviço candidato foi identificado) e Modelo conceitual (nome dos modelos conceituais ligados ao serviço identificado a partir de *cluster*).

As duas últimas colunas indicam se o requisito já está implementado, no caso de serviço identificado a partir de requisito de negócio; e se o serviço refere-se diretamente à demanda, no caso do serviço ter sido identificado a partir de um elemento do modelo de processos que está relacionado à demanda. A Tabela 5-6 ilustra um exemplo da tabela de consolidação dos serviços candidatos.

Tabela 5-6 – Consolidação dos serviços candidatos

Índice	Serviço candidato	Grau de reuso	Múltiplas instâncias	Sistemas	Papéis	Modelo conceitual	Req. Implementado	Req. Demanda
1	Cliente novo	4	NÃO	Crédito Direto;	Crédito Direto;	-		
2	Cadastro de cliente desatualizado	4	NÃO	Crédito Direto;	Crédito Direto;	-		
3	Comprometimento do limite de crédito	2	NÃO	Crédito Direto;	Crédito Direto;	-		
4	Determinação da taxa de juros	2	NÃO	Crédito Direto;	Crédito Direto;	-		
5	Valor de limite de crédito	2	NÃO	Crédito Direto;	Crédito Direto;	-		
6	Cliente novo	4	NÃO	Crédito Direto;	Crédito Direto;	-		

- Tabela de associação de Serviço X Atividades

A tabela de associação de Serviço × Atividade relaciona as atividades que fazem parte de um dado serviço. Ela permite rastrear, por exemplo, quantos serviços estão relacionados a uma dada atividade ou, do ponto de vista inverso, quantas atividades estão relacionadas a um serviço. A Tabela 5-7 ilustra um exemplo da tabela de associação de Serviços × Atividades.

Tabela 5-7 – Associação Serviço x Atividade

Serviço candidato	Cadastrar cliente	Atualizar cadastro do cliente	Comprometer limite de crédito	Determinar taxa de juros a ser cobrada do cliente	Verificar limite de crédito do cliente	Verificar cadastro do cliente
1 - Cliente novo	X					
2 - Cadastro de cliente desatualizado		X				
3 - Comprometimento do limite de crédito			X			
4 - Determinação da taxa de juros				X		
5 - Valor de limite de crédito					X	
6 - Cliente novo						X
7 - Cadastro de cliente desatualizado						X

- Tabela de associação de Serviço × Papéis

A tabela de associação de Serviço × Papéis relaciona os papéis que executam um dado serviço. Ela permite rastrear, por exemplo, quais papéis executam um dado serviço ou que serviços um papel executa. O mapeamento entre papéis e serviços permitirá, por exemplo, ao analista configurar opções de segurança em relação ao

acesso aos serviços. A Tabela 5-8 ilustra um exemplo da tabela de associação de Serviços × Papéis.

Tabela 5-8 – Associação Serviço x Papéis

Serviço candidato	Crédito Direto	Atendente	Analista de crédito	Cliente
1 - Cliente novo	X			
2 - Cadastro de cliente desatualizado	X			
3 - Comprometimento do limite de crédito		X		
4 - Determinação da taxa de juros			X	
5 - Valor de limite de crédito	X			
6 - Cliente novo				X
7 - Cadastro de cliente desatualizado				X

- Tabela resumo dos serviços de fluxo

A partir da análise dos arquitetos SOA, identificou-se a necessidade de uma tabela consolidando informações específicas para serviços identificados a partir de fluxo. Neste caso, as heurísticas de ciclos, atividades sequenciais e interface de processo, as quais foram implementadas neste trabalho. Esta tabela apresenta o nome dos modelos que possuem ocorrências do serviço; número de entidades envolvidas nas atividades (quando os *clusters* estão associados a diagramas entidade-relacionamento); número de raias distintas em que as atividades do serviço aparecem. As informações produzidas auxiliaram os arquitetos SOA a definir melhor as orquestrações de serviços. A Tabela 5-9 ilustra um exemplo da tabela de resumo de fluxo.

Tabela 5-9 – Resumo dos serviços de fluxo

Índice	Serviço candidato	Nº atividades	Ocorrência em modelos	Nº entidades	Nº raias
34	Loop - 1	6	Analisar pedido de crédito; Analisar pedido de crédito;	1	2
38	Analisar pedido de crédito (Recebe) Receber proposta de crédito	1		-	1
39	Cancelar proposta de crédito_Comunicar proposta não aprovada	2	Analisar pedido de crédito; Analisar pedido de crédito;	-	2
40	Comprometer limite de crédito_Calcular alíquota de imposto_Determinar taxa de juros a ser cobrada do cliente_Gerar proposta de contrato_Analisar contrato	5	Analisar pedido de crédito; Analisar pedido de crédito;	1	2
41	Comunicar não aprovação de contrato de risco_Cancelar contrato de risco_Comunicar proposta não aprovada	3	Analisar pedido de crédito; Analisar pedido de crédito;	-	2

## 5.1 Resumo

Este capítulo apresentou a automatização do processo de identificação de serviços candidatos, exemplificou a implementação das heurísticas com alguns

algoritmos e demonstrou os relatórios de saída que contêm informações consolidadas dos serviços candidatos identificados.

O relatório de serviços candidatos representa um artefato da fase de identificação de serviços, do ciclo de vida de serviços (ou levantamento de requisitos, comparando com um ciclo de vida tradicional), contendo os requisitos (serviços candidatos) para o processo de negócio que teve seu modelo analisado pelo script de identificação de serviços candidatos.

Isso significa que o processo de negócio avaliado possui um conjunto de requisitos que podem ser implementados através de serviços que irão automatizar ou apoiar o processo como um todo ou somente uma ou mais atividades.

Este relatório servirá como insumo para a próxima fase do ciclo de desenvolvimento (análise e projeto de serviços), onde auxiliará o analista a tomar decisões quanto aos serviços que serão implementados.

## **Capítulo 6: RESULTADOS OBTIDOS EM ESTUDO DE CASO COM O MÉTODO DE IDENTIFICAÇÃO AUTOMATIZADA DE SERVIÇOS CANDIDATOS**

---

A automatização do método de identificação de serviços foi avaliada em um estudo de caso em relação ao processo “Analisar pedido de crédito”, que se encontra detalhado no anexo II. O objetivo deste teste foi avaliar os benefícios do uso de uma ferramenta para identificação automática de serviços candidatos, bem como produzir insumos para propostas de novas tabelas de informações geradas, para auxiliar o arquiteto SOA.

Um dos benefícios identificados com o uso do método implementado foi a rapidez para identificar e gerar serviços candidatos. A tarefa que anteriormente demandava semanas, e cujos resultados eram revisados várias vezes em função dos ajustes necessários, passou a ser realizada de forma imediata. Isto permitiu que os resultados fossem regerados a qualquer momento. Além disso, uma modificação no modelo de processos passou a requerer apenas a re-execução da ferramenta para regerar os relatórios.

O resultado da execução do script de identificação de serviços candidatos a partir do modelo “Analisar pedido de crédito” é detalhado a seguir (Tabela 6-1, Tabela 6-2, Tabela 6-3, Tabela 6-4, Tabela 6-5, Tabela 6-6, Tabela 6-7, Tabela 6-8, Tabela 6-9, Tabela 6-10, Tabela 6-11):

Tabela 6-1 – Serviços identificados a partir da heurística de regras de negócio

Índice	Nome	Tipo	Atividade(s)	Papéis Executores	Sistemas de apoio	Descrição
1	Cliente novo		Cadastrar cliente;	Crédito Direto;	Crédito Direto;	Um cliente é novo se não existir cadastro de cliente com mesmo CPF que o informado na proposta de crédito.
2	Cadastro de cliente desatualizado		Atualizar cadastro do cliente;	Crédito Direto;	Crédito Direto;	O cadastro do cliente está desatualizado se o telefone, endereço e renda informados na proposta de crédito forem diferentes das informações do cliente existentes na base de dados. A comparação entre proposta de crédito e cadastro do cliente deve ser feita pelo CPF.
3	Comprometimento do limite de crédito		Comprometer limite de crédito;	Crédito Direto;	Crédito Direto;	No comprometimento do crédito para o cliente a situação de crédito concedido deve ser igual a comprometido.
4	Valor limite para taxa de juros		Determinar taxa de juros a ser cobrada do cliente;	Crédito Direto;	Crédito Direto;	O valor limite para a taxa de juros é de 12%.
5	Determinação da taxa de juros		Determinar taxa de juros a ser cobrada do cliente;	Crédito Direto;	Crédito Direto;	A taxa de juros deve ser determinada de acordo com o valor total solicitado para financiamento e o número de parcelas.
6	Valor de limite de crédito		Verificar limite de crédito do cliente;	Crédito Direto;	Crédito Direto;	O limite de crédito do cliente é igual a 20% da sua renda menos o valor mensal referentes às parcelas ainda em aberto dos créditos anteriormente concedidos ao cliente.
7	Cliente novo		Verificar cadastro do cliente;	Crédito Direto;	Crédito Direto;	Um cliente é novo se não existir cadastro de cliente com mesmo CPF que o informado na proposta de crédito.
8	Cadastro de cliente desatualizado		Verificar cadastro do cliente;	Crédito Direto;	Crédito Direto;	O cadastro do cliente está desatualizado se o telefone, endereço e renda informados na proposta de crédito forem diferentes das informações do cliente existentes na base de dados. A comparação entre proposta de crédito e cadastro do cliente deve ser feita pelo CPF.
9	Comunicado de não aprovação de proposta		Comunicar proposta não aprovada;	Atendente;	Crédito Direto;	O Atendente deve informar ao cliente caso a sua proposta de contrato não tenha sido aprovada.
10	Comunicado de não aprovação de contrato de risco		Comunicar não aprovação de contrato de risco;	Analista de crédito;	Crédito Direto;	O Analista de crédito deve comunicar ao Atendente caso o contrato não seja aprovado por ser considerado de risco.
11	Alteração da proposta de crédito		Alterar proposta de crédito;	Analista de crédito;	Crédito Direto;	O valor total de crédito concedido e/ou o número de parcelas devem ser reduzidos e a taxa de juros aumentada quando for necessário reduzir o risco da proposta de crédito.
12	Cálculo das alíquotas de impostos		Calcular alíquota de imposto;	Crédito Direto;	Crédito Direto;	Deverá ser calculada a alíquota de imposto para cada tipo de imposto existente.

13	Norma de cálculo da alíquota de impostos		Calcular alíquota de imposto;	Crédito Direto;	Crédito Direto;	O cálculo da alíquota de impostos deve considerar as normas da legislação local vigente.
----	--	--	-------------------------------	-----------------	-----------------	--

Tabela 6-2 – Serviços identificados a partir da heurística de requisitos de negócio

Índice	Nome	Tipo	Atividade(s)	Papéis Executores	Sistemas de apoio	Descrição
14	Incluir cliente		Cadastrar cliente;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto deve criar cadastro do cliente se o cliente for novo (de acordo com a regra de negócio Verificação cliente novo), contendo as seguintes informações: - nome, - telefone, - endereço, - renda, - data de cadastro.
15	Atualizar informações do cliente		Atualizar cadastro do cliente;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto deve atualizar as seguintes informações do cliente (de acordo com a regra de negócio Verificação cadastro cliente desatualizado): - telefone, - endereço, - renda.
16	Consultar informações do cliente		Atualizar cadastro do cliente;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto deve consultar as seguintes informações do cliente: - nome, - CPF, - identidade, - telefone, - endereço, - renda, - data da última atualização, - data de cadastro.
17	Gerar proposta de contrato		Gerar proposta de contrato;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto deve gerar a proposta de contrato, contendo: - código da proposta de crédito, - nome do cliente, - CPF, - endereço do cliente, - telefone do cliente, - lista de peças, - valor a ser financiado, - número de parcelas, - taxa de juros, - para cada parcela: - data a ser realizado o pagamento, - valor a ser pago, - juros correspondentes à multa por atraso.
18	Consultar informações do cliente		Gerar proposta de contrato;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto deve consultar as seguintes informações do cliente: - nome, - CPF, - identidade, - telefone, - endereço, - renda, - data da última atualização, - data de cadastro.
19	Comprometer limite de crédito		Comprometer limite de crédito;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto deve comprometer o limite de crédito do cliente para a proposta de crédito aprovada, contendo: - CPF do cliente, - código da proposta de crédito, - valor total concedido, - número de parcelas, - situação de crédito (comprometido).
20	Registrar taxa de juros		Determinar taxa de juros a ser cobrada do cliente;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto deve registrar a taxa de juros (de acordo com a regra de negócio Determinação da taxa de juros) e parcelas dos créditos concedidos, contendo: - taxa de juros, - para cada parcela: - data a ser realizado o pagamento, - valor a ser pago, - juros correspondentes à multa por atraso, - valor pago, - data em que o pagamento foi realizado.
21	Cancelar proposta		Cancelar proposta de crédito;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto deve cancelar a proposta do cliente.
22	Verificar limite de crédito do cliente		Verificar limite de crédito do cliente;	Crédito Direto;	Crédito Direto;	O sistema crédito direto deve verificar o limite de crédito do cliente de acordo com a regra de negócio Verificação de limite de crédito.
23	Consultar informações do cliente		Verificar cadastro do cliente;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto deve consultar as seguintes informações do cliente: - nome, - CPF, - identidade, - telefone, - endereço, - renda, - data da última atualização, - data de cadastro.

24	Consultar proposta de contrato		Comunicar proposta não aprovada;	Atendente;	Crédito Direto;	O Atendente deve consultar as seguintes informações da proposta de contrato: - código da proposta de crédito, - nome do cliente, - CPF, - lista de peças, - valor a ser financiado, - número de parcelas, - taxa de juros, - para cada parcela: - data a ser realizado o pagamento, - valor a ser pago, - juros correspondentes à multa por atraso.
25	Consultar proposta de contrato		Verificar condições de contrato com cliente;	Atendente; Cliente;	Crédito Direto;	O Atendente deve consultar as seguintes informações da proposta de contrato: - código da proposta de crédito, - nome do cliente, - CPF, - lista de peças, - valor a ser financiado, - número de parcelas, - taxa de juros, - para cada parcela: - data a ser realizado o pagamento, - valor a ser pago, - juros correspondentes à multa por atraso.
26	Consultar proposta de contrato para análise		Analisar contrato;	Analista de crédito;	Crédito Direto;	O Analista de Crédito Contratos deve consultar as seguintes informações da proposta de contrato: - código da proposta de crédito, - nome do cliente, - CPF, - lista de peças, - valor a ser financiado, - número de parcelas, - taxa de juros, - para cada parcela: - data a ser realizado o pagamento, - valor a ser pago, - juros correspondentes à multa por atraso.
27	Analisar contrato		Analisar contrato;	Analista de crédito;	Crédito Direto;	O Analista de Crédito Contratos deve analisar o contrato ajustando os créditos concedidos na proposta de contrato para cancelado ou aprovado de acordo com a análise realizada.
28	Consultar créditos concedidos		Analisar contrato;	Analista de crédito;	Crédito Direto;	O Atendente deve consultar as seguintes informações de cada concessão de crédito feita para o cliente anteriormente: - CPF do cliente, - valor total concedido, - número de parcelas, - taxa de juros, - situação de crédito, - para cada parcela: - data a ser realizado o pagamento, - valor a ser pago, - juros correspondentes à multa por atraso, - valor pago, - data em que o pagamento foi realizado.
29	Cancelar contrato de risco		Cancelar contrato de risco;	Atendente;	Crédito Direto;	O Analista de Crédito Contratos deve cancelar os créditos concedidos para o cliente.
30	Cancelar créditos concedidos		Cancelar contrato de risco;	Atendente;	Crédito Direto;	O Atendente deve cancelar os créditos concedidos para o cliente.
31	Consultar créditos concedidos		Aprovar contrato;	Atendente;	Crédito Direto;	O Atendente deve consultar as seguintes informações de cada concessão de crédito feita para o cliente anteriormente: - CPF do cliente, - valor total concedido, - número de parcelas, - taxa de juros, - situação de crédito, - para cada parcela: - data a ser realizado o pagamento, - valor a ser pago, - juros correspondentes à multa por atraso, - valor pago, - data em que o pagamento foi realizado.
32	Aprovar créditos concedidos		Aprovar contrato;	Atendente;	Crédito Direto;	O Atendente deve aprovar os créditos concedidos para atendimento da proposta do cliente.
33	Cancelar créditos concedidos		Cancelar contrato;	Atendente;	Crédito Direto;	O Atendente deve cancelar os créditos concedidos para o cliente.
34	Consultar créditos concedidos		Cancelar contrato;	Atendente;	Crédito Direto;	O Atendente deve consultar as seguintes informações de cada concessão de crédito feita para o cliente anteriormente: - CPF do cliente, - valor total concedido, - número de parcelas, - taxa de juros, - situação de crédito, - para cada parcela: - data a ser realizado o pagamento, - valor a ser pago, - juros correspondentes à multa por atraso, - valor pago, - data em que o pagamento foi realizado.



35	Consultar créditos de contrato de risco		Alterar proposta de crédito;	Analista de crédito;	Crédito Direto;	O Analista de Crédito Contratos deve consultar as seguintes informações concessão de crédito para o cliente: - código da proposta de crédito, - CPF do cliente, - valor total concedido, - número de parcelas, - taxa de juros, - situação de crédito, - para cada parcela: - data a ser realizado o pagamento, - valor a ser pago, - juros correspondentes à multa por atraso, - valor pago, - data em que o pagamento foi realizado.
36	Registrar alíquota de imposto		Calcular alíquota de imposto;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto deve registrar cada alíquota de imposto calculada contendo: - código da proposta de crédito, - CPF do cliente, - montante de crédito concedido, - nome do imposto - fórmula da alíquota - valor da alíquota do imposto.
37	Calcular alíquota de imposto		Calcular alíquota de imposto;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto calcula a alíquota de imposto baseada no montante que consta no pedido de crédito.

Tabela 6-3 – Serviços identificados a partir da heurística de Loop

Índice	Nome	Tipo	Atividade(s)	Papéis Executores	Sistemas de apoio	Descrição
38	Loop - 1		Comprometer limite de crédito; Calcular alíquota de imposto; Determinar taxa de juros a ser cobrada do cliente; Gerar proposta de contrato; Analisar contrato; Alterar proposta de crédito;	Crédito Direto; Analista de crédito;	Crédito Direto;	-

Tabela 6-4 – Serviços identificados a partir da heurística de atividades de múltiplas instâncias

Índice	Nome	Tipo	Atividade(s)	Papéis Executores	Sistemas de apoio	Descrição
39	Calcular alíquota de imposto		Calcular alíquota de imposto;	Crédito Direto;	Crédito Direto;	O sistema Crédito Direto calcula a alíquota do imposto. As informações necessárias são: créditos concedidos e impostos sobre crédito. A informação gerada é: alíquota do imposto.
40	Enviar mensagem para Calcular alíquota de imposto	Dado		-	-	Este serviço emite mensagem para atividade de múltipla instância Calcular alíquota de imposto
41	Consolidar mensagens de Calcular alíquota de imposto	Dado		-	-	Este serviço consolida o produto da atividade de múltipla instância Calcular alíquota de imposto

Tabela 6-5 – Serviços identificados a partir da heurística de interface de processo

Índice	Nome	Tipo	Atividade(s)	Papéis Executores	Sistemas de apoio	Descrição
42	Analisar pedido de crédito (Recebe) Receber proposta de crédito		Verificar cadastro do cliente;	Crédito Direto;	Crédito Direto;	-

Tabela 6-6 – Serviços identificados a partir de heurística de atividades sequenciais

Índice	Nome	Tipo	Atividade(s)	Papéis Executores	Sistemas de apoio	Descrição
43	Cancelar proposta de crédito_Comunicar proposta não aprovada		Cancelar proposta de crédito; Comunicar proposta não aprovada;	Crédito Direto; Atendente;	Crédito Direto;	-

44	Comprometer limite de crédito_Calcular alíquota de imposto_Determinar taxa de juros a ser cobrada do cliente_Gerar proposta de contrato_Analisar contrato		Comprometer limite de crédito; Calcular alíquota de imposto; Determinar taxa de juros a ser cobrada do cliente; Gerar proposta de contrato; Analisar contrato;	Crédito Direto; Analista de crédito;	Crédito Direto;	-
45	Comunicar não aprovação de contrato de risco_Cancelar contrato de risco_Comunicar proposta não aprovada		Comunicar não aprovação de contrato de risco; Cancelar contrato de risco; Comunicar proposta não aprovada;	Analista de crédito; Atendente;	Crédito Direto;	-

Tabela 6-7 – Serviços identificados a partir da heurística de *cluster*

Índice	Nome	Tipo	Atividade(s)	Papéis Executores	Sistemas de apoio	Modelo Conceitual	Entidades	Entrada atividade	Saída atividade	Descrição
46	Proposta de crédito	Dado	Alterar proposta de crédito; Verificar cadastro do cliente; Cancelar proposta de crédito;	Analista de crédito; Crédito Direto;	Crédito Direto;	-	-	10	3	Representa as informações do solicitante e dos produtos a serem financiados necessárias para solicitação de crédito, contendo: - código da proposta de crédito, - nome do solicitante, - CPF, - identidade, - telefone do solicitante, - endereço do solicitante, - lista de peças, - valor a ser financiado, - número de parcelas, - renda, - data de cadastro, - responsável pelo cadastro, - resultado da verificação de crédito (aprovado ou reprovado).
47	Cadastro do cliente	Dado	Verificar cadastro do cliente; Cadastrar cliente; Atualizar cadastro do cliente;	Crédito Direto;	Crédito Direto;	Cadastro do cliente	Cliente	4	3	Representa as informações do cadastro do cliente, contendo: - nome, - CPF, - identidade, - telefone, - endereço, - renda, - data da última atualização, - data de cadastro.
48	Créditos concedidos	Dado	Cancelar contrato; Comprometer limite de crédito; Cancelar contrato de risco; Aprovar contrato;	Atendente; Crédito Direto;	Crédito Direto;	-	-	8	5	Representa as informações dos créditos concedidos aos clientes, contendo para cada concessão de crédito: - código da proposta de crédito, - CPF do cliente, - valor total concedido, - número de parcelas, - taxa de juros, - situação de crédito (comprometido, cancelado pelo analista de crédito, aprovado pelo analista de crédito, rejeitado pelo cliente, aprovado pelo cliente), - para cada parcela: - data a ser realizado o pagamento, - valor a ser pago, - juros correspondentes à multa por atraso, - valor pago, - data em que o pagamento foi realizado.
49	Proposta de contrato	Dado	Cancelar contrato de risco; Analisar contrato; Cancelar contrato; Aprovar contrato; Verificar condições de contrato com cliente; Gerar proposta de contrato;	Atendente; Analista de crédito; Cliente; Crédito Direto;	Crédito Direto;	Proposta não aprovada	Taxa de juros, Cliente, Crédito, Alíquota do imposto, Proposta de contrato, Proposta de crédito	7	5	Representa a proposta de contrato, contendo: - código da proposta de crédito, - nome do cliente, - CPF, - endereço do cliente, - telefone do cliente, - lista de peças, - valor a ser financiado, - número de parcelas, - taxa de juros, - para cada parcela: - data a ser realizado o pagamento, - valor a ser pago, - juros correspondentes à multa por atraso.
50	Limite de crédito do cliente	Dado	Verificar limite de crédito do cliente; Comprometer limite de crédito;	Crédito Direto;	Crédito Direto;	-	-	2	1	Representa o valor de limite de crédito do cliente.

51	Taxa de juros do cliente	Dado	Determinar taxa de juros a ser cobrada do cliente;	Crédito Direto;	Crédito Direto;	-	-	1	1	Representa o valor da taxa de juros que foi determinada para o cliente.
52	Taxa de juros	Dado	Determinar taxa de juros a ser cobrada do cliente;	Crédito Direto;	Crédito Direto;	-	-	1	0	Representa as taxas de juros cadastradas, contendo para cada faixa de valor de financiamento: - valor total mínimo de financiamento, - valor total máximo de financiamento, - número de parcelas mínimo, - número de parcelas máximo, - taxa de juro.
53	Notificação de proposta de contrato	Dado	Analisar contrato; Gerar proposta de contrato;	Analista de crédito; Crédito Direto;	Crédito Direto;	-	-	1	1	Representa a notificação de proposta de contrato emitida por e-mail pelo sistema Crédito Direto ao Analista de crédito, após gerar a proposta de contrato.
54	Impostos sobre crédito	Dado	Calcular alíquota de imposto;	Crédito Direto;	Crédito Direto;	-	-	1	0	Representa os impostos que devem ser incluídos em contratos de crédito segundo a legislação vigente local.

Tabela 6-8 – Relatório de serviços consolidados

Índice	Serviço candidato	Grau de reuso	Múltiplas instâncias	Sistemas	Papéis	Modelo conceitual	Req. Implementado	Req. Demanda
1	Cliente novo	4	NÃO	Crédito Direto;	Crédito Direto;	-		
2	Cadastro de cliente desatualizado	4	NÃO	Crédito Direto;	Crédito Direto;	-		
3	Comprometimento do limite de crédito	2	NÃO	Crédito Direto;	Crédito Direto;	-		
4	Valor limite para taxa de juros	2	NÃO	Crédito Direto;	Crédito Direto;	-		
5	Determinação da taxa de juros	2	NÃO	Crédito Direto;	Crédito Direto;	-		
6	Valor de limite de crédito	2	NÃO	Crédito Direto;	Crédito Direto;	-		
7	Cliente novo	4	NÃO	Crédito Direto;	Crédito Direto;	-		
8	Cadastro de cliente desatualizado	4	NÃO	Crédito Direto;	Crédito Direto;	-		
9	Comunicado de não aprovação de proposta	4	NÃO	Crédito Direto;	Atendente;	-		
10	Comunicado de não aprovação de contrato de risco	2	NÃO	Crédito Direto;	Analista de crédito;	-		
11	Alteração da proposta de crédito	2	NÃO	Crédito Direto;	Analista de crédito;	-		
12	Cálculo das alíquotas de impostos	2	NÃO	Crédito Direto;	Crédito Direto;	-		
13	Norma de cálculo da alíquota de impostos	2	NÃO	Crédito Direto;	Crédito Direto;	-		
14	Incluir cliente	2	NÃO	Crédito Direto;	Crédito Direto;	-		
15	Atualizar informações do cliente	2	NÃO	Crédito Direto;	Crédito Direto;	-		
16	Consultar informações do cliente	6	NÃO	Crédito Direto;	Crédito Direto;	-		
17	Gerar proposta de contrato	2	NÃO	Crédito Direto;	Crédito Direto;	-		
18	Consultar informações do cliente	6	NÃO	Crédito Direto;	Crédito Direto;	-		
19	Comprometer limite de crédito	2	NÃO	Crédito Direto;	Crédito Direto;	-		
20	Registrar taxa de juros	2	NÃO	Crédito Direto;	Crédito Direto;	-		
21	Cancelar proposta	2	NÃO	Crédito Direto;	Crédito Direto;	-		
22	Verificar limite de crédito do cliente	2	NÃO	Crédito Direto;	Crédito Direto;	-		
23	Consultar informações do cliente	6	NÃO	Crédito Direto;	Crédito Direto;	-		
24	Consultar proposta de contrato	6	NÃO	Crédito Direto;	Atendente;	-		
25	Consultar proposta de contrato	6	NÃO	Crédito Direto;	Atendente; Cliente;	-		
26	Consultar proposta de contrato para análise	2	NÃO	Crédito Direto;	Analista de crédito;	-		
27	Analisar contrato	2	NÃO	Crédito Direto;	Analista de crédito;	-		

28	Consultar créditos concedidos	6	NÃO	Crédito Direto;	Analista de crédito;	-		
29	Cancelar contrato de risco	2	NÃO	Crédito Direto;	Atendente;	-		
30	Cancelar créditos concedidos	4	NÃO	Crédito Direto;	Atendente;	-		
31	Consultar créditos concedidos	6	NÃO	Crédito Direto;	Atendente;	-		
32	Aprovar créditos concedidos	2	NÃO	Crédito Direto;	Atendente;	-		
33	Cancelar créditos concedidos	4	NÃO	Crédito Direto;	Atendente;	-		
34	Consultar créditos concedidos	6	NÃO	Crédito Direto;	Atendente;	-		
35	Consultar créditos de contrato de risco	2	NÃO	Crédito Direto;	Analista de crédito;	-		
36	Registrar alíquota de imposto	2	NÃO	Crédito Direto;	Crédito Direto;	-		
37	Calcular alíquota de imposto	2	NÃO	Crédito Direto;	Crédito Direto;	-		
38	Loop - 1	2	NÃO	Crédito Direto;	Crédito Direto; Analista de crédito;	-		
39	Calcular alíquota de imposto	-	SIM	Crédito Direto;	Crédito Direto;	-		
40	Enviar mensagem para Calcular alíquota de imposto	-	NÃO	-	-	-		
41	Consolidar mensagens de Calcular alíquota de imposto	-	NÃO	-	-	-		
42	Analisar pedido de crédito (Recebe) Receber proposta de crédito	1	NÃO	Crédito Direto;	Crédito Direto;	-		
43	Cancelar proposta de crédito_Comunicar proposta não aprovada	2	NÃO	Crédito Direto;	Crédito Direto; Atendente;	-		
44	Comprometer limite de crédito_Calcular alíquota de imposto_Determinar taxa de juros a ser cobrada do cliente_Gerar proposta de contrato_Analisar contrato	2	NÃO	Crédito Direto;	Crédito Direto; Analista de crédito;	-		
45	Comunicar não aprovação de contrato de risco_Cancelar contrato de risco_Comunicar proposta não aprovada	2	NÃO	Crédito Direto;	Analista de crédito; Atendente;	-		
46	Proposta de crédito	6	NÃO	Crédito Direto;	Analista de crédito; Crédito Direto;	-		

47	Cadastro do cliente	6	NÃO	Crédito Direto;	Crédito Direto;	Cadastro do cliente		
48	Créditos concedidos	10	NÃO	Crédito Direto;	Atendente; Crédito Direto;	-		
49	Proposta de contrato	14	NÃO	Crédito Direto;	Atendente; Analista de crédito; Cliente; Crédito Direto;	Proposta não aprovada		
50	Limite de crédito do cliente	4	NÃO	Crédito Direto;	Crédito Direto;	-		
51	Taxa de juros do cliente	2	NÃO	Crédito Direto;	Crédito Direto;	-		
52	Taxa de juros	2	NÃO	Crédito Direto;	Crédito Direto;	-		
53	Notificação de proposta de contrato	4	NÃO	Crédito Direto;	Analista de crédito; Crédito Direto;	-		
54	Impostos sobre crédito	2	NÃO	Crédito Direto;	Crédito Direto;	-		

Tabela 6-9 – Relatório de Serviços X Atividades

<b>Serviço candidato</b>	Cadastrar cliente	Atualizar cadastro do cliente	Comprometer limite de crédito	Determinar taxa de juros a ser cobrada do cliente	Verificar limite de crédito do cliente	Verificar cadastro do cliente	Comunicar proposta não aprovada	Comunicar não aprovação de contrato de risco	Alterar proposta de crédito	Calcular alíquota de imposto	Gerar proposta de contrato	Cancelar proposta de crédito	Verificar condições de contrato com cliente	Analisar contrato	Cancelar contrato de risco	Aprovar contrato	Cancelar contrato
1 - Cliente novo	X																
2 - Cadastro de cliente desatualizado		X															
3 - Comprometimento do limite de crédito			X														
4 - Valor limite para taxa de juros				X													
5 - Determinação da taxa de juros				X													
6 - Valor de limite de crédito					X												
7 - Cliente						X											

novo																	
8 - Cadastro de cliente desatualizado						X											
9 - Comunicado de não aprovação de proposta							X										
10 - Comunicado de não aprovação de contrato de risco								X									
11 - Alteração da proposta de crédito									X								
12 - Cálculo das alíquotas de impostos										X							
13 - Norma de cálculo da alíquota de impostos										X							
14 - Incluir cliente	X																
15 - Atualizar informações do cliente		X															
16 - Consultar informações do cliente		X															
17 - Gerar proposta de contrato											X						
18 - Consultar informações do cliente											X						
19 - Comprometer limite de crédito			X														



20 - Registrar taxa de juros				X													
21 - Cancelar proposta											X						
22 - Verificar limite de crédito do cliente					X												
23 - Consultar informações do cliente						X											
24 - Consultar proposta de contrato							X										
25 - Consultar proposta de contrato												X					
26 - Consultar proposta de contrato para análise													X				
27 - Analisar contrato													X				
28 - Consultar créditos concedidos													X				
29 - Cancelar contrato de risco														X			
30 - Cancelar créditos concedidos														X			
31 - Consultar créditos concedidos															X		
32 - Aprovar créditos concedidos															X		
33 - Cancelar créditos																X	

concedidos																	
34 - Consultar créditos concedidos																	X
35 - Consultar créditos de contrato de risco									X								
36 - Registrar alíquota de imposto										X							
37 - Calcular alíquota de imposto										X							
38 - Loop - 1			X	X					X	X	X			X			
39 - Calcular alíquota de imposto										X							
40 - Enviar mensagem para Calcular alíquota de imposto																	
41 - Consolidar mensagens de Calcular alíquota de imposto																	
42 - Analisar pedido de crédito (Recebe) Receber proposta de crédito						X											
43 - Cancelar proposta de crédito_Comunicar proposta não aprovada							X					X					
44 -			X	X						X	X			X			

Comprometer limite de crédito_Calcular alíquota de imposto_Determinar taxa de juros a ser cobrada do cliente_Gerar proposta de contrato_Analisar contrato																	
45 - Comunicar não aprovação de contrato de risco_Cancelar contrato de risco_Comunicar proposta não aprovada							X	X							X		
46 - Proposta de crédito						X			X			X					
47 - Cadastro do cliente	X	X				X											
48 - Créditos concedidos			X												X	X	X
49 - Proposta de contrato											X		X	X	X	X	X
50 - Limite de crédito do cliente			X		X												
51 - Taxa de juros do cliente				X													
52 - Taxa de juros				X													
53 - Notificação de proposta de contrato										X			X				
54 - Impostos sobre crédito										X							

Tabela 6-10 – Relatório de Serviços x Papéis

Serviço candidato	Crédito Direto	Atendente	Analista de crédito	Cliente
1 - Cliente novo	X			
2 - Cadastro de cliente desatualizado	X			
3 - Comprometimento do limite de crédito	X			
4 - Valor limite para taxa de juros	X			
5 - Determinação da taxa de juros	X			
6 - Valor de limite de crédito	X			
7 - Cliente novo	X			
8 - Cadastro de cliente desatualizado	X			
9 - Comunicado de não aprovação de proposta		X		
10 - Comunicado de não aprovação de contrato de risco			X	
11 - Alteração da proposta de crédito			X	
12 - Cálculo das alíquotas de impostos	X			
13 - Norma de cálculo da alíquota de impostos	X			
14 - Incluir cliente	X			
15 - Atualizar informações do cliente	X			
16 - Consultar informações do cliente	X			
17 - Gerar proposta de contrato	X			
18 - Consultar informações do cliente	X			
19 - Comprometer limite de crédito	X			
20 - Registrar taxa de juros	X			
21 - Cancelar proposta	X			
22 - Verificar limite de crédito do cliente	X			
23 - Consultar informações do cliente	X			
24 - Consultar proposta de contrato		X		
25 - Consultar proposta de contrato		X		X
26 - Consultar proposta de contrato para análise			X	
27 - Analisar contrato			X	
28 - Consultar créditos concedidos			X	
29 - Cancelar contrato de risco		X		
30 - Cancelar créditos concedidos		X		
31 - Consultar créditos concedidos		X		
32 - Aprovar créditos concedidos		X		
33 - Cancelar créditos concedidos		X		
34 - Consultar créditos concedidos		X		
35 - Consultar créditos de contrato de risco			X	
36 - Registrar alíquota de imposto	X			
37 - Calcular alíquota de imposto	X			
38 - Loop - 1	X		X	
39 - Calcular alíquota de imposto	X			
40 - Enviar mensagem para Calcular alíquota de imposto				

41 - Consolidar mensagens de Calcular alíquota de imposto				
42 - Analisar pedido de crédito (Recebe) Receber proposta de crédito	X			
43 - Cancelar proposta de crédito_Comunicar proposta não aprovada	X	X		
44 - Comprometer limite de crédito_Calcular alíquota de imposto_Determinar taxa de juros a ser cobrada do cliente_Gerar proposta de contrato_Analisar contrato	X		X	
45 - Comunicar não aprovação de contrato de risco_Cancelar contrato de risco_Comunicar proposta não aprovada		X	X	
46 - Proposta de crédito	X		X	
47 - Cadastro do cliente	X			
48 - Créditos concedidos	X	X		
49 - Proposta de contrato	X	X	X	X
50 - Limite de crédito do cliente	X			
51 - Taxa de juros do cliente	X			
52 - Taxa de juros	X			
53 - Notificação de proposta de contrato	X		X	
54 - Impostos sobre crédito	X			

Tabela 6-11 – Tabela de consolidação de serviços de fluxo

Índice	Serviço candidato	Nº atividades	Ocorrência em modelos	Nº entidades	Nº raiais
38	Loop - 1	6	Analisar pedido de crédito; Analisar pedido de crédito;	1	2
42	Analisar pedido de crédito (Recebe) Receber proposta de crédito	1		-	1
43	Cancelar proposta de crédito_Comunicar proposta não aprovada	2	Analisar pedido de crédito; Analisar pedido de crédito;	-	2
44	Comprometer limite de crédito_Calcular alíquota de imposto_Determinar taxa de juros a ser cobrada do cliente_Gerar proposta de contrato_Analisar contrato	5	Analisar pedido de crédito; Analisar pedido de crédito;	1	2
45	Comunicar não aprovação de contrato de risco_Cancelar contrato de risco_Comunicar proposta não aprovada	3	Analisar pedido de crédito; Analisar pedido de crédito;	-	2

## Capítulo 7: CONCLUSÃO

---

A Arquitetura Orientada a Serviços é um assunto bastante discutido atualmente e muitas empresas estão adotando este paradigma devido aos benefícios que podem ser alcançados. Entretanto, sua complexidade de implantação é uma grande barreira para muitas organizações que, algumas vezes, não alcançam o sucesso ao tentar implantar SOA. Isso abre espaço para pesquisas que desenvolvam produtos que auxiliem aos analistas a reduzir os riscos e o alto investimento envolvidos neste projeto.

A modelagem de processos de negócio pode ser um ponto de partida para um projeto de implantação de SOA. Durante o levantamento de informações sobre os processos, os analistas poderão aprender mais sobre o negócio, além de registrar formalmente os processos. Esse conhecimento prévio auxiliará futuramente no desenvolvimento das aplicações, reduzindo os riscos de interpretações incorretas sobre requisitos do negócio. Além disso, os modelos de processos podem ser utilizados como insumos para a identificação de serviços.

Neste trabalho foi descrita a extensão e automatização do método para identificação de serviços candidatos proposto originalmente em [AZEVEDO *et al.*, 2009a, 2009b]. Esta abordagem propõe heurísticas que tomam como base informações contidas em modelos de processos de negócios e disponibilizam um conjunto de atributos dos serviços úteis para os projetistas tomarem decisões quanto à análise, projeto e implementação dos mesmos.

A partir da automatização proposta foi realizado um estudo de caso no modelo de processo descrito no anexo II, cujos resultados são apresentados no capítulo 6. Isto demonstrou a eficácia do método proposto no que se refere à rapidez, organização, padronização, redução de erros e facilidade de identificar serviços candidatos.

Concluiu-se também que, em grandes repositórios de modelos de processos, a automatização é fundamental. Pode-se afirmar que a verificação manual torna-se inviável à medida que se torna maior a complexidade de um modelo de negócio e/ou quando se deseja analisar vários processos ao mesmo tempo.

A abordagem vem sendo ampliada para contemplar outras heurísticas relacionadas com as fases de projeto e implementação de serviços. As heurísticas apresentadas neste trabalho que ainda não foram implementadas também deverão ser abordadas futuramente. Além disso, pretende-se realizar novos estudos de caso em diferentes domínios, para que seja possível generalizar os resultados obtidos até agora.

## REFERÊNCIAS BIBLIOGRÁFICAS

---

ARAÚJO, R., **Gestão de Processos de Negócio**, Notas de aula, 2008.

ARIS, 2006, *Help Documentation*, ARIS Business Architect 7.0 v. 7.0.2.234414, IDS Scheer AG.

AZEVEDO, L., BAIÃO, F., SANTORO, F., SOUZA, J., REVOREDO, K., PEREIRA, V., HERLAIN, I. **Identificação de serviços a partir da modelagem de processos de negócio**. Simpósio Brasileiro de Sistemas de Informação (SBSI'09), Brasília, 2009a.

AZEVEDO, L., SANTORO, F., BAIÃO, F., SOUZA, J., REVOREDO, K., PEREIRA, V., HERLAIN, I. *A Method for Service Identification from Business Process Models in a SOA Approach*. 10th International Workshop on Business Process Modeling, Development, and Support (BPMDS), 2009b, Amsterdam. *Enterprise, Business-Process, and Information Systems Modelling*. v. 29.

AZEVEDO, L.; PEREIRA, V.; REVOREDO, K.; SOUZA, J.; SANTORO, F.; BAIÃO, F.; SOUSA, H. P. **Metodologia de identificação de serviços a partir da modelagem de processos de negócio**. Relatórios Técnicos do DIA/UNIRIO (RelaTe-DIA), RT-0021/2009, 2009c. Disponível (também) em: <http://seer.unirio.br/index.php/monografiasppgi>.

AZEVEDO, L.; SOUSA, H. P.; SOUZA, J. F., SANTORO, F. BAIÃO, F.; **Identificação automática de serviços candidatos a partir de modelos de processos de negócio**. Conferência IADIS Ibero Americana WWW/INTERNET 2009 (CIAWI'09), Outubro, 21-23, Madrid, Espanha, 2009d.

BASTOS, A. L. S., CAMAREIRA, R. F., **Ferramentas de Apoio à Engenharia de Processos de Negócios: Critérios de Classificação e Método de Análise de Adequação a um Projeto**, 2000, disponível em [http://www.abepro.org.br/biblioteca/ENEGEP2000\\_E0075.PDF](http://www.abepro.org.br/biblioteca/ENEGEP2000_E0075.PDF).

BIEBERSTEIN, N., BOSE, S., FIAMMANTE, M., JONES, K., SHAH, R.; *Service Oriented Architecture (SOA) Compass: Business Value, Planning and Enterprise Roadmap*. NJ, EUA: IBM, 2006, 272 p.

Bibliografia: ISBN-10: 0-13-187002-5 / ISBN-13: 978-0-13-187002-4

BLOOMBERGSCHMELZER, BLOOMBERG, J. , RONALD, S., *Service Orient or Be Doomed!: How Service Orientation Will Change Your Business*. Hoboken; New Jersey: John Wiley & Sons, 2006, 249 p.

Bibliografia: ISBN-10, 0-471-76858-8 / ISBN-13, 978-0471-76858-6.



BPM-ADVISOR, 2009 - **Padrões e Notações**, disponível em <http://www.bpm-advisor.com.br/padnotac.htm>. Acessado em 6 de janeiro de 2010, às 20:00.

COMPUTERWORLD, 2007 **Governança de SOA fracassa na maior parte das empresas, diz estudo**, disponível em: <http://computerworld.uol.com.br/gestao/2007/10/23/idgnoticia.2007-10-23.7690613147/> ; Acessado em 30 de novembro de 2009 às 18:00.

CRUZ, T. **Sistemas, métodos & processos: administrando organizações por meio de processos de negócios**. São Paulo: Editora Atlas, 2003, 274 p.  
Bibliografia: ISBN, 85-224-3329-1.

DAVIS, R., *Business Process Modeling with ARIS – A Practical Guide*. London: Springer, 2002, 531 p.  
Bibliografia: ISBN, 1-85233-434-7.

ERL, T., *Service-Oriented Architecture: concepts, technology, and Design*, Prentice Hall, Crawfordsville: Indiana, 2005, 792 p.  
Bibliografia: ISBN, 0-13-18-5858-0.

FURTADO, C.; PEREIRA, V.; AZEVEDO, L.; BAIÃO, F.; SANTORO, F.; MAGDALENO, A.; CAPELLI, C.; NUNES, V.; **Arquitetura Orientada a Serviço - Conceituação**. Relatórios Técnicos do DIA/UNIRIO (RelaTe-DIA), RT-0012/2009, 2009. Disponível (também) em: <http://seer.unirio.br/index.php/monografiasppgi>.

GARTNER, *Predicts 2006: The Strategic Impact of SOA Broadens*. Disponível em < <http://www.gartner.com> >, 2006.

GONÇALVES, J. E. L., **As empresas são grandes coleções de processos**  
RAE - Revista de Administração de Empresas. Rio de Janeiro, v.40, n.1, p. 6-19, Jan./Mar. 2000.

GONÇALVES, J. E. L., **Processo, que processo?**  
RAE - Revista de Administração de Empresas. Rio de Janeiro, v.40, n.1, p 8-19 Out./Dez. 2000.

GRAHAM, M., LEBARON, M., *The horizontal revolution: Reengineering Your Organization Through Teams*. San Francisco: Jossey-Bass, 1994.

GU, Q., LAGO, P., *A stakeholder-driven service life cycle model for SOA*. In *2nd international Workshop on Service Oriented Software Engineering: in Conjunction with the 6th ESEC/FSE Joint Meeting*, pp 1-7, Croatia, 2007.

*GUIDE BUSINESS RULE PROJECT*, 2008.  
[http://www.businessrulesgroup.org/first\\_paper/br01c1.htm#s1e](http://www.businessrulesgroup.org/first_paper/br01c1.htm#s1e) (Acessado em janeiro de 2010)

HURWITZ, J., BLOOR, R., BAROUDI, C., KAUFMAN, M.; *Service Oriented Architecture for Dummies*, Indianapolis; Indiana: Wiley, 2007, 343 p.  
Bibliografia: ISBN-13: 978-0-470-05435-2 / ISBN-10: 0-470-05435-2

JOSUTTIS, N. M., *SOA in Practice – The Art of Distributed System Design*, Beijing; Cambridge; Farnham; Köln; Paris; Sebastopol; Taipei; Tokyo: O'Reilly, 2007, 324 p.  
Bibliografia: ISBN-10: 0-596-52955-4 / ISBN-13: 978-0-596-52955-0.

JUNIOR, A. C. B. **Roteiro para a definição de uma arquitetura SOA utilizando BPM**. Dissertação de MBA. – Escola Politécnica, Universidade de São Paulo. São Paulo: 2007.

KLÜCKMANN, J., *On the way to SOA. ARIS Expert Paper*. Disponível em < [http://www.ids-scheer.com/en/ARIS/ARIS\\_Solutions/Business-Driven\\_SOA\\_Management/140099.html](http://www.ids-scheer.com/en/ARIS/ARIS_Solutions/Business-Driven_SOA_Management/140099.html) >, acessado em 19 de janeiro de 2010, 2006

KLÜCKMANN, J., *10 Steps to Business-Driven SOA, ARIS Expert*. Disponível em < [http://www.ids-scheer.com/en/ARIS/ARIS\\_Solutions/Business-Driven\\_SOA\\_Management/140099.html](http://www.ids-scheer.com/en/ARIS/ARIS_Solutions/Business-Driven_SOA_Management/140099.html) >, acessado em 19 de janeiro de 2010, 2007.

KOHLBORN, T., KORTHAUS, A., CHAN, T., ROSEMMAN, M.. *Identification and Analysis of Business and Software Services – A Consolidated Approach, IEEE Transactions on Service Computing*, Vol 2, No 1, 2009.

MACHADO, J. C. **Um estudo sobre o desenvolvimento orientado a serviços**. Dissertação de Mestrado. – Departamento de Informática, Programa de Pós-Graduação em Informática, Pontifícia Universidade Católica – PUC - Rio. Rio de Janeiro: 2004, disponível em [http://www2.dbd.puc-rio.br/pergamum/tesesabertas/0210486\\_04\\_Indice.html](http://www2.dbd.puc-rio.br/pergamum/tesesabertas/0210486_04_Indice.html).

MARKS, E. A.; BELL, M., *Service-Oriented Architecture: a planning and implementation guide for business and technology*, John Willey & Sons Inc., 2006.

MOZILLA, *Core Java Script Guide 1.5*, 2000, disponível em [https://developer.mozilla.org/en/Core\\_JavaScript\\_1.5\\_Guide](https://developer.mozilla.org/en/Core_JavaScript_1.5_Guide). Acessado em 14 de janeiro de 2010, às 03:00.

NETO, R. U. , **Arquitetura Orientada a Serviços – SOA Infraestrutura para a Inovação**, Apresentação em mídia eletrônica, 2006, disponível em <http://www.pr.senai.br/posgraduacao/uploadAddress/Introducao%20ao%20SOA%5B31574%5D.pdf>.

OASIS, **Modelo de Referência para Arquitetura Orientada a Serviço 1.0**, disponível em <http://www.pcs.usp.br/~pcs5002/oasis/soa-rm-csbr.pdf>, OASIS, 2006.

OMG, **Business Process Modeling Notation, V1.1**, disponível em <http://www.omg.org/spec/BPMN/1.1/PDF/>, Janeiro de 2008.

PAPAZOGLU, MIKE P.; HEUVEL, WILLEM-JAN, **Service oriented architectures: approaches, technologies and research issues**, VLDB Journal, Springer-Verlag, 2007.

PETROBRAS, **Diretrizes para modelagem de processos na E&P - versão 8.1.** – Petrobras – Exploração e Produção (E&P). Rio de Janeiro, v.8.1, Mai. 2008, 100 p..

RUSSELL, N., TER HOFSTEDE, A., EDMOND, D., VAN DER AALST, W., **Workflow Data Patterns**. QUT Technical report, FIT-TR-2004-01, Queensland University of Technology, Brisbane, 2004.

SCHEER, A.W. **ARIS – Business Process Modeling**. Berlin; Heidelberg; New York; Barcelona; Hong Kong; London; Milan; Paris; Singapore; Tokyo: Springer, 2000, 218 p. 3. ed.  
Bibliografia: ISBN, 3-540-65835-1.

SHARP, A.; MCDERMOTT, P.; **Workflow Modeling: Tools for Process Improvement and Application Development**. Boston; London: Artech House, 2009, 449 p. 2. ed.  
Bibliografia: ISBN, 978-1-59693-192-3.

SILVA, L. M., **Aplicando Composição e Orquestração de Serviços na Organização de Sistemas**, Monografia de Graduação. – Gerência Educacional de Tecnologia da Informação, Centro Federal de Educação Tecnológica do Rio Grande do Norte. Natal: 2007.

SOA CONSORTIUM, Disponível em [www.soa-consortium.org](http://www.soa-consortium.org), acessado em 19 de janeiro de 2010, 2008a.

SOA CONSORTIUM, **Valero Energy: Mergers and Acquisitions: Bringing Acquisitions Online Faster**. Disponível em <[http://www.soa-consortium.org/cs/Business\\_Lead/valero-energy.htm](http://www.soa-consortium.org/cs/Business_Lead/valero-energy.htm)>, acessado em 19 de janeiro de 2010, 2008b.

SOA CONSORTIUM, **Automobile #1: SOA Governance Increasing Agility Through a Governance Focus**. Disponível em <[http://www.soa-consortium.org/cs/Business\\_Lead/automobile.htm](http://www.soa-consortium.org/cs/Business_Lead/automobile.htm)>, acessado em 19 de janeiro de 2010, 2008c.

STANDISHGROUP, **The CHAOS Report**, disponível em <http://www.standishgroup.com/>, 1994-2004, acessado em 19 de janeiro de 2010.

THOM, L.H, IOCHPE, C., CHIAO, C., **Padrões de Workflow para Reuso em Modelagem de Processo de Negócio**, Lucinéia Heloisa Thom, Cirano Iochpe, Carolina Chiao, 2007.

VAN DER AALST, W., TER HOFSTEDE, A., KIEPUZEWSKI, B., BARROS, A., *Advanced workflow patterns*. In *Proceedings of the 7th International Conference on Cooperative Information Systems (CoopIS 2000)*, LNCS 1901, pp. 18–29, 2000.

VAN DER AALST, W., TER HOFSTEDE, A., KIEPUZEWSKI, B., BARROS, A., *Workflow patterns*. In *Distributed and Parallel Databases* 14(1), p.5–51, 2003.

VALERO ENERGY. Disponível em <<http://www.valero.com/>>, acessado em 10 de junho de 2008.


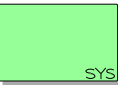
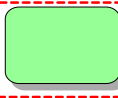
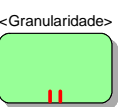
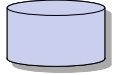



WEBMETHODS, *SOA Governance - Enabling Sustainable Success with SOA*, white paper, p.20, 2006.

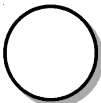
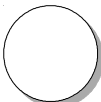
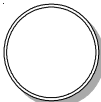
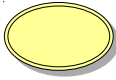

WESKE, M., *Business Process Management – Concepts, Languages, Architectures*, Verlag; Berlin; Heidelberg: Springer, 2007, 368 p.  
Bibliografia: ISBN, 978-3-540-73521-2.


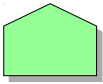



## ANEXO I – TABELA DE NOTAÇÃO DE SÍMBOLOS PARA MODELAGEM DE PROCESSOS

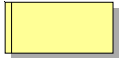







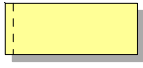
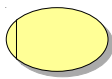
A Tabela 1 – Anexo I ilustra o conjunto de elementos que compõem a notação utilizada na modelagem dos diagramas de processos de negócio apresentados neste trabalho.

Tabela 1 – Anexo I – Notação para modelagem de processos de negócio [PETROBRAS, 2008]

Nome	Semântica	Sintaxe
<b>Atividade</b> ( <i>process</i> )	Representa um processo ou uma etapa de uma sequencia que precisa ser executada para que um processo seja realizado.	
<b>Atividade automatizada (sistêmica)</b> ( <i>system function</i> )	Representa uma atividade executada automaticamente e exclusivamente por um sistema, sem interferência de uma pessoa. Pessoas, equipamentos, sensores, outros sistemas, etc., podem interagir com esta atividade apenas como disparadores ou receptores de resultados.	
<b>Atividade Crítica</b> ( <i>process</i> )	Representa uma atividade definida por padrão rígido de execução por oferecer riscos a pessoas, materiais e/ou ambiente internamente ou externamente à organização, impacto no resultado, complexidade e/ou alta frequência de execução.	
<b>Atividade de múltiplas instâncias</b> ( <i>multiple instances</i> )	Representa uma atividade que pode ser executada várias vezes em um processo, de acordo com a sua granularidade. A granularidade determina o critério de repetição da atividade. Cada instância é executada paralelamente, seguindo, de forma independente uma da outra, o seu curso. Após a execução das várias instâncias da atividade o processo segue o fluxo normal.	
<b>Base de dados</b> ( <i>database</i> )	Representa informações ou dados que estão armazenadas em banco de dados de sistemas de aplicações.	
<b>Dados / Informação</b> ( <i>cluster</i> )	Representa um conjunto de informações (estruturadas ou não) gerado ou consumido durante a execução do processo.	
<b>Documento</b> ( <i>document</i> )	Representa uma informação (documento, relatório, planilha, etc..) impresso em papel, utilizada como apoio durante a realização atividades do processo. No caso de diagramas de FAD de processos informações de entrada e saída não são representadas neste diagrama.	
<b>Documento eletrônico</b> ( <i>electronic document</i> )	Representa uma informação (documento, relatório, planilha, etc..) disponibilizada em meio eletrônico, utilizada como apoio durante a realização de atividades do processo. No caso de diagramas de FAD de Processo informações de entrada e saída não são representadas neste diagrama.	

Nome	Semântica	Sintaxe
<b>Evento Final</b> <i>(end event)</i>	<p>Representa a circunstância ou status final do processo. Um processo pode ter mais de um evento final.</p> <p>Eventos representam não somente acontecimentos ocorridos no ponto onde aparecem no processo, mas também acontecimentos ou situações que já são conhecidos por ele.</p> <p>Deve ser utilizado para representar que um ramo do processo está finalizado (e não necessariamente o processo como um todo), ou seja, que o caminho corrente não possui uma continuidade.</p>	
<b>Evento inicial</b> <i>(start event)</i>	<p>Representa uma circunstância ou status que propicia o início do processo. Um processo pode ter mais de um evento inicial.</p> <p>Quando o processo for antecedido por outro processo na cadeia de valor, o fluxo deverá ter início com um objeto “Interface de Processos” seguido do evento inicial (e de mesmo nome do evento final anterior).</p>	
<b>Evento intermediário</b> <i>(intermediate event)</i>	<p>Representa uma circunstância ou status relevante para o entendimento do processo.</p> <p>Eventos representam não somente acontecimentos ocorridos no ponto onde aparecem no processo, mas também acontecimentos ou situações que já são conhecidos por ele.</p> <p>Deve ser utilizado:</p> <ul style="list-style-type: none"> <li>- Quando preceder ou anteceder uma interface de processo (quando não for o caso de usar eventos inicial e final)</li> <li>- Após os operadores lógicos “Ou exclusivo” e “Ou” quando estes tiverem associadas duas ou mais saídas a eles, identificando o status exigido para a realização de cada caminho disponível;</li> <li>- Antes dos operadores lógicos “Ou exclusivo” e “Ou” quando estes tiverem associadas duas ou mais entradas a eles, identificando o status necessário para a unificação de dois ou mais caminhos do fluxo do processo;</li> <li>- Como entrada e saída de uma atividade detalhada em outro diagrama de processos;</li> </ul>	
<b>Grupo</b> <i>(group)</i>	<p>Representa um grupo, comitê ou célula formado por postos de trabalho que atuam em conjunto para executar uma ou mais atividades do processo. A composição do grupo deve ser explicitada através de um modelo de organograma.</p>	
<b>Indicador de desempenho</b> <i>(key performance indicator instance)</i>	<p>Representa um indicador de desempenho para avaliar o processo, ou seja, forma de medir e analisar o cumprimento dos objetivos previamente traçados pelo planejamento estratégico, através de uma sistemática de avaliação.</p>	

Nome	Semântica	Sintaxe
Interface de Processo (process interface)	<p>Representa a interface entre processos (existentes em algum VAC), indicando que existe uma comunicação entre eles. Em geral, trata-se de uma indicação de outro processo que complementa o fluxo modelado, mas não é objeto principal do modelo em questão.</p> <p>Antes da interface de processo deve-se posicionar um evento final de mesmo nome do evento inicial do processo sucessor se esta interface estiver finalizando o processo.</p> <p>Caso se trate de uma interface para outro processo que retornará para o processo que o chama, utilize o evento intermediário antes e depois da interface de processo.</p> <p>No processo conectado a ele deve existir uma interface de processo para o processo que o chamou, anterior ao evento inicial deste e posterior ao final.</p>	
Objetivo (objective)	Representa o que se deseja alcançar com a execução de um processo de negócio. Deve ser associado ao processo correspondente.	
Operador lógico E (AND)	<p>Operador lógico que representa:</p> <ul style="list-style-type: none"> <li>- quando dividir o fluxo: que todos os caminhos precisam ser percorridos, ou seja, que todos os eventos destino devem ocorrer. A execução de cada caminho é independente da execução dos demais.</li> <li>- quando unir o fluxo: que deve haver sincronismo entre todos os caminhos que antecedem ao operador lógico, ou seja, todos os caminhos devem ser percorridos antes de iniciar a atividade/processo seguinte.</li> </ul>	
Operador lógico Ou (AND/OR)	<p>Operador lógico que representa:</p> <ul style="list-style-type: none"> <li>- quando dividir o fluxo: que pelo menos um dos caminhos precisa ser percorrido, ou seja, no mínimo um dos eventos destino deve ocorrer (podendo ser mais de um ao mesmo tempo).</li> <li>- quando unir o fluxo: que um ou mais caminhos percorridos inicia a atividade/ processo seguinte, ou seja, um ou mais eventos origem podem ocorrer.</li> </ul>	
Operador lógico Ou exclusivo (XOR)	<p>Operador lógico que representa:</p> <ul style="list-style-type: none"> <li>- quando dividir o fluxo: que apenas um dos caminhos deve ser percorrido, ou seja, apenas um dos eventos destino deve ocorrer.</li> <li>- quando unir o fluxo: que apenas um dos caminhos percorridos inicia a atividade ou processo seguinte, ou seja, apenas um dos eventos origem deve ocorrer.</li> </ul>	

Nome	Semântica	Sintaxe
<b>Posto de trabalho</b> <i>(position)</i>	<p>Representa o posto de trabalho (papel/função) que interage com um processo (produzindo ou consumindo informações).</p> <p>Um posto de trabalho não deve ser utilizado para representar o cargo ou competência de uma pessoa e sim sua função na organização.</p> <p>Postos de trabalho podem ser formais (reconhecidos e registrados formalmente pela Petrobras) ou informais (reconhecidos somente internamente uma gerência formal)</p>	
<b>Processo</b> <i>(value-added chain)</i>	<p>Representa um processo intermediário ou final de uma sequência de processos.</p>	
<b>Processo inicial</b> <i>(value-added chain)</i>	<p>Representa um macroprocesso ou processo inicial de uma sequência da cadeia de valor. Pode representar também um processo superior, a partir do qual existe uma sequência de processos a ele subordinados (ou outra cadeia de valor).</p>	
<b>Regra de negócio</b> <i>(business rule)</i>	<p>Representa uma diretiva destinada a influenciar ou guiar o comportamento do negócio, como suporte à política de negócio que é formulada em resposta a uma oportunidade.</p>	
<b>Requisito de negócio</b> <i>(business requirement)</i>	<p>Representa requerimentos provenientes do negócio que irão definir ou restringir aspectos dos sistemas de informação.</p>	
<b>Risco</b> <i>(risk)</i>	<p>Representa um risco apresentado na execução de um processo. Ou seja, algo que ocorrendo impactará na execução da atividade.</p>	
<b>Sistema de aplicação</b> <i>(application system type)</i>	<p>Representa um sistema de informação que apoia a execução ou executa uma ou mais atividades do processo.</p>	
<b>Termo técnico</b> <i>(technical term)</i>	<p>Representa o significado de um termo ou uma sigla necessária para o entendimento da atividade e do processo.</p>	
<b>Tipo de posto de trabalho</b> <i>(position type)</i>	<p>Representa um conjunto de postos de trabalho que possuem características em comum do ponto de vista de suas responsabilidades e atribuições com relação a um processo.</p>	
<b>Unidade organizacional</b> <i>(organizational unit)</i>	<p>Representa uma área (unidade de negócio, gerência, coordenação ou departamento) (formal ou informal) da organização, que interage com algum processo.</p>	



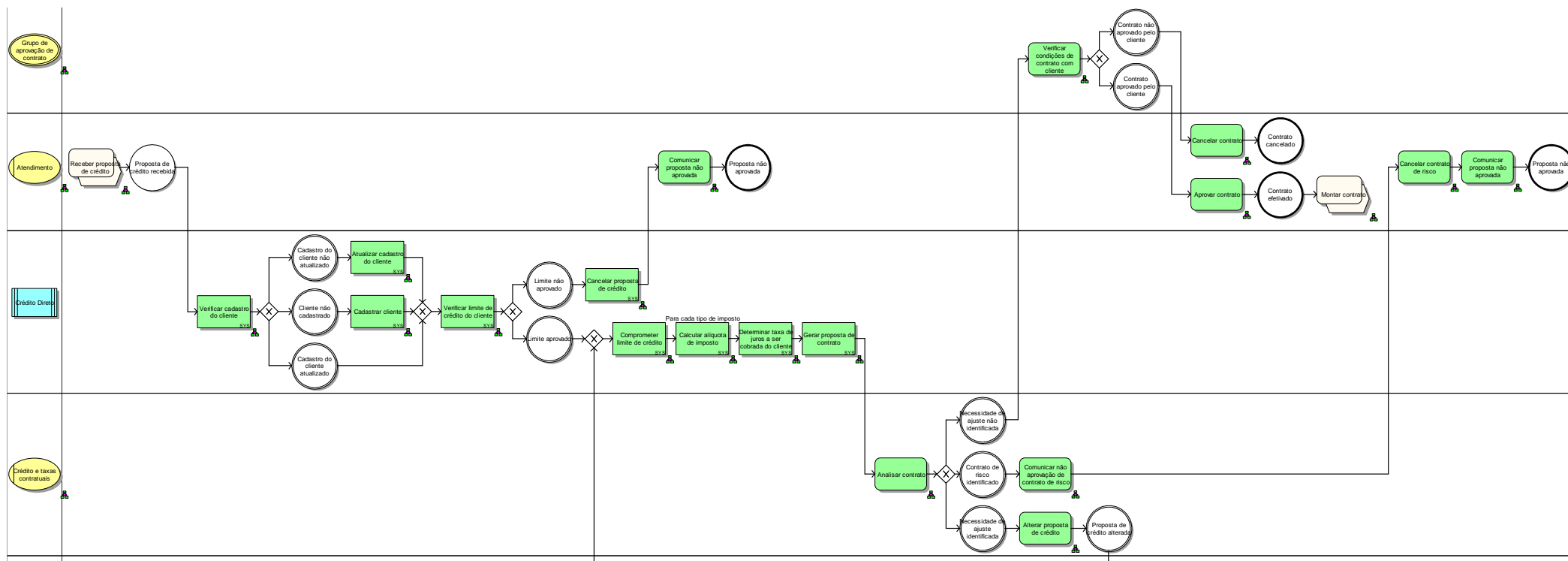
## **ANEXO II – PROCESSO ANALISAR PEDIDO DE CRÉDITO**

---

Este anexo detalha o processo de negócio “Analisar Pedido de Crédito” que é utilizado no estudo de caso apresentado neste trabalho como insumo para a execução do *script* de identificação de serviços candidatos.

### **2.1 Analisar pedido de crédito**



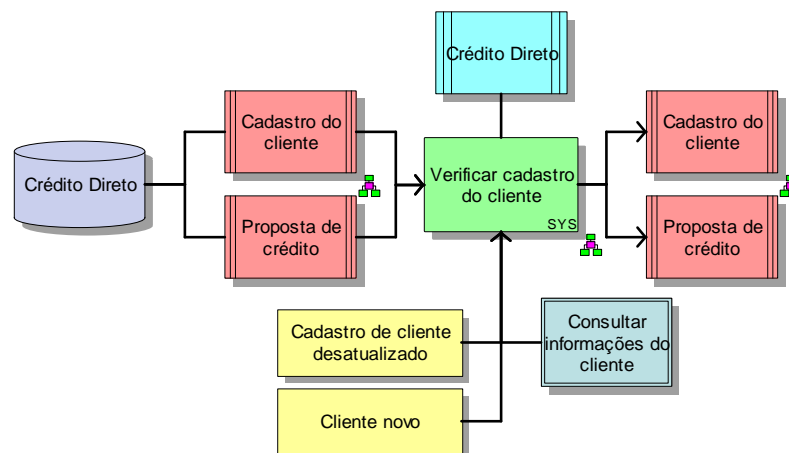


## 2.2 Verificar cadastro do cliente

O sistema Crédito Direto verifica se o cadastro do cliente está desatualizado baseado nas informações contidas na proposta de crédito. Caso o cadastro não seja encontrado, é porque o cliente é novo e deve ser cadastrado.

As informações necessárias são: cadastro do cliente e proposta de crédito.

As informações geradas são: cadastro do cliente e proposta de crédito.



Regra de Negócio	Descrição
Cadastro de cliente desatualizado	O cadastro do cliente está desatualizado se o telefone, endereço e renda informados na proposta de crédito forem diferentes das informações do cliente existentes na base de dados. A comparação entre proposta de crédito e cadastro do cliente deve ser feita pelo CPF.
Cliente novo	Um cliente é novo se não existir cadastro de cliente com mesmo CPF que o informado na proposta de crédito.

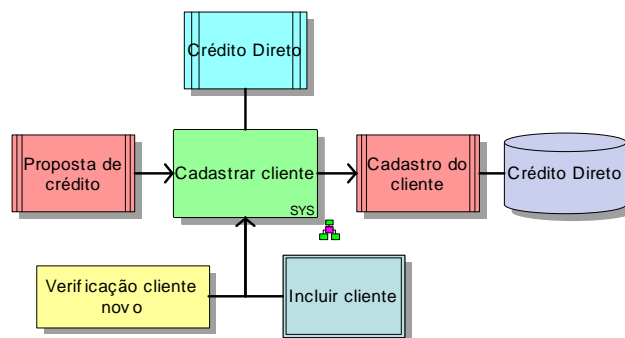
Requisito de Negócio	Descrição
Consultar informações do cliente	<p>O sistema Crédito Direto deve consultar as seguintes informações do cliente:</p> <ul style="list-style-type: none"> <li>- nome,</li> <li>- CPF,</li> <li>- identidade,</li> <li>- telefone,</li> <li>- endereço,</li> <li>- renda,</li> <li>- data da última atualização,</li> <li>- data de cadastro.</li> </ul>

## 2.3 Cadastrar cliente

O sistema Crédito Direto cadastra o cliente utilizando as informações contidas na proposta de crédito.

As informações necessárias são: proposta de crédito.

A informação gerada é: cadastro do cliente.



Regra de Negócio		Descrição
Verificação cliente novo	cliente	Um cliente é novo se não existir cadastro de cliente com mesmo CPF que o informado na proposta de crédito.

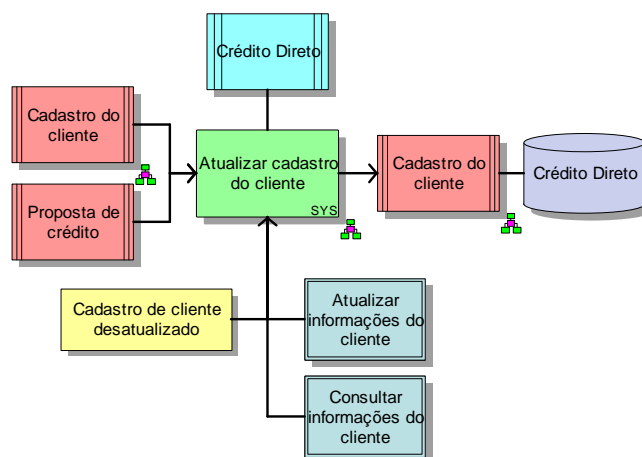
Requisito de Negócio	Descrição
Incluir cliente	<p>O sistema Crédito Direto deve criar cadastro do cliente se o cliente for novo (de acordo com a regra de negócio Verificação cliente novo), contendo as seguintes informações:</p> <ul style="list-style-type: none"> <li>- nome,</li> <li>- telefone,</li> <li>- endereço,</li> <li>- renda,</li> <li>- data de cadastro.</li> </ul>

## 2.4 Atualizar cadastro do cliente

O sistema Crédito Direto atualiza o cadastro do cliente utilizando as informações contidas na proposta de crédito.

As informações necessárias são: cadastro do cliente e proposta de crédito.

A informação gerada é: cadastro do cliente (atualizado).

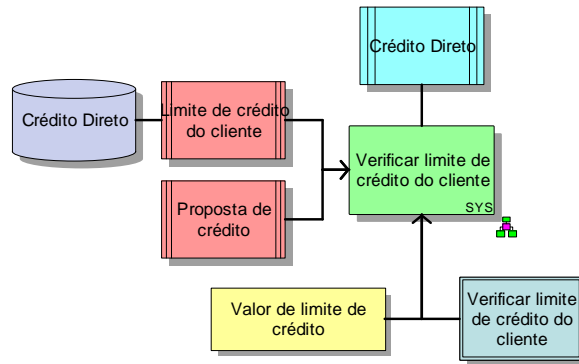


Regra de Negócio	Descrição
Verificação cadastro cliente desatualizado	O cadastro do cliente está desatualizado se o telefone, endereço e renda informados na proposta de crédito forem diferentes das informações do cliente existentes na base de dados. A comparação entre proposta de crédito e cadastro do cliente deve ser feita pelo CPF.

Requisito de Negócio	Descrição
Atualizar informações do cliente	O sistema Crédito Direto deve atualizar as seguintes informações do cliente (de acordo com a regra de negócio Verificação cadastro cliente desatualizado): <ul style="list-style-type: none"> <li>- telefone,</li> <li>- endereço,</li> <li>- renda.</li> </ul>
Consultar informações do cliente	O sistema Crédito Direto deve consultar as seguintes informações do cliente: <ul style="list-style-type: none"> <li>- nome,</li> <li>- CPF,</li> <li>- identidade,</li> <li>- telefone,</li> <li>- endereço,</li> <li>- renda,</li> <li>- data da última atualização,</li> <li>- data de cadastro.</li> </ul>

## 2.5 Verificar limite de crédito do cliente

O sistema Crédito Direto verifica o limite de crédito do cliente.  
As informações necessárias são: limite de crédito do cliente e proposta de crédito.



Regra de Negócio	Descrição
Valor de limite de crédito	O limite de crédito do cliente é igual a 20% da sua renda menos o valor mensal referentes às parcelas ainda em aberto dos créditos anteriormente concedidos ao cliente.

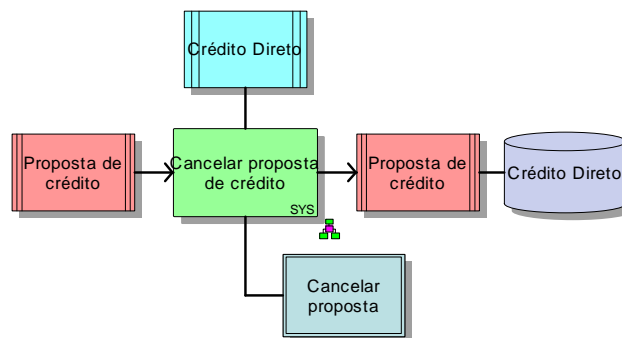
Requisito de Negócio	Descrição
Verificar limite de crédito do cliente	O sistema crédito direto deve verificar o limite de crédito do cliente de acordo com a regra de negócio Verificação de limite de crédito.

## 2.6 Cancelar proposta de crédito

O sistema Crédito Direto cancela a proposta de crédito.

A informação necessária é: proposta de crédito.

A informação gerada é: proposta de crédito (na situação reprovado).



Requisito de Negócio	Descrição
Cancelar proposta	O sistema Crédito Direto deve cancelar a proposta do cliente.

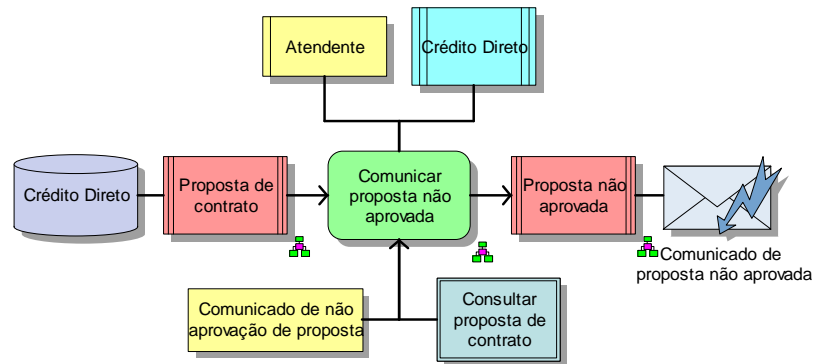
## 2.7 Comunicar proposta não aprovada

O Atendente comunica por telefone, a não aprovação do crédito para o cliente.

A informação necessária é: proposta de crédito.

A informação gerada é: proposta não aprovada.

O sistema Crédito Direto apoia a atividade na consulta da proposta de contrato.



Regra de Negócio	Descrição
Comunicado de não aprovação de proposta	O Atendente deve informar ao cliente caso a sua proposta de contrato não tenha sido aprovada.

Requisito de Negócio	Descrição
Consultar proposta de contrato	<p>O Atendente deve consultar as seguintes informações da proposta de contrato:</p> <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- nome do cliente,</li> <li>- CPF,</li> <li>- lista de peças,</li> <li>- valor a ser financiado,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso.</li> </ul> </li> </ul>

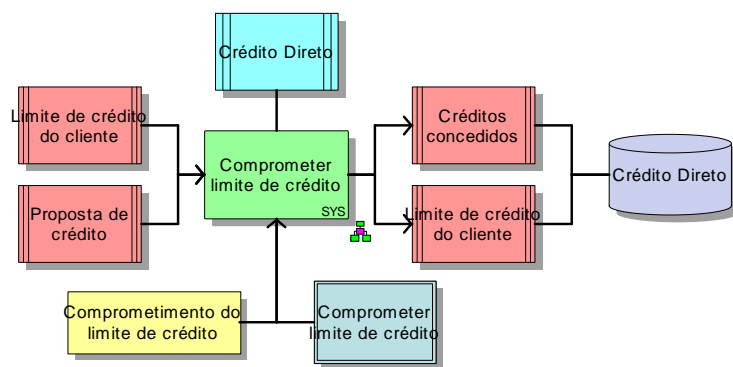
## 2.8 Comprometer limite de crédito

O sistema Crédito Direto compromete o limite de crédito do cliente.

As informações necessárias são: limite de crédito do cliente e proposta de crédito (código da proposta de crédito, lista de peças, valor a ser financiado, número de parcelas, resultado da verificação de crédito (ajustado para aprovado)).

As informações geradas são: créditos concedidos (CPF do cliente, código da proposta de crédito, valor total concedido, número de parcelas, situação de crédito (ajustado para comprometido)) e limite de crédito do cliente.





Regra de Negócio	Descrição
Comprometimento do limite de crédito	No comprometimento do crédito para o cliente a situação de crédito concedido deve ser igual a comprometido.

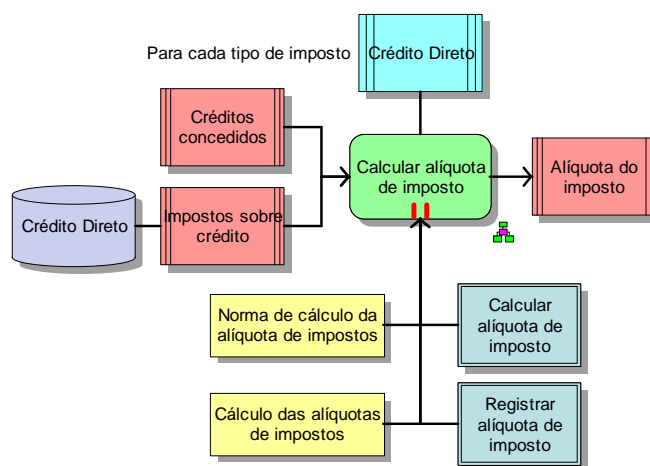
Requisito de Negócio	Descrição
Comprometer limite de crédito	<p>O sistema Crédito Direto deve comprometer o limite de crédito do cliente para a proposta de crédito aprovada, contendo:</p> <ul style="list-style-type: none"> <li>- CPF do cliente,</li> <li>- código da proposta de crédito,</li> <li>- valor total concedido,</li> <li>- número de parcelas,</li> <li>- situação de crédito (comprometido).</li> </ul>

## 2.9 Calcular alíquota de imposto

O sistema Crédito Direto calcula a alíquota do imposto.

As informações necessárias são: créditos concedidos e impostos sobre crédito.

A informação gerada é: alíquota do imposto.



Regra de Negócio	Descrição
Cálculo das alíquotas de impostos	Deverá ser calculada a alíquota de imposto para cada tipo de imposto existente.

Norma de cálculo da alíquota de impostos	O cálculo da alíquota de impostos deve considerar as normas da legislação local vigente.
--	--

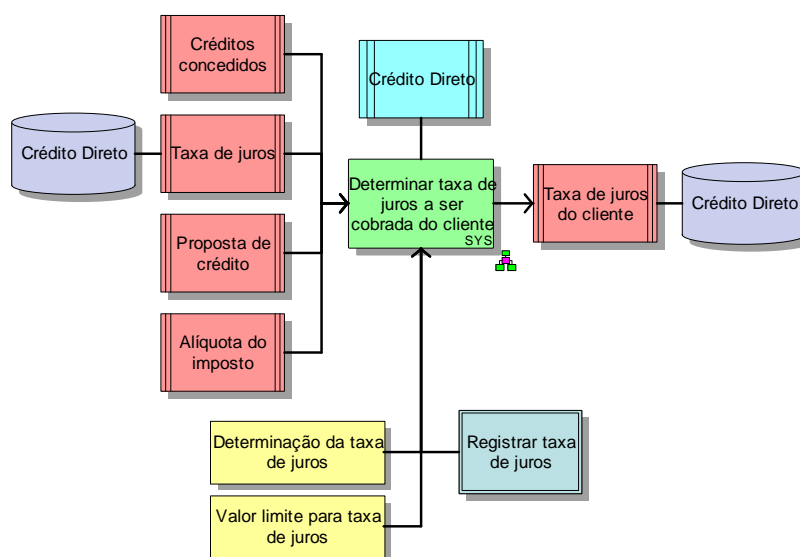
Requisito de Negócio	Descrição
Calcular alíquota de imposto	O sistema Crédito Direto calcula a alíquota de imposto baseada no montante que consta no pedido de crédito.
Registrar alíquota de imposto	O sistema Crédito Direto deve registrar cada alíquota de imposto calculada contendo: <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- CPF do cliente,</li> <li>- montante de crédito concedido,</li> <li>- nome do imposto</li> <li>- fórmula da alíquota</li> <li>- valor da alíquota do imposto.</li> </ul>

## 2.10 Determinar taxa de juros a ser cobrada do cliente

O sistema Crédito Direto determina a taxa de juros a ser cobrada do cliente.

As informações necessárias são: créditos concedidos (código da proposta de crédito, valor total concedido, número de parcelas), taxas de juros e proposta de crédito (código da proposta).

A informação gerada é: taxa de juros do cliente (taxa de juros, para cada parcela: data a ser realizado o pagamento, valor a ser pago, juros correspondentes à multa por atraso).



Regra de Negócio	Descrição
Determinação da taxa de juros	A taxa de juros deve ser determinada de acordo com o valor total solicitado para financiamento e o número de parcelas.
Valor limite para taxa de juros	O valor limite para a taxa de juros é de 12%.

Requisito de Negócio	Descrição
Registrar taxa de juros	<p>O sistema Crédito Direto deve registrar a taxa de juros (de acordo com a regra de negócio Determinação da taxa de juros) e parcelas dos créditos concedidos, contendo:</p> <ul style="list-style-type: none"> <li>- taxa de juros,</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso,</li> <li>- valor pago,</li> <li>- data em que o pagamento foi realizado.</li> </ul> </li> </ul>

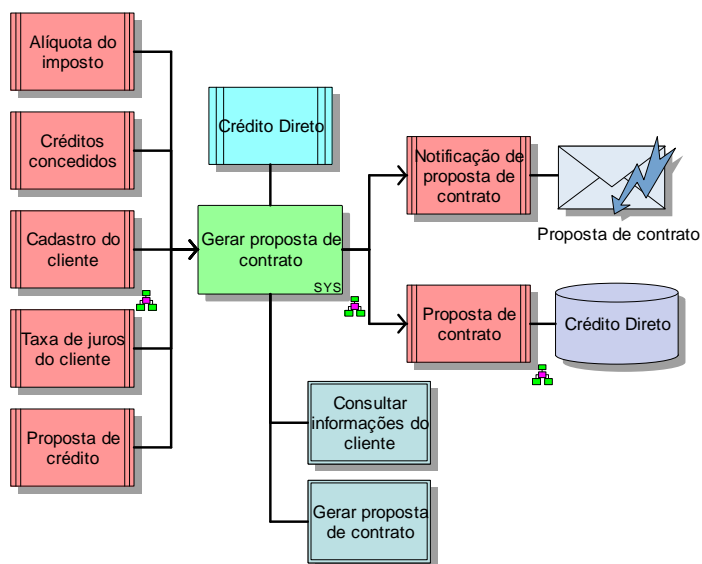
## 2.11 Gerar proposta de contrato

O sistema Crédito Direto gera a proposta de contrato e envia por email para o Analista de Crédito.

As informações necessárias são: alíquota de imposto, créditos concedidos (código da proposta de crédito, taxa de juros, para cada parcela: data a ser realizado o pagamento, valor a ser pago, juros correspondentes à multa por atraso), cadastro do cliente (CPF do cliente, nome do cliente, endereço do cliente, telefone do cliente), taxa de juros do cliente e proposta de crédito (código da proposta, CPF, lista de peças, valor a ser financiado, número de parcelas).

A informação gerada é: notificação de proposta de contrato e proposta de contrato.

A atividade produz como saída o e-mail contendo a notificação de proposta de contrato.



Requisito de Negócio	Descrição
Consultar informações do cliente	<p>O sistema Crédito Direto deve consultar as seguintes informações do cliente:</p> <ul style="list-style-type: none"> <li>- nome,</li> <li>- CPF,</li> <li>- identidade,</li> </ul>

	<ul style="list-style-type: none"> <li>- telefone,</li> <li>- endereço,</li> <li>- renda,</li> <li>- data da última atualização,</li> <li>- data de cadastro.</li> </ul>
Gerar proposta de contrato	<p>O sistema Crédito Direto deve gerar a proposta de contrato, contendo:</p> <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- nome do cliente,</li> <li>- CPF,</li> <li>- endereço do cliente,</li> <li>- telefone do cliente,</li> <li>- lista de peças,</li> <li>- valor a ser financiado,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso.</li> </ul> </li> </ul>

### 2.12 Analisar contrato

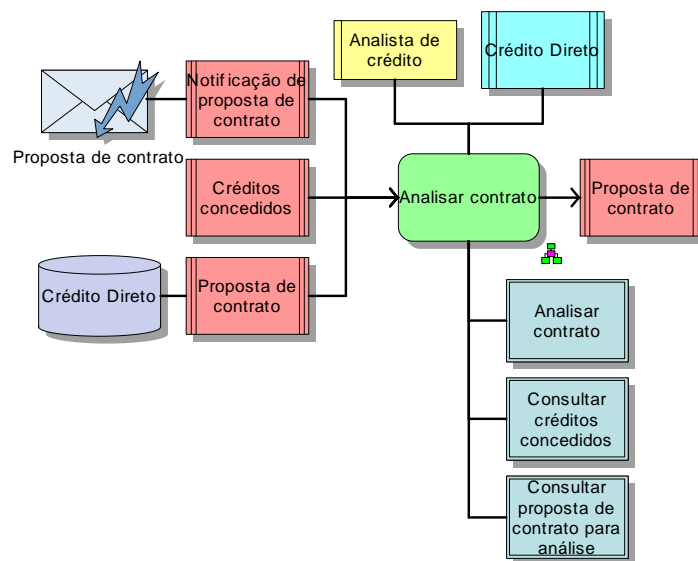
O Analista de Crédito analisa a proposta de contrato e o histórico de créditos concedidos para o cliente a fim de verificar se não há risco elevado no contrato.

A informação necessária é: créditos concedidos e proposta de contrato.

A informação gerada é: proposta de contrato.

A atividade recebe como entrada o e-mail com contendo a notificação de proposta de contrato.

O sistema Crédito Direto apoia a atividade na consulta da proposta de contrato e dos créditos concedidos e na análise do contrato.



Requisito de Negócio	Descrição
Analisar contrato	O Analista de Crédito Contratos deve analisar o contrato ajustando os créditos concedidos na proposta de contrato para cancelado ou aprovado de acordo com a análise realizada.
Consultar créditos concedidos	O Analista de Crédito Contratos deve consultar as seguintes informações de cada concessão de crédito feita para o cliente anteriormente: <ul style="list-style-type: none"> <li>- CPF do cliente,</li> <li>- valor total concedido,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- situação de crédito,</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso,</li> <li>- valor pago,</li> <li>- data em que o pagamento foi realizado.</li> </ul> </li> </ul>
Consultar proposta de contrato para análise	O Analista de Crédito Contratos deve consultar as seguintes informações da proposta de contrato: <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- nome do cliente,</li> <li>- CPF,</li> <li>- lista de peças,</li> <li>- valor a ser financiado,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> </ul> </li> </ul>

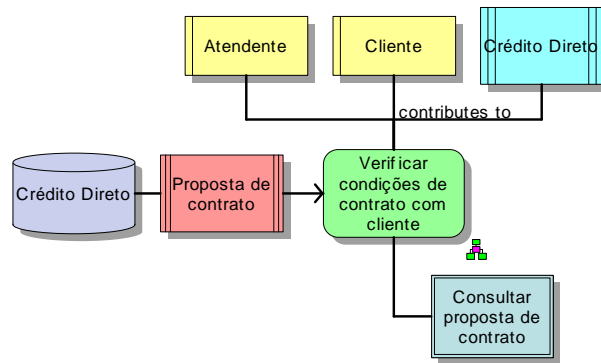
- juros correspondentes à multa por atraso.
---

### 2.13 Verificar condições de contrato com cliente

O Atendente verifica com o cliente, por telefone, se as condições do contrato atendem ao cliente.

A informação necessária é: proposta de contrato.

O sistema Crédito Direto apoia a atividade na consulta da proposta de contrato.



Requisito de Negócio	Descrição
Consultar proposta de contrato	<p>O Atendente deve consultar as seguintes informações da proposta de contrato:</p> <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- nome do cliente,</li> <li>- CPF,</li> <li>- lista de peças,</li> <li>- valor a ser financiado,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- para cada parcela:               <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso.</li> </ul> </li> </ul>

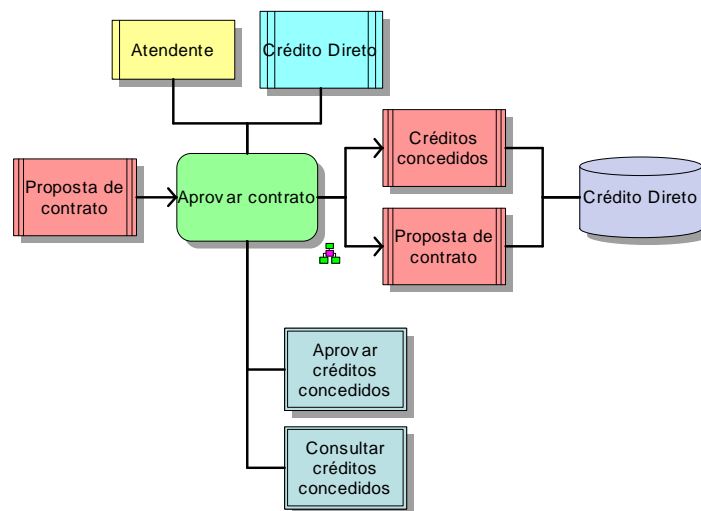
### 2.14 Aprovar contrato

O Atendente registra a aprovação do contrato pelo cliente.

A informação necessária é: proposta de contrato.

A informação gerada é: créditos concedidos e proposta de contrato (com situação igual a "Aprovado pelo cliente").

O sistema Crédito Direto apoia a atividade no acesso a proposta de contrato e no registro dos créditos concedidos e da aprovação da proposta de contrato.



Requisito de Negócio	Descrição
Aprovar créditos concedidos	O Atendente deve aprovar os créditos concedidos para atendimento da proposta do cliente.
Consultar créditos concedidos	<p>O Atendente deve consultar as seguintes informações de cada concessão de crédito feita para o cliente anteriormente:</p> <ul style="list-style-type: none"> <li>- CPF do cliente,</li> <li>- valor total concedido,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- situação de crédito,</li> <li>- para cada parcela:               <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso,</li> <li>- valor pago,</li> <li>- data em que o pagamento foi realizado.</li> </ul> </li> </ul>

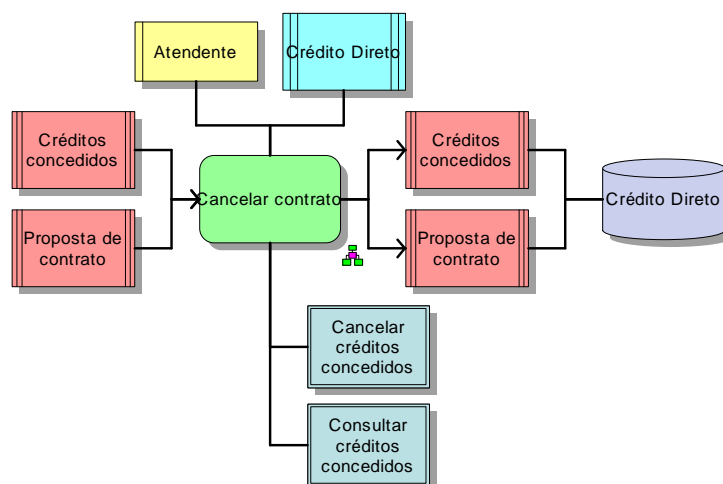
## 2.15 Cancelar contrato

O Atendente cancela o contrato que não foi aceito pelo cliente.

A informação necessária é: créditos concedidos e proposta de contrato.

A informação gerada é: créditos concedidos e proposta de contrato.

O sistema Crédito Direto apoia a atividade no cancelamento da proposta de contrato.



Requisito de Negócio	Descrição
Cancelar créditos concedidos	O Atendente deve cancelar os créditos concedidos para o cliente.
Consultar créditos concedidos	O Analista de Crédito Contratos deve consultar as seguintes informações de cada concessão de crédito feita para o cliente anteriormente: <ul style="list-style-type: none"> <li>- CPF do cliente,</li> <li>- valor total concedido,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- situação de crédito,</li> <li>- para cada parcela:               <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso,</li> <li>- valor pago,</li> <li>- data em que o pagamento foi realizado.</li> </ul> </li> </ul>

## 2.16 Comunicar não aprovação de contrato de risco

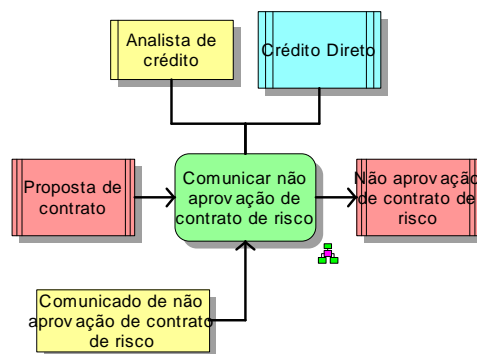
O Analista de Crédito comunica ao Atendente a não aprovação de contrato considerado de risco.

A informação necessária é: proposta de contrato.

A informação gerada é: não aprovação de contrato de risco.

O sistema Crédito Direto apoia a atividade na consulta da proposta de contrato e no envio do comunicado de não aprovação de contrato de risco.





Regra de Negócio	Descrição
Comunicado de não aprovação de contrato de risco	O Analista de crédito deve comunicar ao Atendente caso o contrato não seja aprovado por ser considerado de risco.

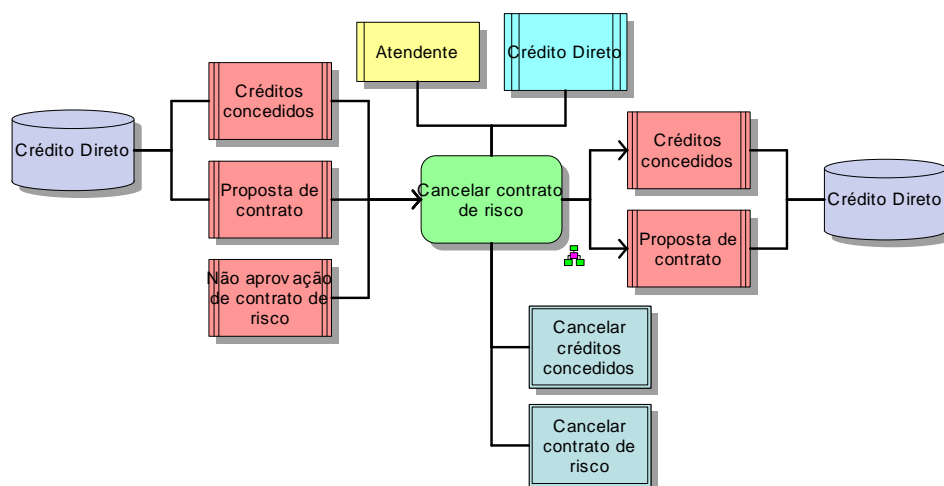
## 2.17 Cancelar contrato de risco

O Atendente registra o cancelamento dos créditos concedidos.

A informação necessária é: créditos concedidos, proposta de contrato e não aprovação de contrato de risco.

A informação gerada é: créditos concedidos (com situação igual a rejeitado pelo cliente) e proposta de contrato.

O sistema Crédito Direto apoia a atividade no recebimento do comunicado de não aprovação de contrato de risco, na consulta dos créditos e da proposta de contrato e no registro do cancelamento dos créditos concedidos e da proposta de contrato.



Requisito de Negócio	Descrição
Cancelar contrato de risco	O Analista de Crédito Contratos deve cancelar os créditos concedidos para o cliente.
Cancelar créditos concedidos	O Atendente deve cancelar os créditos concedidos para o cliente.

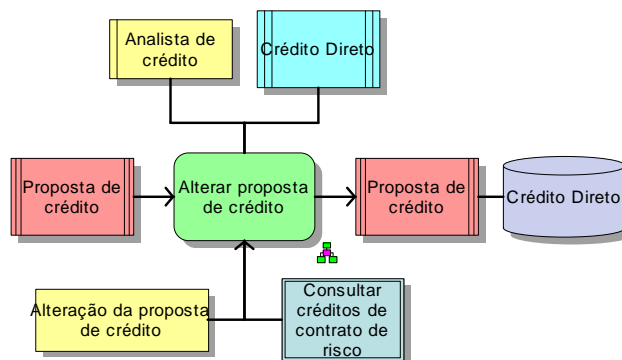
## 2.18 Alterar proposta de crédito

O Analista de Crédito altera a proposta de crédito.

A informação necessária é: proposta de crédito.

A informação gerada é: proposta de crédito.

O sistema Crédito Direto apoia a atividade na alteração da proposta de crédito.



Regra de Negócio	Descrição
Alteração da proposta de crédito	O valor total de crédito concedido e/ou o número de parcelas devem ser reduzidos e a taxa de juros aumentada quando for necessário reduzir o risco da proposta de crédito.

Requisito de Negócio	Descrição
Consultar créditos de contrato de risco	O Analista de Crédito Contratos deve consultar as seguintes informações concessão de crédito para o cliente: <ul style="list-style-type: none"><li>- código da proposta de crédito,</li><li>- CPF do cliente,</li><li>- valor total concedido,</li><li>- número de parcelas,</li><li>- taxa de juros,</li><li>- situação de crédito,</li><li>- para cada parcela:<ul style="list-style-type: none"><li>- data a ser realizado o pagamento,</li><li>- valor a ser pago,</li><li>- juros correspondentes à multa por atraso,</li><li>- valor pago,</li><li>- data em que o pagamento foi realizado.</li></ul></li></ul>

## 2.19 Elementos de Apoio ao Processo

## 2.20 Regras de Negócio

A tabela abaixo apresenta as regras de negócio impostas ao macroprocesso Analisar pedido de crédito.

Regra de Negócio	Descrição
Alteração da proposta	O valor total concedido, o número de parcelas e a taxa de

de crédito	juros devem ser alterados quando for necessário reduzir o risco da proposta de crédito.
Cálculo das alíquotas de impostos	Deverá ser calculada a alíquota de imposto para cada tipo de imposto existente.
Comprometimento do limite de crédito	No comprometimento do crédito para o cliente a situação de crédito concedido deve ser igual a comprometido.
Comunicado de não aprovação de contrato de risco	O Analista de crédito deve comunicar ao Atendente caso o contrato não seja aprovado por ser considerado de risco.
Comunicado de não aprovação de proposta	O Atendente deve informar ao cliente caso a sua proposta de contrato não tenha sido aprovada.
Determinação da taxa de juros	A taxa de juros deve ser determinada de acordo com o valor total solicitado para financiamento e o número de parcelas de acordo com as taxas de juros cadastradas.
Norma de cálculo da alíquota de impostos	O cálculo da alíquota de impostos deve considerar as normas da legislação local vigente.
Valor limite para taxa de juros	O valor limite para a taxa de juros é de 12%.
Verificação cadastro cliente desatualizado	O cadastro do cliente está desatualizado se o telefone, endereço e renda informados na proposta de crédito forem diferentes das informações do cliente existentes na base de dados. A comparação entre proposta de crédito e cadastro do cliente deve ser feita pelo CPF.
Verificação cliente novo	Um cliente é novo se não existir cadastro de cliente com mesmo CPF que o informado na proposta de crédito.
Verificação de limite de crédito	O limite de crédito do cliente é igual a 20% da sua renda menos o valor mensal referentes às parcelas ainda em aberto dos créditos anteriormente concedidos ao cliente.

## 2.21 Bases de Dados

A tabela abaixo apresenta as bases de dados do macroprocesso Analisar pedido de crédito.

Base de Dados	Descrição
Crédito Direto	Representa a base de dados do sistema Crédito Direto.

## 2.22 Informações

As tabelas abaixo apresentam as informações do macroprocesso Analisar pedido de crédito.

Campo	Descrição
Nome:	Alíquota do imposto
Descrição:	Representa o valor calculado da alíquota de um imposto sobre crédito.

Campo	Descrição
Nome:	Cadastro do cliente
Descrição:	Representa as informações do cadastro do cliente, contendo: <ul style="list-style-type: none"> <li>- nome,</li> <li>- CPF,</li> <li>- identidade,</li> <li>- telefone,</li> <li>- endereço,</li> <li>- renda,</li> <li>- data da última atualização,</li> <li>- data de cadastro.</li> </ul>

Campo	Descrição
Nome:	Créditos concedidos
Descrição:	Representa as informações dos créditos concedidos aos clientes, contendo para cada concessão de crédito: <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- CPF do cliente,</li> <li>- valor total concedido,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- situação de crédito (comprometido, cancelado pelo analista de crédito, aprovado pelo analista de crédito, rejeitado pelo cliente, aprovado pelo cliente),</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso,</li> <li>- valor pago,</li> <li>- data em que o pagamento foi realizado.</li> </ul> </li> </ul>

Campo	Descrição
Nome:	Impostos sobre crédito
Descrição:	Representa os impostos que devem ser incluídos em contratos de crédito segundo a legislação vigente local.

Campo	Descrição
Nome:	Limite de crédito do cliente
Descrição:	Representa o valor de limite de crédito do cliente.

Campo	Descrição
Nome:	Não aprovação de contrato de risco
Descrição:	Representa o comunicado de não aprovação do contrato de crédito solicitado por conta do nível de risco envolvido, emitido pelo Analista de crédito ao Atendente.

Campo	Descrição
Nome:	Notificação de proposta de contrato
Descrição:	Representa a notificação de proposta de contrato emitida por e-mail pelo sistema Crédito Direto ao Analista de crédito, após gerar a proposta de contrato.

Campo	Descrição
Nome:	Proposta de contrato
Descrição:	Representa a proposta de contrato, contendo: <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- nome do cliente,</li> <li>- CPF,</li> <li>- endereço do cliente,</li> <li>- telefone do cliente,</li> <li>- lista de peças,</li> <li>- valor a ser financiado,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso.</li> </ul> </li> </ul>

Campo	Descrição
Nome:	Proposta de crédito
Descrição:	Representa as informações do solicitante e dos produtos a serem financiados necessárias para solicitação de crédito, contendo: <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- nome do solicitante,</li> <li>- CPF,</li> <li>- identidade,</li> <li>- telefone do solicitante,</li> <li>- endereço do solicitante,</li> <li>- lista de peças,</li> <li>- valor a ser financiado,</li> <li>- número de parcelas,</li> <li>- renda,</li> <li>- data de cadastro,</li> <li>- responsável pelo cadastro,</li> <li>- resultado da verificação de crédito (aprovado ou reprovado).</li> </ul>

Campo	Descrição
Nome:	Proposta não aprovada

Descrição:	Representa o comunicado de não aprovação de crédito enviado pelo atendente ao cliente.
------------	--

Campo	Descrição
Nome:	Taxa de juros
Descrição:	Representa as taxas de juros cadastradas, contendo para cada faixa de valor de financiamento: - valor total mínimo de financiamento, - valor total máximo de financiamento, - número de parcelas mínimo, - número de parcelas máximo, - taxa de juro.

Campo	Descrição
Nome:	Taxa de juros do cliente
Descrição:	Representa o valor da taxa de juros que foi determinada para o cliente.

## 2.23 Requisitos de Negócio

A tabela abaixo apresenta os requisitos do macroprocesso Analisar pedido de crédito.

Requisito de Negócio	Descrição
Analisar contrato	O Analista de Crédito Contratos deve analisar o contrato ajustando os créditos concedidos na proposta de contrato para cancelado ou aprovado de acordo com a análise realizada.
Aprovar créditos concedidos	O Atendente deve aprovar os créditos concedidos para atendimento da proposta do cliente.
Atualizar informações do cliente	O sistema Crédito Direto deve atualizar as seguintes informações do cliente (de acordo com a regra de negócio Verificação cadastro cliente desatualizado): - telefone, - endereço, - renda.
Calcular alíquota de imposto	O sistema Crédito Direto calcula a alíquota de imposto baseada no montante que consta no pedido de crédito.
Cancelar contrato de risco	O Analista de Crédito Contratos deve cancelar os créditos concedidos para o cliente.
Cancelar créditos concedidos	O Atendente deve cancelar os créditos concedidos para o cliente.
Cancelar proposta	O sistema Crédito Direto deve cancelar a proposta do cliente.
Comprometer limite de crédito	O sistema Crédito Direto deve comprometer o limite de crédito do cliente para a proposta de crédito aprovada, contendo:

	<ul style="list-style-type: none"> <li>- CPF do cliente,</li> <li>- código da proposta de crédito,</li> <li>- valor total concedido,</li> <li>- número de parcelas,</li> <li>- situação de crédito (comprometido).</li> </ul>
Consultar créditos concedidos	<p>O Analista de Crédito Contratos deve consultar as seguintes informações de cada concessão de crédito feita para o cliente anteriormente:</p> <ul style="list-style-type: none"> <li>- CPF do cliente,</li> <li>- valor total concedido,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- situação de crédito,</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso,</li> <li>- valor pago,</li> <li>- data em que o pagamento foi realizado.</li> </ul> </li> </ul>
Consultar créditos de contrato de risco	<p>O Analista de Crédito Contratos deve consultar as seguintes informações concessão de crédito para o cliente:</p> <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- CPF do cliente,</li> <li>- valor total concedido,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- situação de crédito,</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso,</li> <li>- valor pago,</li> <li>- data em que o pagamento foi realizado.</li> </ul> </li> </ul>
Consultar informações do cliente	<p>O sistema Crédito Direto deve consultar as seguintes informações do cliente:</p> <ul style="list-style-type: none"> <li>- nome,</li> <li>- CPF,</li> <li>- identidade,</li> <li>- telefone,</li> <li>- endereço,</li> <li>- renda,</li> <li>- data da última atualização,</li> <li>- data de cadastro.</li> </ul>
Consultar proposta de contrato	<p>O Atendente deve consultar as seguintes informações da proposta de contrato:</p> <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- nome do cliente,</li> <li>- CPF,</li> </ul>

	<ul style="list-style-type: none"> <li>- lista de peças,</li> <li>- valor a ser financiado,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso.</li> </ul> </li> </ul>
Consultar proposta de contrato para análise	<p>O Analista de Crédito Contratos deve consultar as seguintes informações da proposta de contrato:</p> <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- nome do cliente,</li> <li>- CPF,</li> <li>- lista de peças,</li> <li>- valor a ser financiado,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso.</li> </ul> </li> </ul>
Consultar proposta de crédito	<p>O sistema Crédito Direto deve consultar as seguintes informações da proposta de crédito:</p> <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- nome do solicitante,</li> <li>- CPF,</li> <li>- identidade,</li> <li>- telefone do solicitante,</li> <li>- endereço do solicitante,</li> <li>- lista de peças,</li> <li>- valor a ser financiado,</li> <li>- número de parcelas,</li> <li>- renda,</li> <li>- data de cadastro,</li> <li>- responsável pelo cadastro,</li> <li>- resultado da verificação de crédito.</li> </ul>
Gerar proposta de contrato	<p>O sistema Crédito Direto deve gerar a proposta de contrato, contendo:</p> <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- nome do cliente,</li> <li>- CPF,</li> <li>- endereço do cliente,</li> <li>- telefone do cliente,</li> <li>- lista de peças,</li> <li>- valor a ser financiado,</li> <li>- número de parcelas,</li> <li>- taxa de juros,</li> <li>- para cada parcela:</li> </ul>



	<ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso.</li> </ul>
Incluir cliente	<p>O sistema Crédito Direto deve criar cadastro do cliente se o cliente for novo (de acordo com a regra de negócio Verificação cliente novo), contendo as seguintes informações:</p> <ul style="list-style-type: none"> <li>- nome,</li> <li>- telefone,</li> <li>- endereço,</li> <li>- renda,</li> <li>- data de cadastro.</li> </ul>
Registrar alíquota de imposto	<p>O sistema Crédito Direto deve registrar cada alíquota de imposto calculada contendo:</p> <ul style="list-style-type: none"> <li>- código da proposta de crédito,</li> <li>- CPF do cliente,</li> <li>- montante de crédito concedido,</li> <li>- nome do imposto</li> <li>- fórmula da alíquota</li> <li>- valor da alíquota do imposto.</li> </ul>
Registrar taxa de juros	<p>O sistema Crédito Direto deve registrar a taxa de juros (de acordo com a regra de negócio Determinação da taxa de juros) e parcelas dos créditos concedidos, contendo:</p> <ul style="list-style-type: none"> <li>- taxa de juros,</li> <li>- para cada parcela: <ul style="list-style-type: none"> <li>- data a ser realizado o pagamento,</li> <li>- valor a ser pago,</li> <li>- juros correspondentes à multa por atraso,</li> <li>- valor pago,</li> <li>- data em que o pagamento foi realizado.</li> </ul> </li> </ul>
Verificar limite de crédito do cliente	<p>O sistema crédito direto deve verificar o limite de crédito do cliente de acordo com a regra de negócio Verificação de limite de crédito.</p>

# ANEXO III – MODELOS DE CLASSES

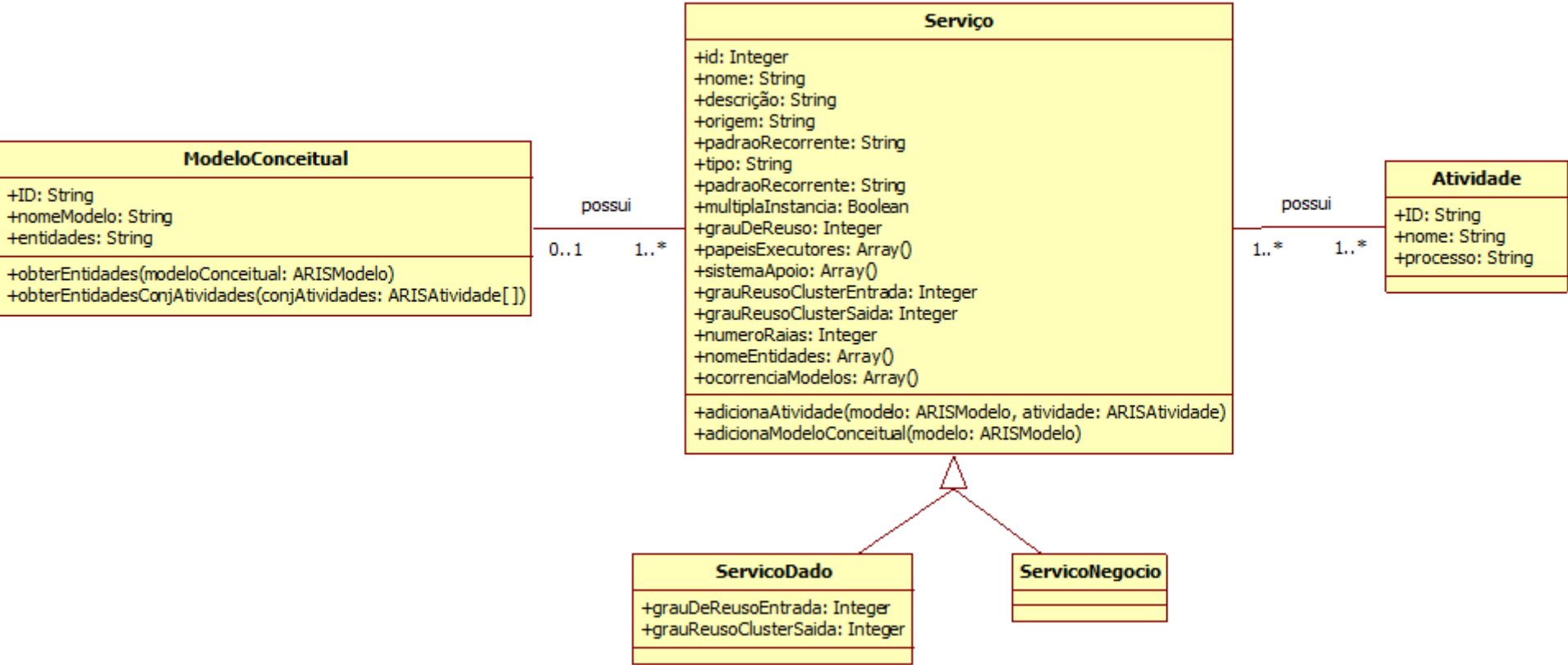


Figura 1 - Anexo III - Diagrama de classes contendo as principais classes do processo de identificação automática de serviços candidatos (a)

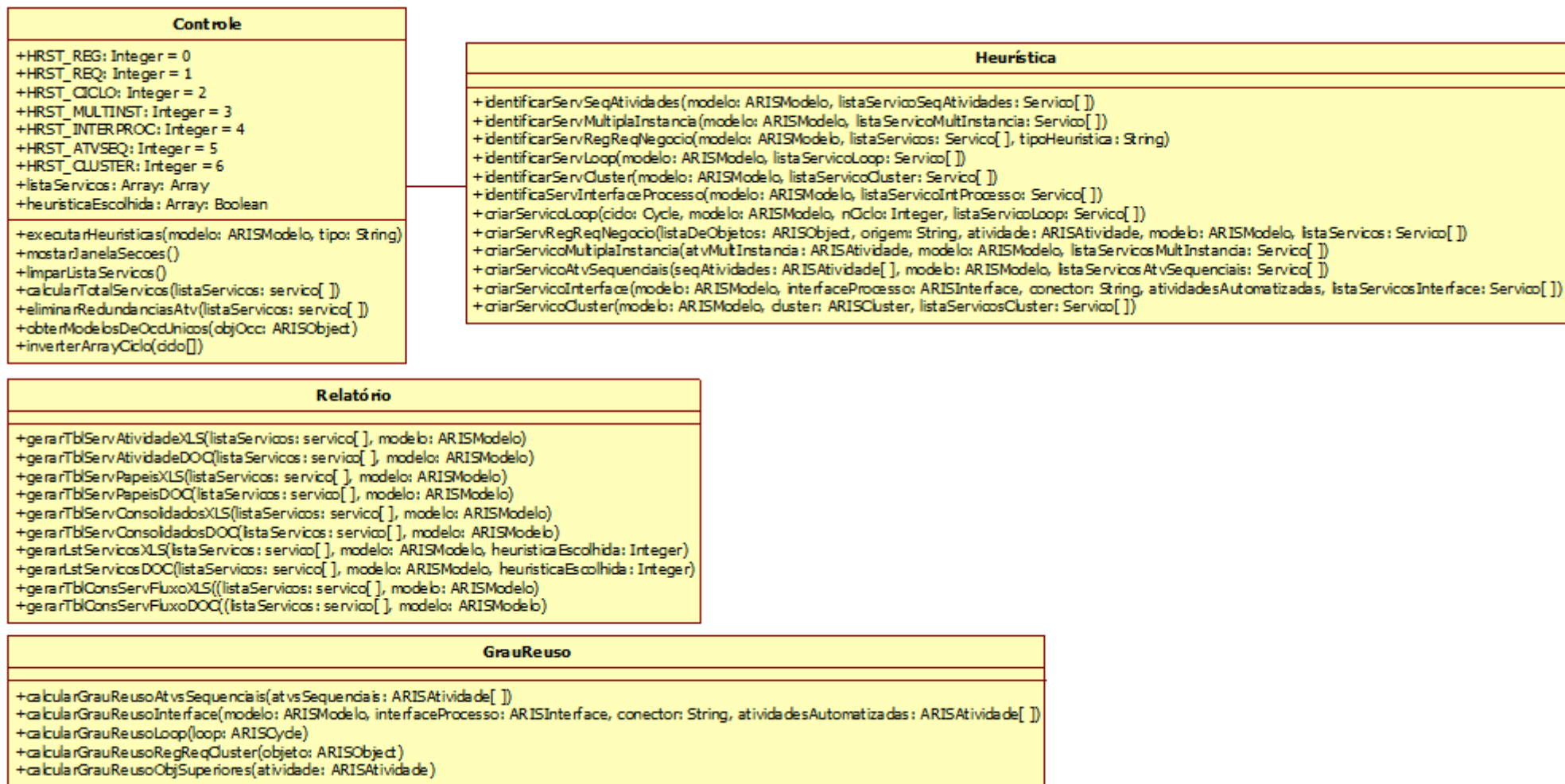


Figura 2 - Anexo III - Diagrama de classes contendo as principais classes do processo de identificação automática de serviços candidatos (b)

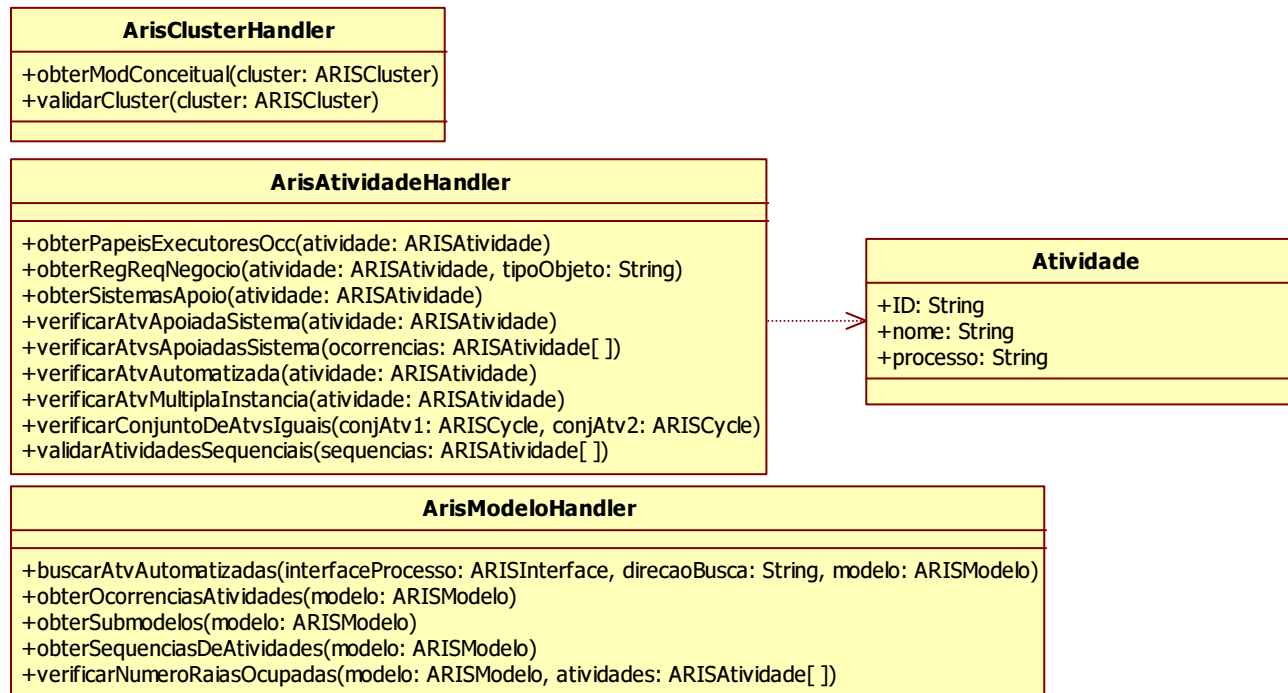


Figura 3 - Anexo III - Diagrama de classes contendo as classes de manipulação de objetos do ARIS

