

ANDRÉ FERREIRA DIAS MARTINS

TRANSFORMAÇÃO ARQUITETURAL DE CONTACT CENTER MULTICANAL COM
RM-ODP E UML4ODP

São Paulo

2014

ANDRÉ FERREIRA DIAS MARTINS

TRANSFORMAÇÃO ARQUITETURAL DE CONTACT CENTER MULTICANAL COM
RM-ODP E UML4ODP

Monografia apresentada ao PECE –
Programa de Educação Continuada de
Engenharia da Universidade de São Paulo.

São Paulo
2014

ANDRÉ FERREIRA DIAS MARTINS

TRANSFORMAÇÃO ARQUITETURAL DE CONTACT CENTER MULTICANAL COM
RM-ODP E UML4ODP

Monografia apresentada ao PECE –
Programa de Educação Continuada de
Engenharia da Universidade de São Paulo.

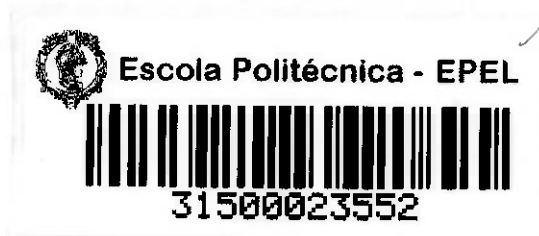
Área de Concentração:
MBA em Tecnologia da Informação.

Orientador:
Prof. Dr. Jorge Luis Risco Becerra

São Paulo
2014

MBA/TI
2014
M366T

552



Catologação na publicação

M2014AE ✓ ★

Martins, André Ferreira Dias

Transformação arquitetural de contact center multicanal com RM-ODP e UML4ODP / A.F.D. Martins. -- São Paulo, 2014. 82 p.

Monografia (MBA em Tecnologia da Informação) – Escola Politécnica da Universidade de São Paulo. Programa de Educação Continuada em Engenharia. PUS

1.Sistemas de informação 2.Sistemas distribuídos 3.Central de atendimento 4. Contact Center 5. Transformação Arquitetural 6. Architectural Driven Modernization 7.RM-ODP 8.UML4ODP

I. Universidade de São Paulo. Escola Politécnica. Programa de Educação Continuada em Engenharia

[2679544]

Dedico este trabalho a:
minha esposa Suzana,
a minha avô Celeste,
e aos meus pais Edgard e Elisabete.

AGRADECIMENTOS

Ao orientador e amigo Dr. Prof. Jorge Luis Risco Becerra, pela orientação, compreensão, paciência e motivação em todos os momentos.

Aos meus pais Edgard e Elisabete pelo apoio e motivação pelo estudo desde criança.

A minha esposa Suzana, pela paciência, incentivo e pela compreensão do tempo em que não pude dar atenção a ela.

A minha tia Mirian Celeste pela revisão do texto, mesmo não conhecendo a área de tecnologia, e por suas valiosas dicas de redação e formatação do texto.

A todos os professores da Escola Politécnica da USP e colegas de classe que colaboraram direta ou indiretamente para a realização do trabalho.

RESUMO

Com a evolução das Centrais de Atendimento nos últimos anos, principalmente com os avanços tecnológicos nas áreas de telefonia e computação, houve uma expansão natural na forma que as pessoas se comunicam. A Internet trouxe novas formas de interagir com as empresas e, assim, nos últimos anos os canais digitais estão cada vez mais presentes oferecendo um serviço rápido, fácil, seguro e simples. A integração dos conceitos estratégicos, como Customer Relationship Management - CRM, Base de Conhecimento, *Data Warehouse* e *Data Mining* é assunto consolidado no mercado, e a central de atendimento passou a ser chamada de *Contact Center*. Sendo o foco contemporâneo da Central de Atendimento a integração dos dados gerados por clientes e consumidores nas mídias sociais com os demais conceitos estratégicos, surge um novo desafio de coletar, processar e agir sobre essas informações disponibilizadas na Internet.

Dessa forma, tomando como arquitetura 'as-is' um modelo de arquitetura aberta de Central de Atendimento, utilizamos técnicas de transformação arquitetural do *Architectural Driven Modernization*, para obter uma nova arquitetura 'to-be' de *Contact Center*, em que são integrados ao modelo de Central de Atendimento os meios de comunicação contemporâneos.

Portanto o desenvolvimento de uma arquitetura aberta de *Contact Center* é facilitado quando utilizamos modelos testados, em uso e maduros como modelos de referencia para a transformação arquitetural, garantindo a evolução do ambiente existente para as necessidades contemporâneas através do reuso do acervo existente e a integração de novos requisitos e novos contextos no desenvolvimento do sistema desejado.

ABSTRACT

With the evolution of the call center in the last years, mainly because of the technological innovations in the telephony and computing areas, there were a natural expansion in the way people communicate. The internet brought us new ways to interact with businesses and in recent years digital channels are ever more present, offering fast, easy, secure and simple services to the customer. Integration with strategical concepts as CRM, Knowledge Base, Data Warehouse and Data mining are already well established in the Market, and in recent years the Call Centers are now known as contact centers. Being the contemporary focus of the call center the integration of the generated data by clients and consumers in social-media with the remaining strategical concepts, a new challenge arises to collect, process and act upon these new information sources.

Thus, taking an existing 'as-is' architectural model of a call center, we use the *Architectural Driven Model* tools and techniques to obtain a new 'to-be' architecture that are integrated with contemporary communication media.

This way the development of a new architecture model is facilitated with the use of mature well known models as reference for the architectural transformation. Ensuring the evolution of the existing systems to the contemporary needs with the reuse of the existent collection of parts and the integration of new requirements and new context to obtain the desired system.

LISTA DE FIGURAS

Figura 1.1 – Desafio de gerenciamento do contact center	3
Figura 1.2 – Metodologia utilizada para desenvolvimento do trabalho.....	5
Figura 2.1 – Triangulo de confluência da estratégia de negócio do contact center.....	7
Figura 2.2 – Arquitetura em Camadas de um Unified Communication System.....	13
Figura 2.3 – Fluxo básico de um chamado no contact center Multicanal.....	15
Figura 2.4 – Proposta de Arquitetura Aberta de Gonçalves (2001).....	17
Figura 2.5 – Domínios Arquiteturais de Negócio e TI.....	23
Figura 2.6 – Modelo Ferradura do ADM.....	23
Figura 2.7 – Tipos de modernização orientado a arquitetura	25
Figura 2.8 – O perfil UML4ODP para o ponto de vista informação.	31
Figura 2.9 – Especificação geral do sistema do contact center em UML4ODP	32
Figura 3.1 – Transformação arquitetural com ADM e RM-ODP	35
Figura 3.2 – Metodologia de transformação arquitetural	36
Figura 3.3 – Proposta de Arquitetura Aberta de contact center.....	38
Figura 3.4 – Especificação UML4ODP para o contact center	39
Figura 3.5 – Alguns dos requisitos identificados na Arquitetura Aberta de Central de Atendimento proposta por (GONÇALVES, 2001) e dos novos requisitos	40
Figura 3.6 - Especificação da Comunidade contact center	42
Figura 3.7 - A Estrutura Organizacional da Comunidade do Contact Center	43
Figura 3.8 – Diagrama de Casos de Uso para o Papel Cliente	45
Figura 3.9 – Diagrama de Casos de Uso para o papel Atendente	46
Figura 3.10 – Diagrama de Casos de Usos para os Atores BackOffice e Distribuidor	47
Figura 3.11 – Os processos do contact center descritos como comportamentos	48
Figura 3.12 – Detalhamento do processo de negócio direcionador de ramais.....	50
Figura 3.13 - Máquina de estados para a navegação do processo direcionador de ramais	52
Figura 3.14 - Alguns objetos ações-tipos do ponto de vista informação	54
Figura 3.15 – Esquema invariante do contact center (Diagrama de Classes).....	55
Figura 3.16 – Esquema dinâmico do objeto informação atendimento	56
Figura 3.17 – Exemplo de esquema estático capturado durante a resolução de um Chamado.....	57
Figura 3.18 – Divisão em camadas do ponto de vista computacional.....	60
Figura 3.19 – Parte da Arquitetura de Software para o contact center.....	61
Figura 3.20 – Especificação detalhada das assinaturas das interfaces	62
Figura 3.21 – Alguns componentes computacionais do sistema com suas interfaces expostas e os relacionamentos entre eles.	63
Figura 3.22 – Diagrama de sequencia de registro de solicitação geral.....	64

LISTA DE TABELAS

Tabela 1 - Mapeamento da arquitetura modelo (Gonçalves, 2001) para os pontos de vista RM-ODP	37
--	----

Lista de Abreviaturas

ACD	Automatic Call Distributor (ver DAC)
API	Application Programming Interface
BI	Business Intelligence
BPMN	Business Process Model Notation
CRM	Customer Relationship Management
CTI	Computer Telephony Integration
DAC	Distribuidor Automatico de Chamadas (ver ACD)
DDR	Discagem Direta à Ramais
IEEE	Institute of Electrical and Electronics Engineers
ISO	International Organization for Standardization
ITU-T	International Telecommunication Union – Telecommunication Standardization Sector
IVR	Interactive Voice Response (ver URA)
KDM	Knowledge Discovery Metamodel
PBX	Private Branch Exchange
PABX	Private Automatic Branch Exchange
PDA	Personal Digital Assistant
POP3	Post Office Protocol version 3
PPP	Point-to-Point Protocol
ODBC	Open Data Base Connection
ODP	Open Distributed Processing
RM-ODP	Reference Model – Open Distributed Processing
SAA	Sistema Aberto de Automação
SBVR	Semantics of Business Vocabulary and Business Rules
SMTP	Simple Mail Transmission Protocol
TAPI	Telephony Application Programming Interface
TCP/IP	Transmission Control Protocol/ Internet Protocol
UC	Unified Communications
UML	Unified Modeling Language
UML4ODP	Unified Modeling Language for Open Distributed Processing
URA	Unidade de Resposta Audivel (ver IVR)
VoIP	Voice over Internet Protocol
WAP	Wireless Application Protocol

SUMÁRIO

1 Introdução	1
1.1 Apresentação.....	1
1.2 Objetivo.....	1
1.3 Justificativa.....	2
1.4 Organização.....	4
1.5 Metodologia.....	5
2 Fundamentação Conceitual.....	6
2.1 <i>A central de atendimento e o contact center</i>	6
2.1.1 Arquitetura Aberta no contact center	8
2.1.2 Objetivos e Serviços.....	8
2.1.3 Tipos de Serviços de uma Central de Atendimento	9
2.1.4 Conceitos estratégicos	10
2.1.4.1 CRM.....	10
2.1.4.2 Gerenciamento do Conhecimento.....	11
2.1.4.3 Data Warehouse.....	11
2.1.4.4 Data Mining.....	12
2.1.5 Tecnologias de Integração	12
2.1.5.1 Unified Communications.....	12
2.1.5.1.1 CTI.....	12
2.1.5.1.2 PBX / IP PBX.....	13
2.1.5.1.3 DAC	14
2.1.5.1.4 URA	14
2.1.5.1.5 Communication Manager	14
2.1.5.1.6 Media Controller	14
2.1.5.2 Internet / Intranet	16
2.1.6 Arquitetura aberta de central de atendimento	16
2.1.6.1 Visão Operacional:	18
2.1.6.1.1 Operação via Telefonia convencional	18
2.1.6.1.2 Operação via Internet/Intranet.....	19
2.1.6.2 Visão Técnica	20
2.1.6.2.1 Níveis de integração do CTI.....	20
2.1.6.2.2 Protocolos de Telefonia.....	21
2.1.6.2.3 Protocolos Computacionais.....	22
2.1.6.2.4 Métodos de Implantação do CTI	22
2.2 <i>Modernização da arquitetura pelo ADM</i>	22
2.2.1 Tipos de modernização para o Modelo Ferradura	24
2.2.1.1 Modernização orientada para a Técnica	25

2.2.1.2 Modernização orientada para a Aplicação / Dados.....	25
2.2.1.3 Modernização orientada para o negócio	26
2.3 Modelo de referência RM-ODP	26
2.3.1 Pontos de Vistas.....	27
2.3.1.1 Ponto de vista – Empresa.....	28
2.3.1.2 Ponto de vista – Informação.....	28
2.3.1.3 Ponto de vista – Tecnologia	29
2.3.1.4 Ponto de vista – Engenharia.....	30
2.3.1.5 Ponto de vista – Computação.....	30
2.3.2 UML4ODP	30
3 Transformação arquitetural com o uso do RM-ODP e UML4ODP	33
3.1 Contexto do contact center.....	33
3.2 Metodologia de transformação arquitetural com ADM e RM-ODP.....	34
3.2.1 Levantamento da arquitetura 'as-is' (1ª fase).....	35
3.2.2 Construção do modelo 'to-be' (2ª fase).....	35
3.2.3 Mapeamento da arquitetura aberta para o ODP	36
3.3 Arquitetura do contact center	37
3.4 Modelagem RM-ODP	38
3.4.1 Ponto de vista empresa	39
3.4.2 Ponto de vista informação	53
3.4.3 Ponto de vista computação	58
4 Conclusão.....	65

1 Introdução

1.1 Apresentação

É inegável o interesse de qualquer corporação em prover informação e suporte de seus produtos e/ou serviços para atuais e futuros clientes. Para que isso ocorra de forma eficiente e precisa é necessário que o maior número de informações esteja disponível para a corporação, quando e onde for necessário, como afirmam Aksin, Armony, & Mehrotra (2007). Para eles, os custos reduzidos de telecomunicações e tecnologias de informação tornou a consolidação de funções de entrega de informações economicamente viáveis, o que proporcionou o surgimento de grupos especializados em manusear as ligações dos consumidores e clientes.

Os custos cada vez mais reduzidos de tecnologias de informação e telecomunicação – as mesmas forças que promoveram o surgimento e o crescimento da central de atendimento – também levou a rápida desagregação das atividades que lidam com grande quantidade de informações. Junto a essas reduções de custo, Aksin, Armony, & Mehrotra (2007) acrescentam que os avanços nas tecnologias de telecomunicação permitiram fluxos de trabalho muito mais ricos e dinâmicos.

Este grupo corporativo, no início era denominado de '*call center*', ou central de atendimento, pois lidava primariamente com os contatos realizados por ligação telefônica. Hoje, com o avanço das tecnologias disponíveis e o advento da sociedade de novos meios de comunicação com a popularização da internet, e-mail, chat, celular e mídias-sociais, o nome dessa unidade de negócio foi alterado para '*contact center*'. Sem dúvida, esta nomeação implica em revisão conceitual e metodológica.

1.2 Objetivo

O objetivo deste trabalho é o desenvolvimento de uma arquitetura aberta de contact center, através da transformação arquitetural de uma arquitetura 'as-is' para uma arquitetura 'to-be'. Para isso foi utilizada como arquitetura base a arquitetura aberta de central de atendimento proposta por Gonçalves (2001) onde ela propõe uma arquitetura aberta através da integração de conceitos estratégicos de atendimento com as tecnologias de telefonia e computação.

Foi realizada também uma atualização das ferramentas e tecnologias utilizadas no desenvolvimento dessa arquitetura, através do uso das ferramentas definidas pelo ADM (Architectural-Driven Modernization) Task Force da OMG (Object Management Group), e do uso do modelo de referencia "ITU-T Rec. X.901-X.904 and ISO/IEC 10746", mais conhecido como RM-ODP e o uso da notação "ITU-T Rec. X.906 | ISO/IEC 19793: Information technology - Open distributed processing - Use of UML for ODP system specifications", conhecido também como *UML4ODP*.

1.3 Justificativa

Consumidores e clientes estão a cada ano tornando-se mais exigentes em relação aos produtos e/ou serviços prestados pelas empresas, além disso as próprias empresas estão cada vez mais competitivas. Com isso os executivos das empresas estão atentos a satisfazer e fidelizar os clientes à marca através de investimentos em CRM (Customer Relationship Management), e a central de atendimento se insere neste contexto como o principal canal de relacionamento com o cliente (GONÇALVES, 2001).

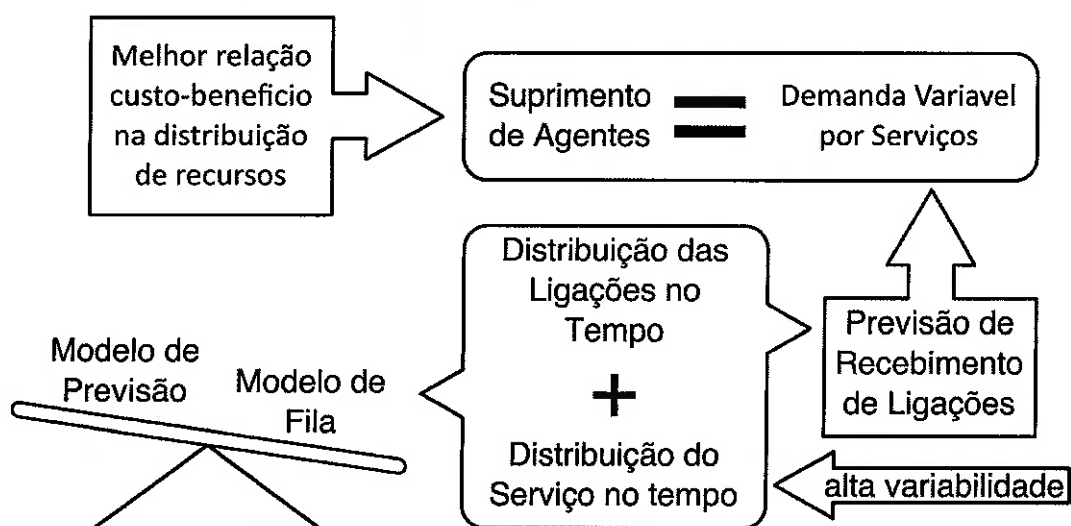
Por muito tempo, a distribuição das consultas realizadas pelos clientes eram feitas, exclusivamente, para o *call center*, e o senso comum do mercado seguia o principio de Pareto, onde 80% dos contatos eram referentes a questões básicas e de fácil resolução, e os restantes 20% dos contatos são as perguntas específicas que não encaixam em um FAQ, sendo difíceis de serem respondidas necessitando de assistência humana (LEE, 2014).

Nos últimos cinco anos, graças aos avanços tecnológicos e a expansão nos canais de atendimento, houve grandes mudanças na forma como os clientes e consumidores entram em contato com a empresa. De acordo com Lee (2014), os clientes primeiramente buscam as respostas aos seus questionamentos por meio de canais de autoatendimento, seja ele por voz ou internet, e buscam o atendimento pessoal por meio da central de atendimento apenas quando a solução não está disponível facilmente. Houve ainda um aumento na terceirização dos serviços providos por centrais de atendimento (*outsourcing*) e a dispersão desse serviço para locais globais (*offshoring*). Do mesmo modo, o desenvolvimento da tecnologia possibilitou fluxos de trabalho mais ricos, incluindo roteamento de ligações entre agentes e sites, interações com o cliente enquanto na fila de espera e *call-back* das

ligações abandonadas pelo cliente enquanto na fila de espera (AKSIN, ARMONY, & MEHROTRA, 2007).

Da mesma forma, que os executivos deram maior importância a central de atendimento, os gerentes desse grupo corporativo tiveram sensibilidade para gerar lucros através da alavancagem, na prestação de serviços, com a oportunidade de vendas. Isto agregou uma complexidade adicional ao gerenciamento do ambiente com esse duplo papel de prestação de serviços e de vendas, tendo em vista a não comprometer os tempos de resposta, a qualidade do serviço e principalmente a satisfação do cliente. Com isso gerentes de centrais de atendimento são desafiados a determinar quantos agentes contratar conforme a necessidade baseados em uma previsão de longo-prazo das demandas por serviço da central de atendimento, decisão conhecida no mercado como "aquisição de recursos" e nas previsões de curto-prazo das escalas de horários para o 'pool' de agentes disponíveis. Além destas há decisões de curtíssimo-prazo, como atualizações de previsões, atualizações de agendamentos e roteamento em tempo real das ligações (AKSIN, ARMONY, & MEHROTRA, 2007).

Figura 1.1 – Desafio de gerenciamento do contact center



Fonte: Autor

As empresas esperam cada vez mais que os gerentes de *call centers* entreguem ao mesmo tempo um serviço de excelência e um baixo custo de operação. Como pode ser observado na Figura 1.1 para chegar a um equilíbrio, os gerentes são constantemente desafiados a disponibilizar o número correto de agentes, com as

capacitações necessárias, nos horários de atendimento corretos, em um ambiente em que a volatilidade da demanda de serviço no tempo é muito alta (AKSIN, ARMONY, & MEHROTRA, 2007).

Portanto com todos esses novos pontos de contato com o cliente que surgiram nos últimos anos, as informações sobre o cliente necessárias para a aplicação dos conceitos de CRM estão dispersos por esses pontos de contato. Tornando a interoperabilidade dos sistemas, através de uma arquitetura aberta, essencial para que a troca das informações seja realizado de forma transparente sobre clientes e consumidores dentro da corporação, com os objetivos de:

- Propiciar aos executivos da corporação informações precisas e recentes para entender e antecipar as necessidades dos clientes e realizar a tomada de decisões estratégicas
- Prever demandas de curtíssimo a longo prazo, prever períodos de pico e ociosidade, de forma a aproveitar o máximos os recursos disponíveis da central de atendimento, para que os gerentes possam eficientemente controlar os custos e qualidade da operação.

1.4 Organização

Para atingir o objetivo proposto nesse trabalho, serão apresentados os conceitos necessários no capítulo 2 – Fundamentação Conceitual – O modelo de central de atendimento proposto por (GONÇALVES, 2001) é explanado e analisado em sua pertinência contemporânea, além da verificação de como a central de atendimento se relaciona com os sistemas corporativos da empresa. Na sequência é acrescentado ao modelo base novos elementos que surgiram no processo de pesquisa e que fazem parte do contexto atual e futuro do *contact center*.

Ainda no capítulo 2, será apresentada técnicas de transformação arquitetural utilizando o *Architectural-Driven Modernization* e o modelo de referência RM-ODP que tem como principais vantagens simplificar o processo de modelagem de sistemas complexos através da divisão de um sistema de informação em cinco pontos de vistas abstratas e distintas, mas interligados entre si por pontos de conformidade entre eles (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012). E finalmente, é apresentada a notação UML e a sua extensão para o ODP,

i.e. o UML4ODP, que fornece ao RM-ODP uma linguagem familiar e altamente expressiva.

No capítulo 3 – Transformação arquitetural com o uso do RM-ODP e UML4ODP – foi desenvolvido, com o uso dos conceitos apresentados no capítulo 2 um novo modelo de referência para *contact center* abrangendo as novas tecnologias e ferramentas existentes no mercado.

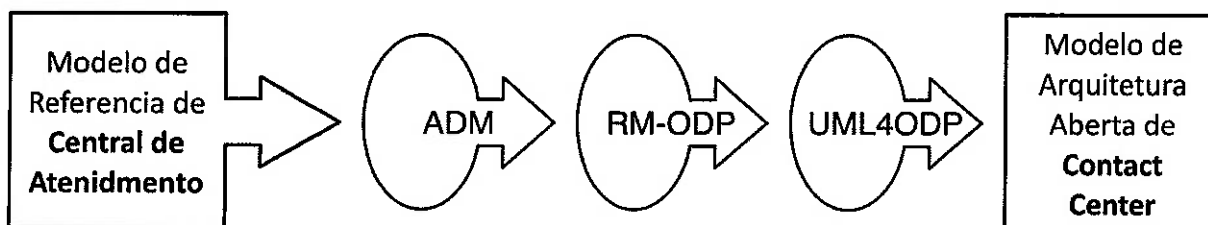
Este trabalho se encerra no capítulo 4 – Conclusão – com a apresentação das conclusões do novo modelo de *contact center* e aponta futuros trabalhos a serem desenvolvidos com os modelos e a metodologia aqui apresentados.

1.5 Metodologia

Foi utilizada a seguinte metodologia para o desenvolvimento do modelo de *contact center* proposto no capítulo 3, como pode ser visto na Figura 1.2:

- Pesquisa bibliográfica
- Análise do modelo de central de atendimento de Gonçalves (2007) para o contexto de *contact center*
- Transformação Arquitetural do modelo de referencia para o modelo de contact center
- Aplicação do RM-ODP com a notação UML4ODP para a definição de um modelo de arquitetura aberta de *Contact center*

Figura 1.2 – Metodologia utilizada para desenvolvimento do trabalho.



Fonte: Autor

2 Fundamentação Conceitual

Neste capítulo estão conceituados os diversos elementos utilizados para o desenvolvimento da arquitetura aberta de *Contact Center* que é apresentada no capítulo 3.

O primeiro item a ser apresentado é o modelo base de Central de Atendimento, conforme proposto por Gonçalves (2001) no seu trabalho "Proposta de arquitetura aberta de central de atendimento", desprezando tecnologias obsoletas, explanando os principais conceitos ainda em uso e acrescentando as tecnologias, ferramentas e conceitos surgidos após o depósito da dissertação.

Em seguida conceituamos o ADM, conforme descrito por Khusidman & Ulrich (2007), seus tipos de transformação arquiteturais e os modelos propostos.

E por último descrevemos o modelo de referência RM-ODP, conforme exposto por Linington, Milosevic, Tanaka, & Vallecillo (2012), descrevendo os pontos de vistas e o relacionamento entre eles, junto com a conceituação da linguagem UML, seus componentes e diagramas tendo como referência Guedes (2011), e ressaltamos as extensões para o UML4ODP conforme descritas por Linington, Milosevic, Tanaka, & Vallecillo (2012).

2.1 A central de atendimento e o contact center

Os *call centers* e *contact centers* são responsáveis por diversas atividades e interações com o cliente, como: Serviço de Atendimento ao Consumidor (SAC), suporte técnico a produtos e serviços (*Help Desk*), informações gerais, prestação de serviços, resolução de problemas nas mais diversas áreas, telemarketing, prospecção de clientes, entre outros.

Esses dois tipos de centrais apesar de terem a mesma função dentro da corporação, utilizam-se das tecnologias disponíveis de forma diferenciada, o *call center* (ou *central de atendimento*) tem seu foco de atendimento voltado ao contatos recebidos pelo telefone, enquanto o *contact center* além do telefone incorpora outros meios de comunicação.

O *contact center* é um ponto central da empresa em que o contato com o cliente é gerenciado através da disponibilidade de vários canais de comunicação, que inclui chamadas de voz, mensagens de texto ou SMS, e-mails, fax, web chat, entre outros,

cumprindo um importante papel no relacionamento corpo-a-corpo com o cliente (ITHAPE & HIWALE, 2014).

Cada *central de atendimento* pode utilizar um ou mais dos canais de comunicação disponíveis no mercado, devendo cada empresa analisar quais os canais tem os melhores benefícios e quais são os mais adequados para atendimento dos seus clientes (GONÇALVES, 2001).

Um *contact center multicanal* é a extensão natural da central de atendimento que lida apenas com chamadas de voz, oferecendo uma gama muito maior de canais para se comunicar com a organização do que a Central de Atendimento, promovendo interações mais personalizadas que enriquecem o relacionamento com o cliente, melhorando as oportunidades de negócio, reduzindo os custos e otimizando a experiência do cliente quando comparado à Central de Atendimento. (ITHAPE & HIWALE, 2014).

Este ambiente hoje é definido por um conjunto de tecnologias em evolução que vão além das chamadas de voz e permite que clientes entrem em contato através de uma ampla gama de componentes integrados, permitindo além da voz, diferentes opções multimídia, como email, SMS, Facebook, Twitter, e mensageria instantânea, i.e. MSN, Skype, Yahoo Messenger, WhatsApp, entre outros (ITHAPE & HIWALE, 2014).

Figura 2.1 – Triangulo de confluência da estratégia de negócio do contact center



Fonte: Gonçalves, 2001

Uma central de atendimento dispõe ainda, de um sistema de tecnologia de informação, que permite a esse departamento exercer todas as atividades relacionadas acima, e ainda manter os registros de um grande volume de chamados independente do canal utilizado. Portanto uma central de atendimento é a combinação de informação, marketing, relacionamento com o cliente, tecnologia de

informação e telefonia, que permite a empresa atuar na satisfação e fidelização do cliente, de forma competitiva e produtiva, como pode ser visto na Figura 2.1 (GONÇALVES, 2001).

2.1.1 Arquitetura Aberta no contact center

É essencial para um bom relacionamento com o cliente estar disponível ao cliente a qualquer hora e local, e para atender a essa necessidade, o *Unified Communications* exerce um papel importante, integrando serviços de comunicação em tempo real, como mensageria instantânea e chat, com serviços de comunicação tradicionais, como email, SMS e chamadas de voz (ITHAPE & HIWALE, 2014).

A arquitetura do *Unified Communication* proposta para o contact center deve se preocupar com a convergência das diferentes tecnologias e formas de contato utilizadas nesse ambiente. A integração entre computação e telefonia é fundamental para que todas as informações necessárias ao cliente estejam disponíveis de forma fácil e rápida, e que os contatos dos clientes sejam manuseados de forma eficiente e atendidos apropriadamente pelos agentes (ITHAPE & HIWALE, 2014).

Essa convergência, ou seja, a integração das tecnologias, como as proporcionadas pelo *Unified Communications* é facilitada quando se adota o uso de sistemas abertos, que disponibilizam especificações abertas para interfaces, serviços e formatos de suporte permitindo sua portabilidade para outros sistemas e plataformas com o mínimo de modificações necessárias de forma a interoperar com aplicações de terceiros em sistemas locais ou remotos (GONÇALVES, 2001).

2.1.2 Objetivos e Serviços

O *contact center*, de acordo com Gonçalves (2001), Aksin, Armony, & Mehrotra (2007) e Ithape & Hiwale (2014), tem como principal foco o atendimento a clientes, mas junto com esse objetivo principal existem outros objetivos que surgiram e se aproveitam da infraestrutura desse ambiente corporativo de forma a melhor aproveitar as variações de uso do *contact center* (AKSIN, ARMONY, & MEHROTRA, 2007):

- Prover um canal de relacionamento com o cliente de forma a obter uma sinergia com os clientes, estimulando a fidelidade e valorizando a imagem da empresa.

- Capturar informações a respeito dos clientes, permitindo a adequação dos produtos e/ou serviços, orientar as ações de marketing, direcionar os esforços de vendas e segmentar e classificar os clientes.
- Determinar mercados não explorados e que possam trazer retorno a empresa.
- Monitorar o comportamento do cliente, para assim agilizar e personalizar o atendimento futuro.
- Aproveitar o atendimento receptivo, iniciado pelo cliente, para oferecer produtos ou serviços, de acordo com o perfil do cliente.

Para que uma organização possa explorar todos os objetivos acima é necessário uma série de conceitos e estratégias de negócio e ferramentas tecnológicas, que são abordados adiante.

2.1.3 Tipos de Serviços de uma Central de Atendimento

Basicamente, o *contact center* presta dois tipos de serviços, conhecidos no mercado como *contact center* receptivos ou ativos. Os *contact centers* ativos, tem como objetivo principal objetivo a prospecção de novos clientes através dos meios de comunicação disponíveis da empresa, sendo o mais utilizado o telefone. Além de oferecer produtos e serviços, podem ser utilizados para realizar pesquisas de satisfação, encerrar chamados em aberto, e para complementar as informações de clientes no banco de dados da empresa por meio de comunicação disponíveis no *contact center*. Por outro lado, temos os *contact centers* receptivos que recebem as solicitações pelos diversos canais de atendimento disponibilizados pela corporação para os seus clientes. Esses canais de comunicação podem ser desde um obsoleto FAX, um e-mail, até soluções complexas como serviços de auto atendimento, interligados aos sistemas corporativos da empresa (GONÇALVES, 2001).

Existe um terceiro tipo de *contact center*, cuja principal atividade é o atendimento receptivo, mas que nos momentos de ociosidade dos atendentes, ou seja, os momentos de pouca procura pelo clientes dos serviços de atendimento pessoal, os atendentes passam a cumprir funções de *contact center* de caráter ativo, exercendo a prospecção de novos e antigos clientes, colhendo informações e realizando pesquisas com clientes ativos, entre outras atividades (AKSIN, ARMONY, & MEHROTRA, 2007).

Os *contact centers* receptivos podem ser subdivididos com diferentes atividades e funções dentro da corporação: o tipo mais comum e o primeiro a ser implantado nas organizações é o *contact center* não especializado, mas com habilidades para tratar questionamentos gerais e problemas de fácil resolução, que também é conhecido como atendimento de nível 0. Os demais tipos de *contact center* se caracterizam por algum tipo de especialização na resolução de problemas, como exemplo, o *Help Desk*, que é especializado na resolução de problemas a produtos e/ou serviços fornecidos pela empresa (GONÇALVES, 2001).

A arquitetura aberta proposta neste trabalho será baseada em um *contact center* de caráter receptivo também com atividades de caráter ativo nos períodos de ociosidade do sistema.

2.1.4 Conceitos estratégicos

2.1.4.1 CRM

O *Customer Relationship Management* é a principal ferramenta estratégica de negócio que pode ser utilizada pelas corporações para a captura interna ou externa de informações do cliente, armazenadas em um repositório local, analisadas e distribuídas aos diversos pontos de contato com o cliente de forma a produzir um relacionamento longo e duradouro com os clientes, fornecedores e intermediários.

O CRM busca a otimização de todos os processos que envolvem o cliente, e não apenas da instalação de uma solução (GONÇALVES, 2001), e de acordo com Peppers e Rogers Group (2000), a sua implantação é realizada seguindo quatro passos.

O primeiro passo é a identificação dos clientes, através do maior número possível de detalhes sobre o maior número possíveis de clientes de forma a reconhecê-los em todos os pontos de contato.

O segundo passo é a diferenciação dos clientes para a empresa, o que pode ocorrer de duas formas: pelo nível de valor que o cliente tem para a empresa ou pelas necessidades de produtos e serviços de seus clientes.

O terceiro passo é interagir de forma eficaz e eficiente com seus clientes, procurando não a forma mais barata de automatizar essa interação, mas uma que resulte na aquisição de novas informações de maneira a fortalecer o relacionamento com o cliente. As interações com o cliente deve seguir o histórico do cliente com a

empresa e novos relacionamentos devem continuar do ponto onde se encerrou a interação anterior.

E por último, o quarto passo é a personalização no relacionamento com cada cliente, e isto é obtido através do marketing direto. Isto quer dizer que clientes diferentes são tratados de forma diferente.

Assim, a implantação de um sistema de CRM precisa levar esses quatro passos em consideração, e realizar a captura, processamento, análise e distribuição de dados focado no cliente como porta de entrada para extração de relatórios e consultas. Para isso, um CRM precisa da integração de vários subsistemas e módulos, como por exemplo: automação de vendas, gerencia de vendas, telemarketing, ferramentas para informações gerenciais, comercio eletrônico, além de utilizar como suporte ferramentas de *Data Warehouses*, *Data Mining*, Gerenciamento do Conhecimento.

2.1.4.2 Gerenciamento do Conhecimento

É a transformação das informações e recursos intelectuais da empresa em algo de valor para os clientes. É realizado através da armazenagem, extração e distribuição de conhecimento de seus funcionários, produtos e serviços de forma que possa ser recuperada e utilizada como ferramenta para tomada de decisões ou apoio a processos de negócio. Na central de atendimento, o gerenciamento de conhecimento é realizado na administração das informações e conhecimentos adquiridos de chamados anteriores (GONÇALVES, 2001).

2.1.4.3 Data Warehouse

O *Data Warehouse* armazena não apenas dados da corporação, mas também ferramentas, procedimentos, treinamento, pessoal e outros recursos que através da manipulação de cópia de informações brutas agrega valor aos dados corporativos extraindo novos dados úteis para a empresa. As informações podem ser originadas dos bancos de dados corporativos, bem como podem ser extraídos de arquivos, fotos, informações coletadas de sites sociais e da internet, e é uma importante fonte de informação para análise e tomada de decisão (GONÇALVES, 2001).

Em alguns casos, pode-se optar pela implantação de uma parte menor de um *Data Warehouse*, conhecido como *Data Mart*, para realizar a análise de um objetivo

específico, ou quando a empresa não tem porte ou recursos para implantação de uma solução de *Data Warehouse* (GONÇALVES, 2001).

2.1.4.4 Data Mining

O *Data Mining* é utilizado para extração de informação válida e desconhecida de dados já existentes no banco de dados corporativos permitindo o descobrimento de relacionamentos escondidos nos dados, permitindo a explicação de eventos observados na mídia ou mercado, confirmando (ou não) hipóteses, ou explorando novos relacionamentos entre as informações analisadas (GONÇALVES, 2001).

2.1.5 Tecnologias de Integração

2.1.5.1 Unified Communications

Definido comumente como sendo a integração de serviços de comunicação de tempo-real como mensageria instantânea, informação de presença, telefonia (convencional e IP), vídeo conferência, compartilhamento da área de trabalho, compartilhamento de dados e reconhecimento de voz, com serviços de comunicação de tempo-não-real como mensageria unificada, ou seja, correio de voz, e-mail, SMS e fax. Não é necessariamente um produto, mas um conjunto de produtos, capazes de prover uma interface de usuário consistente por meio de diversos tipos de mídia (ITHAPE & HIWALE, 2014), como pode ser observado na Figura 2.2.

Pode-se observar na Figura 2.2, e conforme exposto por ITHAPE & HIWALE (2014) que o *contact center* em si faz parte da solução de *unified communications*, e não o oposto.

Para um correto entendimento do trabalho proposto no capítulo 3, explicaremos os principais conceitos dessa arquitetura a partir de Gonçalves (2001).

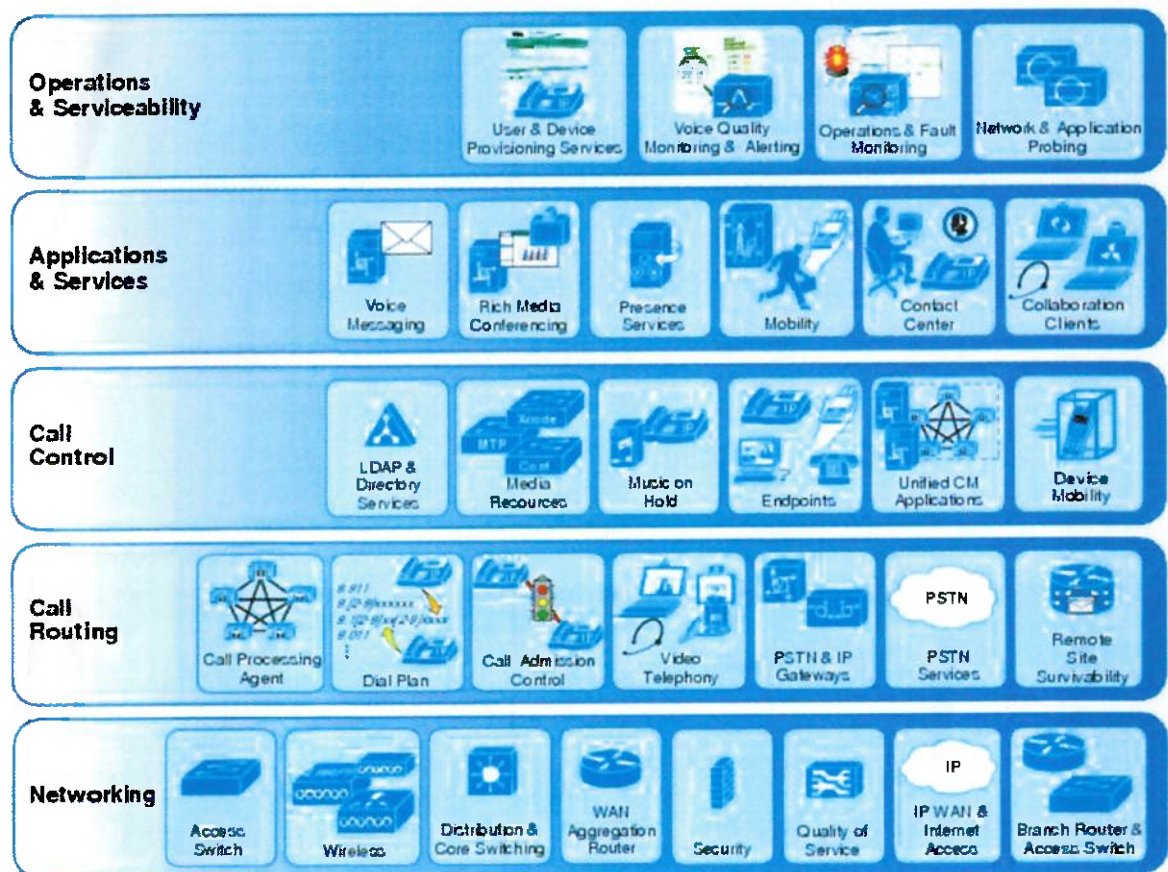
2.1.5.1.1 CTI

O CTI é responsável pela integração entre o sistema telefônico e o sistema computacional, englobando vários serviços, como por exemplo: auto-atendimento, sincronismo de tela para o atendente, discador automático para o telemarketing e servidores de FAX.

A integração pode ser realizada no nível de protocolo ou no nível de aplicativo. A primeira disponibiliza a troca de mensagens, dados, eventos e comandos, enquanto

a segunda foca no fornecimento de dados do sistema computacional para o PBX, ou o estabelecimento ou transferências de chamadas.

Figura 2.2 – Arquitetura em Camadas de um Unified Communication System



Fonte: Cisco, 2012

Tem como principais vantagens: unir os sistemas telefônicos e de computação, através de protocolos padrões, coordenar todo o atendimento telefônico através de aplicativos de software, fornecendo informações do atendimento e do cliente através de consultas ao banco de dados corporativo, propiciar agilidade e facilidade de atendimento ao usuário através da consulta aos bancos de dados pela central de atendimento.

2.1.5.1.2 PBX / IP PBX

É o equipamento de telefonia responsável pelo chaveamento das ligações, interligando ramais e encaminhando as chamadas para os ramais corretos. É ligado

as linhas tronco da prestadora de serviços de telefonia de um lado e do outro lado provê os ramais para as URA's, filas (DAC) e ramais corporativos.

2.1.5.1.3 DAC

O Distribuidor Automático de Chamadas é uma aplicação altamente eficiente e especializada, responsável pela distribuição das ligações recebidas para atendentes dentro de um grupo de atendimento específico, mantendo as ligações em fila quando todos os membros do grupo de atendimento estiverem ocupados e distribuindo-os um-a-um conforme os membros da fila fiquem livres.

Além disso prove estatísticas e informações sobre o funcionamento da fila, permitindo o monitoramento e a contabilização das ligações.

2.1.5.1.4 URA

A Unidade de Resposta Audível prove uma forma de interação entre o usuário e o sistema computacional da empresa de forma autônoma, através de mensagens de voz que orientam o usuário e, a partir da resposta por teclado do aparelho telefônico, captura os dados digitados dando acesso aos serviços e às informações armazenadas nos bancos de dados corporativos.

2.1.5.1.5 *Communication Manager*

Sendo responsável por organizar e distribuir, ou seja fazer o roteamento, dos canais de voz, dados, imagens e transmissões de vídeo, o *Communication Manager* é parte importante da Infraestrutura do *contact center*. Realiza o roteamento inteligente dos contatos, tem funcionalidades de gerenciamento do usuário e do sistema, apresenta características de PBX, é extensível e de fácil integração com outras aplicações voltadas ou não para o *contact center*. Pode ser conectado a rede públicas, como a Internet, ou privadas, como LAN's. É um componente de alta confiabilidade, escalabilidade, e multiprotocolo, ou seja suporta uma grande diversidade de protocolos (ITHAPE & HIWALE, 2014).

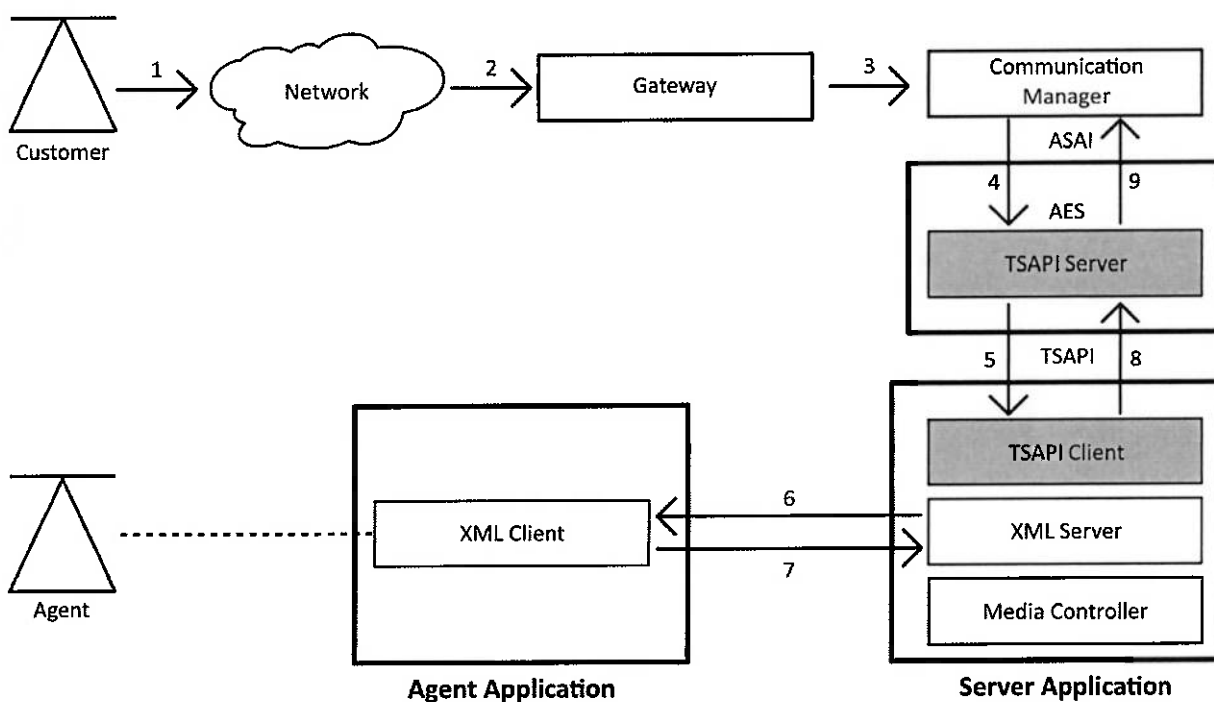
2.1.5.1.6 *Media Controller*

Assim como o *communication manager*, o *media controller* também é um componente essencial ao funcionamento do *contact center* multicanal. Internamente o *media controller* é dividido em dois subcomponentes: o *media director* (MD) e em

um ou mais *media stores* (MS). O *media director* distribui, através da ajuda do componente de fila, presente no *communication manager*, os itens de trabalho não verbais dos clientes para os agentes do *contact center* multicanal. Os itens de trabalho podem ser e-mail, web chat, sms, fax, etc.

Os itens de trabalho distribuídos pelo *media director* são originados dos módulos de *midia stores* apropriados para cada tipo de interação utilizando protocolos de comunicação bem definidos. Por exemplo, o *contact center* utiliza o *e-mail media store* (EMS) para itens de trabalho de *e-mail*, enquanto o *simple messaging media store* (SMMS) é utilizado para itens de trabalho de sms ou web-chat (ITHAPE & HIWALE, 2014).

Figura 2.3 – Fluxo básico de um chamado no *contact center* Multicanal



Fonte: Ithape & Hiwale, 2014

Como podemos observar na Figura 2.3, os módulos do *contact center* multicanais não conseguem se comunicar diretamente e dependem do uso de outros protocolos, para isso é utilizado o *application enablement services* (AES). O AES é uma camada intermediária que permite a comunicação entre o *communication manager* e o servidor de aplicações, através do *telephony server application programming interface* (TSAPI), com o propósito de conseguir a integração telefone-

computador (CTI) e o adjunct/switch application interface (ASAI) (ITHAPE & HIWALE, 2014).

De um lado o cliente entra em contato através da rede, que pode ser o PSTN no caso de chamadas por voz, ou a Internet no caso de ser através de e-mail, sms ou web chat, passando pelo *gateway* apropriado, i.e. *gateway* PSTN ou *gateway* internet, e conectado ao *communication manager*.

Do outro lado desses componentes o agente se conecta através de um cliente XML ao servidor XML do servidor de aplicações, precisando estar autenticado e autorizado, i.e. logado, pelo sistema, tanto no servidor de aplicações quanto no communication manager, e dessa forma o agente está pronto para receber os contatos dos clientes (ITHAPE & HIWALE, 2014).

2.1.5.2 Internet / Intranet

Com a evolução das tecnologias focadas na Internet temos também a evolução das ferramentas que utilizam o meio como forma de criar canais de comunicação entre clientes e empresas (GONÇALVES, 2001). Tecnologias como 'call-me back', Gerenciamento de e-mails, e VoIP adquiriram maturidade e confiabilidade, e novos sistemas de comunicação estão surgindo como o real time communications (RTC) e o WebRTC.

O meio também tem propiciado aos clientes terem voz ativa no relacionamento com as empresas, através de sites sociais, onde os clientes comentam publicamente sobre os produtos, serviços e atendimento de determinada empresa em sites sociais e portais sociais.

De forma idêntica a internet, a intranet, se dá de forma privada pela rede corporativa da empresa pelos funcionários ou pessoas autorizadas para poder registrar, interagir e acompanhar chamados, interagir com um atendente utilizando chat, ou enviar e-mail com sugestões, dúvidas, reclamações ou críticas (GONÇALVES, 2001).

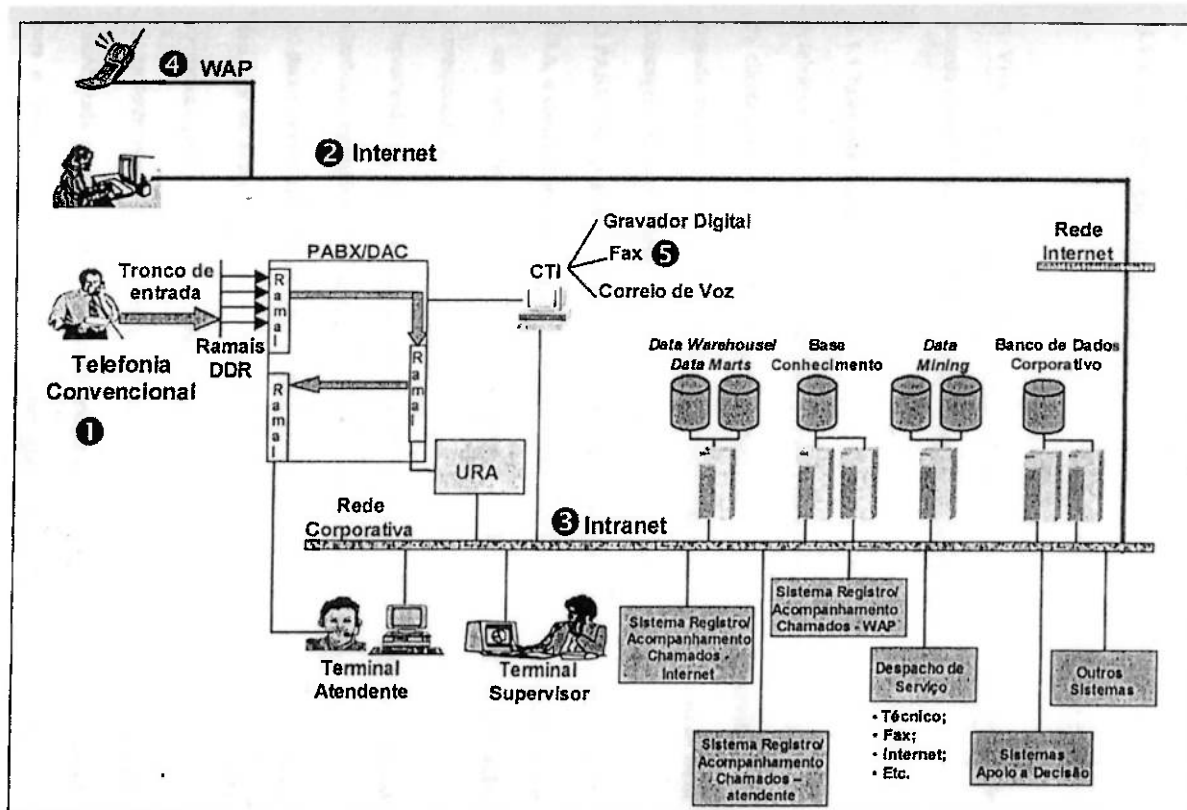
Em ambos os casos, Internet ou Intranet, se utiliza dos mesmos protocolos para comunicação entre os serviços.

2.1.6 Arquitetura aberta de central de atendimento

Antes de prosseguir, é importante notar, com a explanação desse item que o trabalho de (Gonçalves, 2001) foi elaborado para a implantação de um *call center*, e

não um *contact center*, que é o tema deste trabalho. Algumas das tecnologias e conceitos estratégicos utilizados não são mais pertinentes aos dias de hoje, como por exemplo WAP, Gravador Digital e Fax, e não serão detalhados, bem como surgiram novos conceitos e tecnologias da Central de Atendimento que foram abordados nos itens 2.1.4 e 2.1.5. A parte que nós interessa estudada por Gonçalves (2001) é a integração telefonia-computação, conhecida como (CTI), e os sistemas computacionais que dão suporte ao contact center e como o conjunto dessa integração e sistemas computacionais com o CRM corporativo resolvem a questão de custos operacionais sem deixar de lado a preocupação com a satisfação e fidelização dos clientes. Ela demonstra em seu tratado que uma central de atendimento produtiva somente é possível com a integração dos hardware, software, telefonia, metodologias, conceitos estratégicos, banco de dados e tecnologias descritos nos itens acima com o intuito de fornecer serviços de forma transparente com custo operacional compatível, com ótima performance aos seus clientes e profissionais envolvidos de forma a obter a satisfação e a fidelização dos cliente.

Figura 2.4 – Proposta de Arquitetura Aberta de Gonçalves (2001)



Fonte: Gonçalves, 2001

Conforme a Figura 2.4 é possível observar o conjunto da arquitetura proposta por (GONÇALVES, 2001). Essa arquitetura foi criada para atender qualquer segmento de atendimento ao cliente, desde o telemarketing, como também suporte a produtos e serviços (*Help desk*), serviços de atendimento ao consumidor, informações gerais e prestação de serviços.

2.1.6.1 Visão Operacional:

São utilizados por (GONÇALVES, 2001), em sua proposta arquitetural os seguintes canais de comunicação: telefonia convencional (Figura 2.4 (1)), Internet (Figura 2.4 (2)), Intranet (Figura 2.4 (3)), WAP (Figura 2.4 (4)) e Fax (Figura 2.4 (5)), mas por se tratar de uma arquitetura aberta, podem ser facilmente acoplados outras formas de comunicação.

2.1.6.1.1 Operação via Telefonia convencional

Indicado pelo *número 1* na Figura 2.4, representa a ligação de um cliente para um número tronco da empresa, que pode ser composto de um DDR, e conectados a um PABX ou DAC. Estes, por sua vez, encaminham a ligação para um sistema de URA, onde é atendido por uma saudação e um menu de opções navegável através do teclado ou em alguns casos por voz. Dependendo da opção selecionada pelo cliente, o cliente pode continuar sendo atendido pelas opções da URA até obter as informações desejadas do banco de dados corporativo, deixar uma mensagem no sistema de correio de voz, ou caso o cliente precise de auxílio com um assunto que não esteja mapeado nas opções da URA este pode ser transferido para um atendente.

Nesse caso o cliente é então transferido para uma fila de atendimento, onde poderá ser colocado em espera, até que um atendente esteja disponível para atendimento. O sistema de fila identificando a disponibilidade de um atendente transfere simultaneamente para o ramal e o terminal do atendente, i.e. a operação de transferência sincronizada de tela, onde a ligação e os dados coletados durante a passagem do cliente pela URA são transferidos simultaneamente para o atendente.

Os dados recebidos junto com a ligação são visualizados pelo operador no módulo do atendente do Sistema de Registro/Acompanhamento de Chamados. Esse módulo deve ter um roteiro pré-determinado para que o atendimento seja

padronizado, seguindo o perfil do cliente de acordo com a estratégia CRM definida pela empresa para facilitar e agilizar a conversa entre cliente e operador.

2.1.6.1.2 Operação via Internet/Intranet

Representado na Figura 2.4 com o número 2, o meio de comunicação pela Internet pode ser realizado de diversas formas: o cliente pode acessar a página de 'fale conosco' onde preenche um formulário; acessar uma base de conhecimento pública; acessar um portal com usuário e senha para o acompanhamento de chamados abertos, prestação de serviços e acompanhamento de compras; interagir diretamente com um atendente utilizando um sistema de 'web chat' no próprio site da empresa, ou enviar um e-mail com dúvidas, sugestões ou reclamações.

No caso do cliente abrir um chamado com a empresa, os dados do cliente são armazenados no banco de dados corporativo, e dessa forma, é muito importante que a empresa possua toda a infraestrutura de segurança necessária para garantir a integridade e o sigilo dessas informações. Por isso, após o primeiro contato por esse canal de comunicação o cliente recebe, usualmente por e-mail, um usuário e senha para acompanhamento dos chamados abertos.

A integração dos serviços de telefonia com os sistemas computacionais da corporação se dá através do interfaceamento dessas duas tecnologias, telefone e computação, pelo CTI (Computer Telephony Integration). Os sistemas utilizados pela integração entre computação e telefonia são baseadas em conceitos de arquitetura e padrões abertos, como por exemplo, o protocolo TAPI (i.e. Telephony Application Programming Interface) e permite entre outras coisas: o sincronismo do canal de voz com os dados digitados durante todo o ciclo de vida da ligação; a gravação do diálogo entre operador e cliente; o envio e recebimento de FAX de forma automática, a discagem e transferência para um atendente de forma automatizada para prováveis clientes; a gravação de mensagens em correio de voz.

É muito importante para a corporação que todo o histórico de contatos do cliente com a empresa sejam armazenados no banco de dados da empresa, isso permitira que a empresa continue o atendimento do cliente do ponto em que se encerrou o último atendimento, bem como, com a análise dessas informações através de sistemas de *Data Warehouse* e *Data Mining* a empresa pode extrair informações importantes tanto a nível operacional da central de atendimento, bem como padrões de consumo, reclamações e desejos de seus consumidores e gerar uma base de

conhecimento com a solução de dúvidas ou resolução de problemas ocorridos anteriormente.

As informações armazenadas devem estar facilmente acessíveis, independentemente da tecnologia utilizada para acesso a essas informações, pelos usuários do sistema, seja ele o cliente ou funcionários da empresa, e serem filtradas de acordo com a sua função dentro do sistema, i.e. atendentes tem acesso aos dados do cliente e ao histórico de chamados, enquanto supervisores da central tem acesso aos dados de performance do ambiente.

Algumas ligações de clientes para central de atendimento geram chamados que dependem de agentes externos à central de atendimento e não conseguem ser solucionados no momento do contato, dessa forma, a central de atendimento dispara um chamado para o departamento responsável, com um fluxo de trabalho específico e prazo de atendimento máximo para a resolução do problema, de acordo com as estratégias de CRM para o perfil do cliente sendo atendido.

O supervisor da central de atendimento é a pessoa responsável por gerenciar a operação desenvolvendo uma série de atividades, como por exemplo: controlar o volume de chamados em andamento; a qualidade do atendimento; a satisfação do cliente; o tempo de espera na fila; o tempo de atendimento e a troca de responsabilidades.

2.1.6.2 Visão Técnica

Tecnicamente a utilização de tecnologias de padrão aberto e reconhecidos internacionalmente pela *ISO*, *IEEE* e outros garantem a interoperabilidade, a interconectividade e a portabilidade dos aplicativos, operadores e dados propiciando a independência de fornecedores na implementação da central de atendimento.

A especificação técnica ocorre nos protocolos utilizados para a implementação da integração telefonia-computação, nas comunicações de rede computacional, bem como na comunicação com os sistema de banco de dados e até os sistemas operacionais utilizados.

2.1.6.2.1 Níveis de integração do CTI

Nos aspectos de CTI é necessário estar atento a três itens para que a integração ocorra da forma esperada: uma conexão entre o sistema telefônico e o sistema computacional; o reconhecimento dos estados operacionais da telefonia pelo

sistema de computação; e finalmente, o recebimento de comandos do sistema computacional pelo sistema telefônico.

Uma arquitetura aberta de CTI, portanto, depende do uso de uma *conexão* que provê o meio por onde as informações irão trafegar e um *protocolo* que defina as regras, procedimentos e as convenções utilizadas para a troca de dados entre a telefonia e o sistema computacional. É muito importante para a especificação do sistema que o protocolo escolhido seja aberto, pois isso permite que o desenvolvimento de aplicações de telefonia seja feito rapidamente por empresas desenvolvedoras de aplicações, para sistemas de telefonia de diferentes fabricantes.

2.1.6.2.2 Protocolos de Telefonia

Os protocolos de telefonia mais importantes hoje são os seguintes: CSA (CallPath Services Architecture); JTAPI (Java Telephony); CSTA (Computer Supported Telephony Application); TAPI (Telephony Application Programming Interface); e TSAPI (Telephony Services API). Gonçalves (2001) chama atenção para o fato de que dentre todos esses 'protocolos' apenas o CSTA é estritamente um protocolo, sendo os demais considerados API's, mas que para efeitos de simplificação serão todos tratados como protocolo.

Na especificação de uma arquitetura aberta de telefonia deve ser levado em conta alguns aspectos importantes, como o nível de conformidade entre os sistemas especificados na arquitetura, que são categorizados em três níveis:

- Básico – onde existe somente funções para realizar ou atender chamadas telefônicas;
- Suplementar – passa a suportar a transferência de chamados, serviços de conferencia entre outros;
- Avançado – onde suporta funções de fila de atendimento (DAC);

A implementação dos protocolos pelos diversos fornecedores podem ser interpretados de forma diferente, resultando em problemas na integração e dependendo da implementação, os protocolos podem também carregar dados adicionais através da extensão dos protocolos abertos.

2.1.6.2.3 Protocolos Computacionais

Nas comunicações entre sistemas computacionais, utilizam-se protocolos de rede abertos, cujos mais importantes e ainda em uso são: TCP/IP; SMTP e POP3.

A comunicação dos sistemas computacionais com o banco de dados baseado em arquitetura cliente/servidor pode utilizar o ODBC (Open DataBase Connectivity). O ODBC realiza a interface entre os programas aplicativos e o banco de dados, permitindo que as aplicações possam ter acesso aos sistemas gerenciadores de banco de dados e consigam executar comandos SQL na base de dados, independente do fornecedor do Sistema Gerenciador de Banco de Dados, sem alterar o código fonte do programa, desde que este último seja compatível com o ODBC.

A arquitetura do ODBC é constituída da aplicação, do gerenciador de drivers e drivers para conexão com os sistemas gerenciadores de banco de dados e está localizada do lado cliente. Possui três níveis de conformidade API que descrevem o conjunto de funções que a aplicação pode realizar, e três níveis de conformidade da gramática SQL.

2.1.6.2.4 Métodos de Implantação do CTI

Soluções CTI podem ser implementadas tecnicamente de varias maneiras diferentes, para Gonçalves, (2001), as mais comuns e utilizadas são: o *first-party*, e o *third-party*, na primeira solução o CTI está localizado no computador do usuário e pode manusear o recebimento das chamadas, tendo como principal vantagem a facilidade de implementação, mas como desvantagem não tem controle sobre todo o ciclo de vida da ligação telefônica.

A implementação via *third-party*, ao contrario da implementação *first-party*, ocorre diretamente no sistema telefônico, tendo como principal vantagem o controle sobre todo o ciclo de vida da chamada dentro do sistema, mas são normalmente soluções caras e difíceis de implantar (GONÇALVES, 2001).

2.2 Modernização da arquitetura pelo ADM

A modernização das arquiteturas de software com o proposito de auxiliar as iniciativas táticas em sistemas, como por exemplo a manutenção de software, bem como a migrar plataformas e linguagens obsoletas para ambientes mais modernos já

vem ocorrendo a alguns anos, mas só em 2007 os esforços de modernização alcançarão áreas mais significativas e importantes dos domínios de negócios. Abrindo novas oportunidades para os altos escalões das arquiteturas de negócios e TI (KHUSIDMAN & ULRICH, 2007).

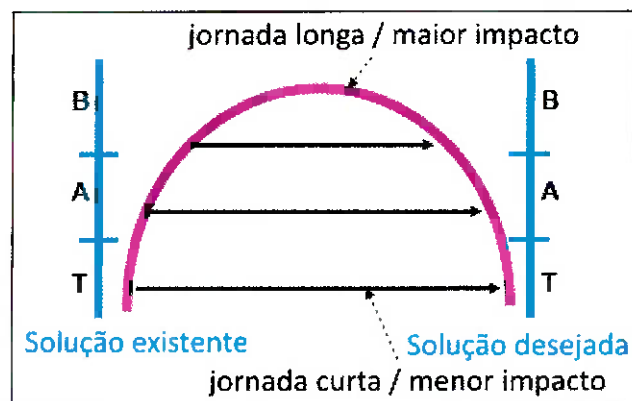
Figura 2.5 – Domínios Arquiteturais de Negócio e TI.



Fonte: Khusidman & Ulrich, 2007

Projetos de modernização tem os mais diversos cenários que podem ser resumidos em 3 perspectivas arquiteturais: Negócio, Aplicação e Dados, e Técnico, dentro de 2 domínios: Negócio e TI, como pode ser observado na Figura 2.5.

Figura 2.6 – Modelo Ferradura do ADM



Fonte: Khusidman & Ulrich, 2007

A Figura 2.5 mostra ao lado esquerdo da figura a solução atual, com seus dados e sistemas, dentro das 3 perspectivas organizacionais, enquanto do lado direito está representado a solução desejada, que é fluida e muda conforme partes da solução existente são migradas para a solução alvo. Essa migração da solução atual para a

solução desejada de maneira fracionada e incremental é a transformação arquitetural do sistema, que podem ser orientados pelo negócio ou pelo TI.

Essas transformações são melhoradas dramaticamente com o uso do ADM (Architectural-Driven Modernization), que é usada para evoluir a solução atual do sistema de maneira incremental para a solução alvo, através da interoperabilidade dos domínios de negócio e TI, como pode ser observado pelo modelo de ferradura do ADM na Figura 2.6 (KHUSIDMAN & ULRICH, 2007).

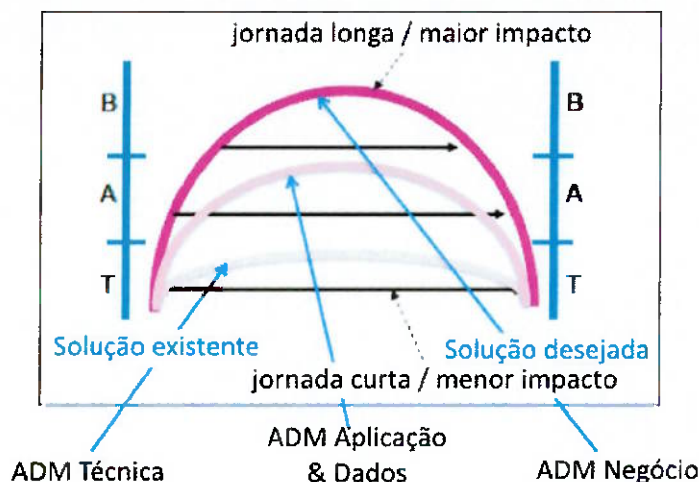
Cada componente do portfólio de TI evolui de forma independente do seu estado *as-is* para o estado *to-be*, mas é importante que os caminhos de transformação desses componentes estejam sincronizados, tanto verticalmente (implementação negócio-para-físico), quanto horizontalmente (existente-para-alvo). Dessa forma o modelo de ferradura do ADM representa, para uma dada estratégia de modernização, o conjunto desses caminhos que são combinados com os requisitos de negócio e os requisitos de TI. Esses diversos caminhos de transformação, independente do impacto arquitetural sofrido, apresentam três elementos em comum (KHUSIDMAN & ULRICH, 2007):

- Descoberta de conhecimento da solução existente: Ocorre em níveis diferentes de abstração e de escopo conforme as necessidades do projeto.
- Definição da arquitetura alvo: Para criar um método de transformação, analistas precisam criar uma solução alvo que serve como framework para que as soluções existentes possam ser modificadas e alteradas.
- Passos transformacionais que movem o estado *as-is* para o estado *to-be*, abrangendo desde o aspecto físico até aspectos mais abstratos.

2.2.1 Tipos de modernização para o Modelo Ferradura

O modelo ferradura, apresentado na Figura 2.6, representa o conjunto completo de caminhos possíveis para a modernização pelo ADM, incluindo a modernização de partes específicas do portfólio, que começam através do domínio de negócio e se deslocam através da arquitetura de dados e aplicativo e arquitetura técnica do domínio de TI ou retornam para o domínio de negócio. Na Figura 2.7 podemos observar os três tipos de modernização que podem ocorrer em projeto.

Figura 2.7 – Tipos de modernização orientado a arquitetura



Fonte: Adaptado de Khusidman & Ulrich, 2007

2.2.1.1 Modernização orientada para a Técnica

É o tipo de projeto mais comum de modernização e é aplicado nas organizações o tempo todo. É dirigido pelo risco devido a obsolescência de plataformas e linguagens, usabilidade do sistema, *cost of ownership*, ou outros fatores que são resolvidos através de uma mudança física. Como por exemplo, exercícios de refatoração: como reestruturação, racionalização da definição de dados, modularização ou troca da interface de usuário (KHUSIDMAN & ULRICH, 2007).

Existe uma linha bem tênue entre os projetos que ocorrem no nível técnico para projetos que transpassam para a camada arquitetural de dados e aplicação. E isso ocorre quando há um impacto a nível de aplicação, nível de sistema ou em fatores de desenho dos dados.

2.2.1.2 Modernização orientada para a Aplicação / Dados

A modernização desse tipo pode ser motivada por uma variedade de fatores e geralmente se estendem por múltiplas aplicações. Projetos nesse nível estão sendo realizados pelas organizações de TI, que nem sempre aplicam as disciplinas formais requeridas em um projeto de modernização, além disso, os modelos de interoperabilidade entre o MDA (Model-Driven Architecture) e outros padrões ainda está evoluindo.

Khusidman & Ulrich (2007), apresentam três exemplos, em cada um desses casos há um trabalho de análise e desenho envolvido, mas a arquitetura de TI ainda não está alinhada com uma arquitetura de negócio revisada:

- Um projeto para abstrair, redesenhar e reimplantar aplicações existentes em uma arquitetura orientada a modelos.
- Um projeto para modernização de múltiplas aplicações que não mais desempenham as necessidades do negócio através de atividades de manutenção incremental, ou envolver uma arquitetura de dados que está fora de sincronia com os requisitos de informações estratégicas por estar alinhado em torno de um modelo de aplicação obsoleto
- O redesenho e a transformação de múltiplas aplicações em uma aplicação comum que utiliza o redesenho de modelo de dados junto com a migração de plataforma, de tal maneira, a ir em direção a solução desejada.

2.2.1.3 Modernização orientada para o negócio

É a solução mais completa de modernização por incorporar os modelos arquiteturais de negócio, arquiteturas de aplicação e dados, e arquiteturas técnicas, alinhando formalmente a arquitetura de aplicação e dados com a arquitetura de negócio.

2.3 Modelo de referência RM-ODP

De acordo com Linington, Milosevic, Tanaka, & Vallecillo (2012), o objetivo do RM-ODP é prover um framework para especificar e construir sistemas grandes e complexos. Esses sistemas podem ser sistemas clássicos de TI, sistemas 'embedded', sistemas de informação, sistemas de negócio, ou qualquer outro sistema em que se está interessado.

Ainda de acordo com Linington, Milosevic, Tanaka, & Vallecillo (2012), em sistemas que são moderadamente complexos, é útil separar a descrição da sua estrutura e propriedades externas dos detalhes de seus componentes e subsistemas. Se essa visão abstrata do sistema é focada nos princípios gerais do sistema, é chamada de arquitetura, se apresentado de uma forma que é possível derivar um conjunto completo de sistemas futuros é chamado de framework.

A primeira versão do RM-ODP foi publicada, conjuntamente entre o ISO e o ITU-T, pela primeira vez nos meados da década de 1990. Dividida em 4 partes (introdução, um conjunto rigoroso de conceitos básicos, o framework arquitetural, e um link para suporte a técnicas formais) foi pensada para que designers de sistemas utilizassem o framework, mas também tem a intenção de ajudar pessoas que constroem ferramentas que suportem as atividades de desenho do processo, bem como as pessoas que produzem padrões de forma a capturar as melhores praticas e reutilização de mecanismos.

Por se tratar de um framework e não de uma metodologia, a ordem em que é aplicado o processo de desenho do sistema não é restrito pelo RM-ODP, mas sim pelos muitos processos de design disponíveis que podem ser utilizados com o framework, e como tipicamente projetos de sistemas ODP são grandes esforços colaborativos, o desenho do sistema acontece de forma iterativa, com a definição das especificações das diferentes partes sendo preenchidas conforme os requerimentos do sistema são melhor compreendidos (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

2.3.1 Pontos de Vistas

O RM-ODP parte do princípio de que a melhor aproximação para desenhar um sistema complexo é a de dividir as especificações do sistema em partes menores mas acopladas entre si. Dessa forma, os autores deste modelo de referência, garantiram que cada '*stakeholder*' seja atendidos por um dos pontos de vista, mantendo, dessa forma, a flexibilidade do projeto e evitando as complexidades associadas com a descrição da construção e manutenção de um sistema complexo (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Sendo os pontos de vistas interligados entre si, a união deles é o equivalente a representação completa e única de um sistema grande, mas essa notação única do sistema não é apresentada a nenhum dos '*stakeholder*', por ser grande e complexa de mais para ser útil.

Apesar de cada ponto de vista estarem interligadas, não é obrigatório que seja utilizado a mesma técnica de desenho para todos os pontos de vista. Cada stakeholder tem o conhecimento de linguagens e técnicas para a resolução dos seus interesses, e dessa forma as técnicas utilizadas serão diferentes. Linington, Milosevic, Tanaka, & Vallecillo (2012) ainda destaca que mesmo com essa

independência entre os pontos de vista, quanto maior a similaridade entre as técnicas utilizadas, mais fácil é a correspondência entre os diversos pontos de vista, e sugerem a busca de um equilíbrio entre a facilidade de solucionar um único ponto de vista contra a facilidade da integração entre os pontos-de-vista do sistema.

Além disso, sendo pontos de vista independentes, mas com certo nível de correlação, permitem que as equipes de desenvolvimento de cada ponto-de-vista trabalhem de forma paralela entre si, permitindo que alguns pontos do projeto cheguem a um nível de maturidade e estabilidade antes do que outros pontos-de-vista (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

A ideia de separar as preocupações do sistema através do uso de pontos-de-vista, podem ser utilizados por muitas atividades de projeto. Entretanto, quanto maior a aceitação do mesmo conjunto de pontos-de-vista de diversas equipes, maior é a chance dos pontos-de-vista serem reutilizados, por isso, os autores do ODP levaram em conta que para criar um padrão internacional é necessário o maior grau possível de comunalidade. Sendo assim, o RM-ODP especifica cinco pontos-de-vista que apelam a cinco grupos de usuários de famílias de padrões (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

2.3.1.1 Ponto de vista – Empresa

O ponto de vista de empresa, foca na situação organizacional em que a atividade do projeto será realizada. De acordo com Linington, Milosevic, Tanaka, & Vallecillo (2012), concentra seus esforços nas regras de negócio e as políticas que precisam ser suportadas pelo sistema sendo projetado. Portanto, os 'stakeholder' são os proprietários dos processos de negócios sendo suportados e os gerentes responsáveis por definir as políticas operacionais.

Linington, Milosevic, Tanaka, & Vallecillo (2012) destaca que a palavra enterprise é utilizada para cobrir qualquer área de interesse, pode ser qualquer coisa que os especificadores tenham sido nomeados a descrever. Pode ser um único produto e seu usuário, uma corporação, ou uma estrutura social maior que envolve diversas empresas.

2.3.1.2 Ponto de vista – Informação

O ponto de vista de informação, concentra na modelagem das informações que são compartilhadas dentro do escopo de interesse do empreendimento

(LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012), desconsiderando detalhes específicos de plataforma e implementação, bem como as interfaces computacionais, as funções usadas para manipular os dados e as tecnologias utilizadas para armazenamento desses dados.

Provendo um modelo comum, que pode ser referenciado ao longo do projeto inteiro, garantimos que a interpretação da informação será a mesma em todos os pontos do projeto. Como resultado, o desenvolvimento do projeto evita o uso divergente de informações incompletas que resultam de interpretações individuais de cada membro da equipe de projeto.

Linington, Milosevic, Tanaka, & Vallecillo (2012), concordam que o ideal seria ter uma fonte de informação universal e única, mas que esse ideal é claramente inviável, portanto recomendam almejar um modelo compartilhado para cada uma e cada atividade de projeto particular.

Assim, o ponto de informação deve responder uma série de questões sobre o sistema, como: “Quais os tipos de dados que o sistema irá suportar?” “Qual a relação entre esses tipos?” “Como o estado dos dados evolui dentro do sistema durante a operação do sistema?” “Quais as ações possíveis que o sistema irá aceitar, e como eles afetam o estado dos dados?” (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

2.3.1.3 Ponto de vista – Tecnologia

O ponto de vista da tecnologia, está preocupado com o gerenciamento das restrições do mundo real, ou seja, as limitações existentes no hardware disponível para a implantação do sistema dentro do orçamento disponível. O projetista desse ponto de vista raramente começa do zero e sempre terá como ponto de partida informações do parque computacional existente, as políticas de compra correntes e problemas de configuração.

O projetista dessa área, também está preocupado com a seleção de padrões onipresentes a serem utilizados no empreendimento, bem como a alocação e configuração dos recursos reais, ou seja, representa os componentes de hardware e o software do sistema implementado, bem como as tecnologias de comunicação entre estes componentes.

Linington, Milosevic, Tanaka, & Vallecillo (2012) enfatizam ainda, que esse ponto de vista tem um papel fundamental no gerenciamento de testes de conformidade

com a especificação geral do sistema, porque especifica a informação requerida dos implementadores para realizar esses testes.

2.3.1.4 Ponto de vista – Engenharia

O ponto de vista de engenharia, lida com o problema da diversidade de provisionamento da infraestrutura; oferecendo receitas para o suporte das interações computacionais abstratas necessárias em diferentes situações, evitando, dessa forma, ficar preso à uma especificação de plataforma específica ou mecanismos de infraestrutura específicos.

Esse ponto de vista também está preocupado com o provimento para os projetistas da visão computacional um conjunto de garantias, chamadas no modelo de transparências, ou seja, a equipe de projetistas desta visão tomam para si a responsabilidade para o problema da distribuição de modo que a equipe da visão computacional não precise se preocupar com esse problema (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

2.3.1.5 Ponto de vista – Computação

O ponto de vista computacional, de acordo com Linington, Milosevic, Tanaka, & Vallecillo (2012), preocupa-se com o desenvolvimento em alto nível dos processos e aplicações que suportam as atividades do empreendimento. Se utiliza de ferramentas de software orientado-à-objetos, para expressar seus modelos em termos de objetos com forte encapsulamento, interagindo através de interfaces com operações sequenciadas ou com um fluxo contínuo de informação. A especificação deste ponto de vista referencia o ponto de vista de informação em busca dos objetos de dados e suas restrições comportamentais.

Linington, Milosevic, Tanaka, & Vallecillo (2012) chama atenção, para o fato, desse ponto-de-vista ser abstrato, ou seja, não é definido no contexto do ODP onde é colocado os objetos e nem onde são alocados recursos computacionais.

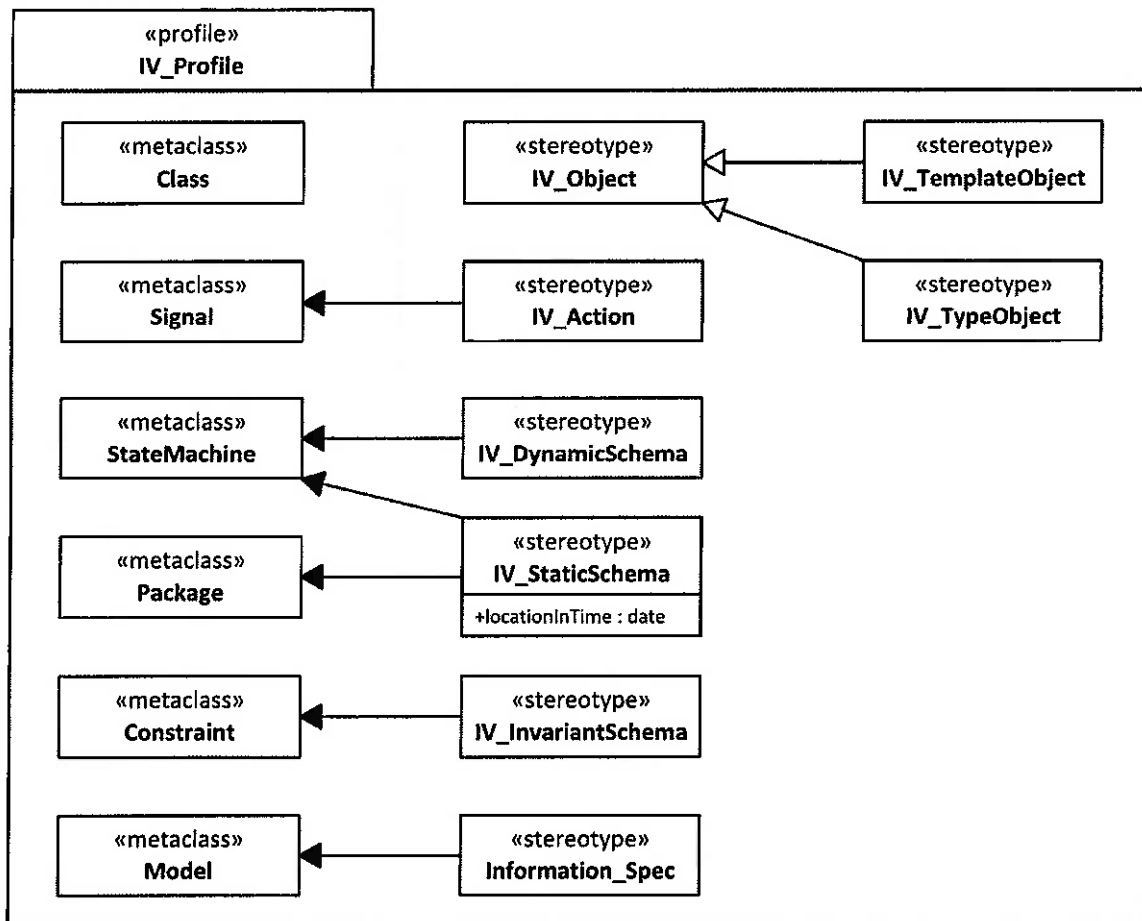
2.3.2 UML4ODP

O padrão UML4ODP prove tanto uma notação baseada na linguagem UML para expressar um sistema ODP quanto uma abordagem para a estruturação do sistema. Isso torna o UML4ODP útil para modeladores ODP que querem utilizar o UML para descrever o sistema, mas também para modeladores UML que precisam lidar com o

desenho de sistemas complexos (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Esta notação utiliza os conceitos padrões do UML e conta com os mecanismos de extensão do UML para definir novas linguagens e especialmente perfis UML, dessa forma o UML4ODP define sete novos perfis UML um para cada ponto de vista do ODP, i.e. (Empresa, Informação, Computação, Engenharia e Tecnologia), um para descrever as correspondências e outro para modelar a conformidade com a especificação de sistemas ODP (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012), como pode ser visto na Figura 2.8, o perfil UML para o ponto de vista informação.

Figura 2.8 – O perfil UML4ODP para o ponto de vista informação.

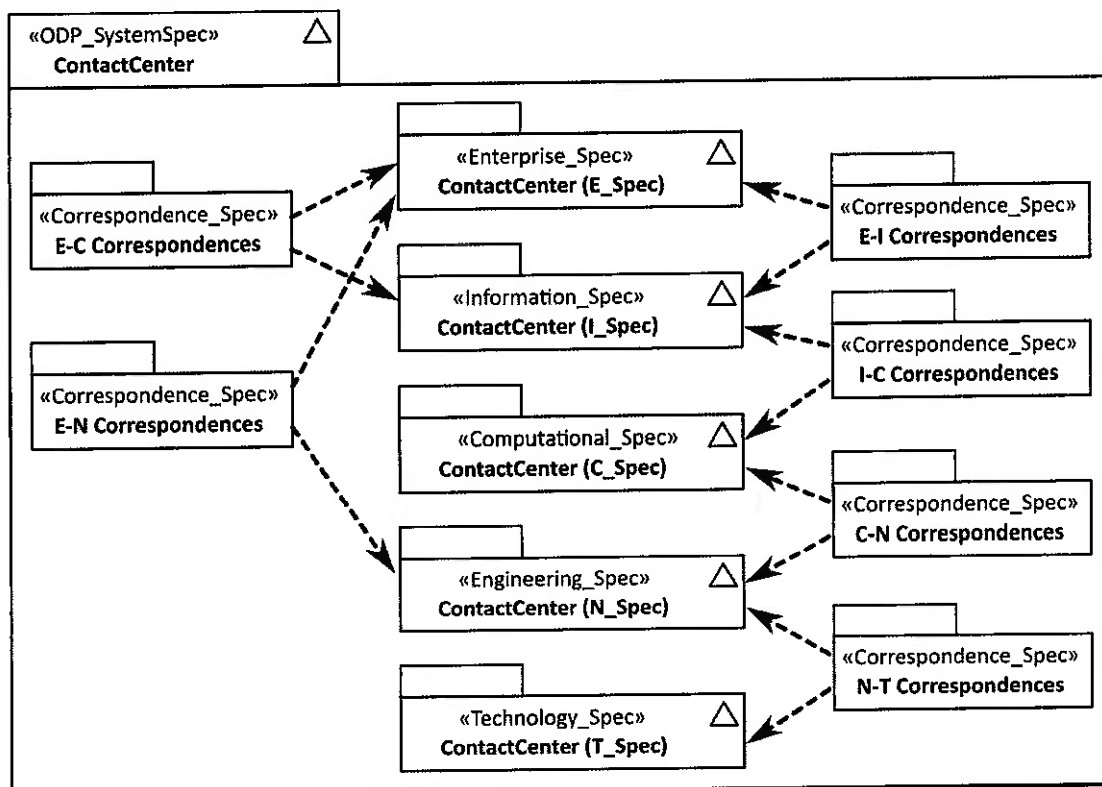


Fonte: Linington, Milosevic, Tanaka, & Vallecillo, 2012

Na Figura 2.9 observamos a especificação de um sistema ODP em UML4ODP que consiste de um único modelo UML estereotipado como «ODP_SystemSpec» contendo o conjunto de modelos, um para cada ponto de vista estereotipado como «X_Spec», onde <X> é o ponto de vista em foco. Por sua vez, a especificação de

cada ponto de vista utiliza o perfil UML apropriado, onde estereótipos são utilizados para representar os conceitos ODP como especializações das metaclasses UML apropriadas como podem ser vistos na Figura 2.9 para a proposta de Contact Center desenvolvido nesse trabalho (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Figura 2.9 – Especificação geral do sistema do contact center em UML4ODP



Fonte: Autor

3 Transformação arquitetural com o uso do RM-ODP e UML4ODP

Este trabalho, busca realizar a atualização da arquitetura aberta de central de atendimento proposta por Gonçalves (2001) para o contexto atual de *contact center*. Para isso, é utilizado o *Architectural-Driven Modernization* como ferramenta de transformação arquitetural do modelo *as-is* para o modelo *to-be*. Com isso, o primeiro passo desse trabalho é o levantamento e mapeamento da central de atendimento existente para um modelo '*as-is*', que será utilizado como base referencial para o restante do projeto.

Foi escolhido como framework para o desenvolvimento deste trabalho o modelo de referencia RM-ODP, por ser um modelo que simplifica o desenvolvimento de sistemas complexos, e do uso da linguagem UML4ODP, para auxiliar na notação da arquitetura proposta.

A modelagem da arquitetura aberta de *contact center*, segue a metodologia utilizada por (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012) em seu livro e descrita no item 2.3, iniciando a modelagem com o ponto de vista empresa, seguindo pelo ponto de vista informação e concluindo, neste trabalho, com o ponto de vista computação, de forma iterativa e com o desenvolvimento das respectivas notações em UML4ODP. Os outros dois pontos de vista do RM-ODP, engenharia e tecnologia, não serão abordados neste trabalho.

É importante notar, que na metodologia proposta por (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012) é fundamental iniciar a modelagem pelo ponto de vista empresa com o intuito de garantir que os *stakeholders* do negócio e arquitetos do sistema tenham um entendimento comum e compartilhado do que o sistema deva cumprir. É essencial que a definição do sistema ODP tenha os seus objetivos bem definidos no ponto de vista empresa entre todas as partes interessadas para que o desenvolvimento do sistema ocorra de forma linear e eficiente (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012), (PUTMAN, 2000).

3.1 Contexto do *contact center*

O *contact center* a ser desenvolvido neste trabalho está inserido dentro de uma corporação que presta serviços de terceirização do gerenciamento de carteira de revista e periódicos, eventos e seguros, em uma plataforma online, oferecendo aos seus clientes:

- Gestão de vendas
- Gestão de cobranças
- Processos de agendamento e controle de cobranças recorrentes
- Gestão de distribuição
- Validação de acesso em tablets e sites
- Comunicações de relacionamento com o cliente
- Relatórios gerenciais, controles financeiros e contábeis online
- Múltiplas plataformas de renovação e recuperação dos clientes
- Atendimento ao Assinante (SAC) por meio eletrônico ou telefônico

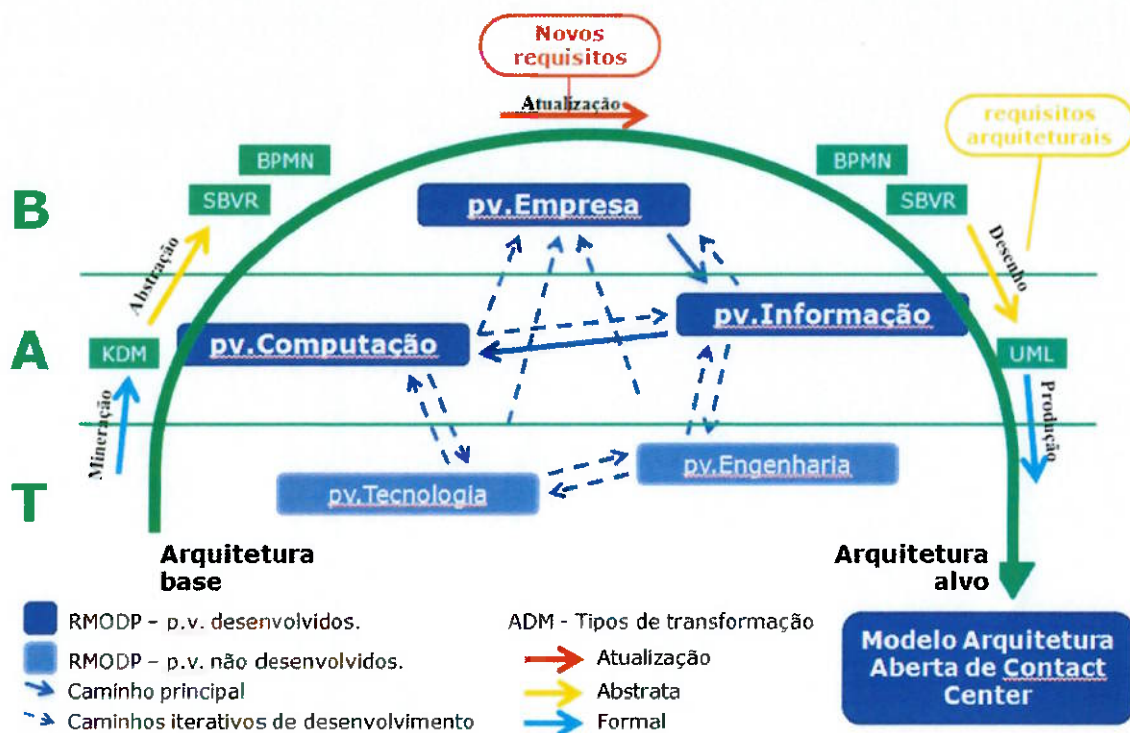
Neste trabalho a arquitetura do *contact center* deve ajudar a corporação a cumprir com as metas acima de forma precisa e eficiente. Para simplificação do desenvolvimento do trabalho será somente abordado o gerenciamento de carteira de revistas e periódicos.

3.2 Metodologia de transformação arquitetural com ADM e RM-ODP

A metodologia a ser aplicada para obter o modelo 'to-be' de *Contact Center* é demonstrando em sua plenitude na Figura 3.1, e pode ser dividido em duas fases distintas: A primeira fase é o caminho de descoberta e mapeamento da arquitetura base, partindo do nível tecnológico, passando pela arquitetura de dados e aplicação e finalizando com o levantamento da arquitetura de negócio da central de atendimento.

Com as informações da arquitetura base mapeadas e compreendidas, iniciamos a segunda fase, onde por um processo de subtração retiramos dos novos requisitos os já contemplados na arquitetura 'as-is'.e tomamos a direção contrária da primeira fase do ADM, iniciando o desenvolvimento da nova arquitetura pela Arquitetura de negócio e o ponto de vista empresa, seguindo para a camada de arquitetural de dados e aplicação, onde são especificados os pontos de vista computação e informação do RM-ODP e finalizando na arquitetura tecnológica com o especificação dos pontos de vista tecnologia e engenharia.

Figura 3.1 – Transformação arquitetural com ADM e RM-ODP



Fonte: Autor

3.2.1 Levantamento da arquitetura 'as-is' (1ª fase)

O primeiro passo para transformar uma arquitetura existente é realizar o levantamento das tecnologias e da infraestrutura de engenharia do modelo existente através da mineração da arquitetura base resultando no KDM (Knowledge Discovery Metamodel), ou seja, uma representação intermediária dos sistemas de software e seus ambientes operacionais.

Na sequência é realizado a abstração da arquitetura de dados e computação, junto com os resultados da mineração para obter o SBVR (Semantics of Business Vocabulary and Business Rules), ou seja, a descrição do negócio como um todo. Com essas informações é possível gerar os diagramas BPMN (Business Process Model Notation) da arquitetura existente ('as-is').

3.2.2 Construção do modelo 'to-be' (2ª fase)

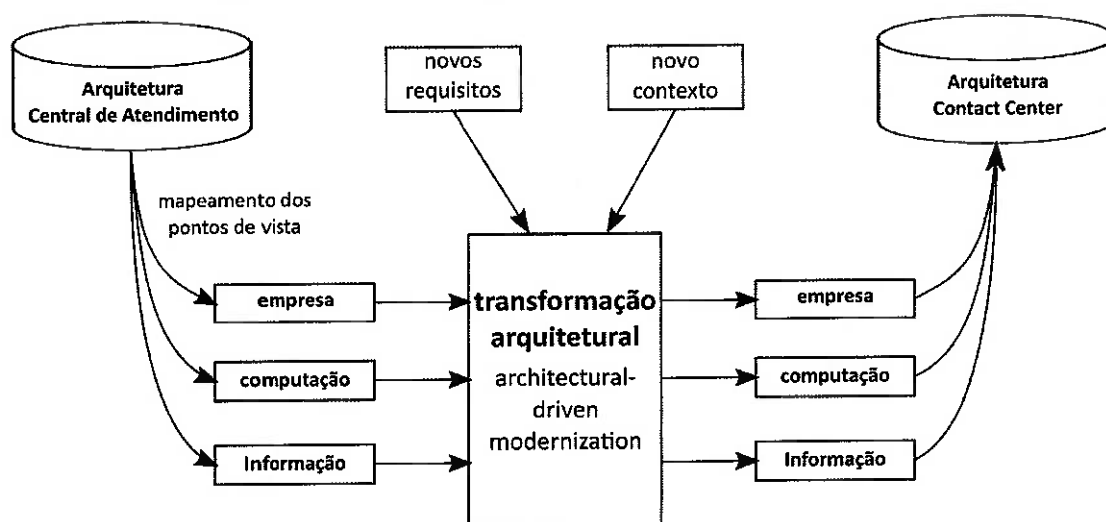
Ainda no domínio de negócio do ADM, e com o modelo 'as-is' bem definido, passamos a fase de atualização do modelo arquitetural para a arquitetura desejada ('to-be') realizando as atualizações necessárias no ponto de vista empresa, a partir

do contexto onde será inserido o novo modelo arquitetural do *Contact Center*, retiramos os novos requisitos funcionais e não funcionais, tomando em consideração os requisitos já identificados no modelo arquitetural 'as-is'.

Com os novos requisitos e contextos incorporados ao modelo arquitetural atualizamos os diagramas do BPMN, o vocabulário e regras do negócio (SBVR). Essas especificações do negócio garantem, como escrito anteriormente, que *stakeholders* e arquitetos do sistema tenham uma visão comum e compartilhada do objetivo a ser alcançado.

Na sequência, partimos para o desenho do sistema, onde acrescentamos os requisitos arquiteturais e é especificado em linguagem UML o comportamento do sistema.

Figura 3.2 – Metodologia de transformação arquitetural



Fonte: Autor

3.2.3 Mapeamento da arquitetura aberta para o ODP

O primeiro passo, para alcançarmos o nosso objetivo é a extração da arquitetura aberta do modelo de Central de Atendimento proposto por Gonçalves (2001), realizando o mapeamento dos elementos do sistema para o RM-ODP, conforme pode ser observado na Figura 3.2.

Tabela 1 - Mapeamento da arquitetura modelo (Gonçalves, 2001) para os pontos de vista RM-ODP

Ponto de vista (RM-ODP)	Diagramas da Arquitetura modelo de Gonçalves (2001) ¹	Diagramas da Arquitetura Alvo em UML4ODP
Empresa	<ul style="list-style-type: none"> • Modelagem Geral do Processo de Negócio da Central de Atendimento. • SAA² : Nível Corporativo 	<ul style="list-style-type: none"> • Casos de Uso • Diagrama de Categoria • Diagrama de Sequencia (Categoria)
Informação	<ul style="list-style-type: none"> • Diagrama de classes da Central de Atendimento • SAA : Nível de Coordenação 	<ul style="list-style-type: none"> • Diagrama de Classes UML (Visão Estática)
Computação	<ul style="list-style-type: none"> • Ciclo de vida de aplicações CTI <i>First-Party</i> e <i>Third-Party</i> • SAA : Nível de Controle 	<ul style="list-style-type: none"> • Diagrama de Sequencia UML (Visão Dinamica) • Diagrama de Estado

Fonte: Autor

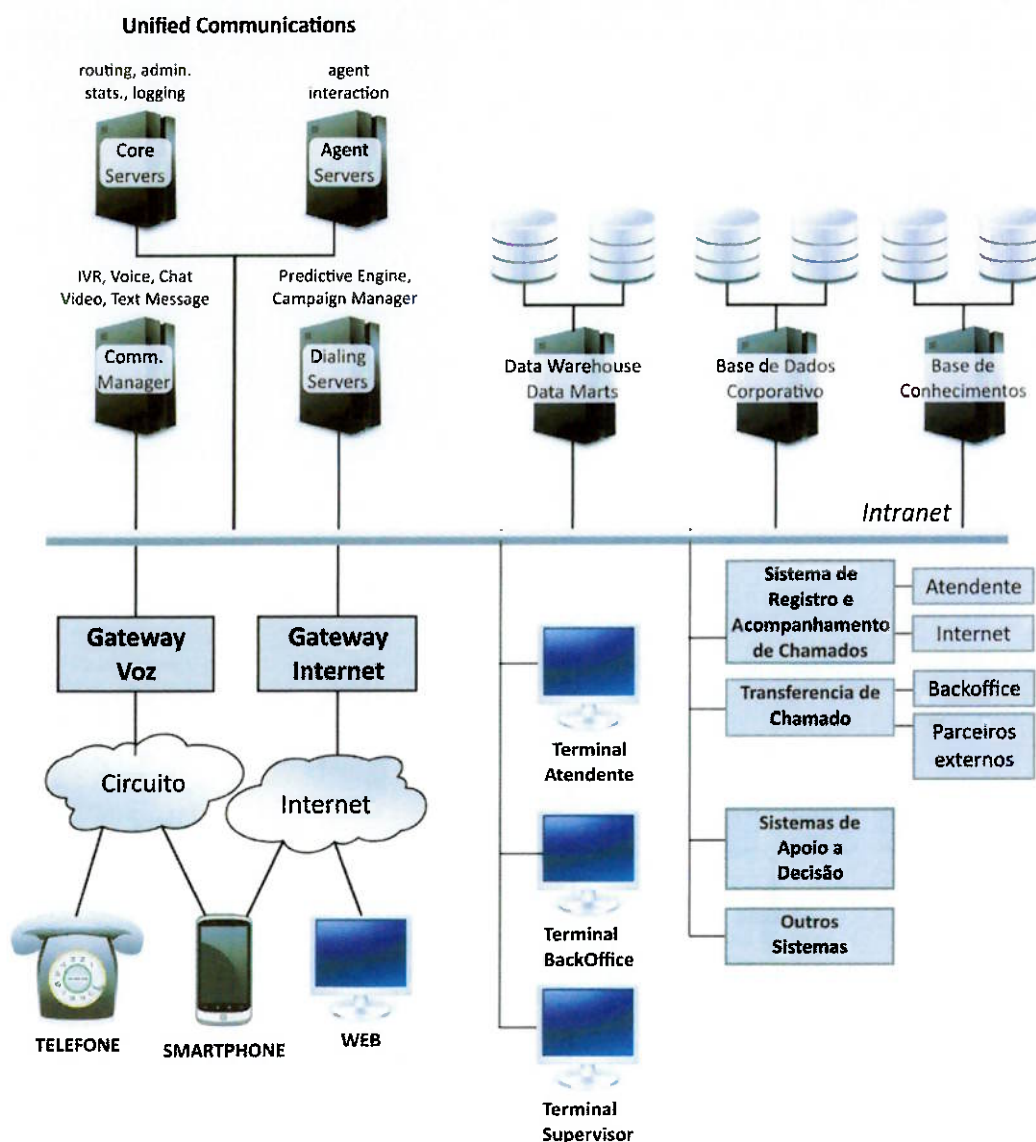
3.3 Arquitetura do contact center

A arquitetura do *contact center* desenvolvida neste trabalho, apresenta uma serie de atualizações tecnológicas de software, hardware e telefonia e a adaptação para o contexto em que se irá implantar a arquitetura proposta (item 3.1) , além dessas atualizações buscou se a otimização de alguns conceitos do modelo base de Gonçalves para o modelo proposto de *contact center*, como pode ser observado na Figura 3.3

¹ As descrições dos diagramas aqui listados seguem as descrições e referencias conforme apresentadas no referido trabalho.

² SAA: Sistema Aberto de Automação

Figura 3.3 – Proposta de Arquitetura Aberta de contact center



Fonte: Autor

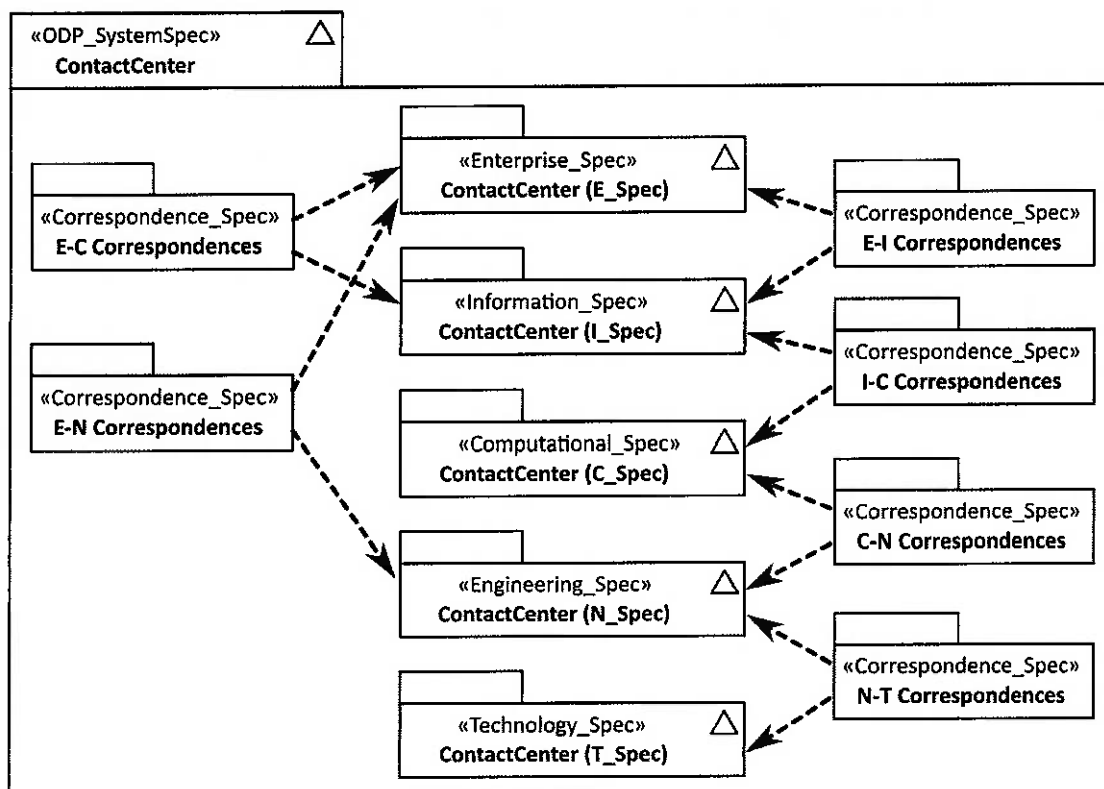
3.4 Modelagem RM-ODP

Foi escolhido o modelo de referencia RM-ODP para uso na especificação da arquitetura de uma central de atendimento por se tratar de um modelo que visa a simplificação do projeto de sistemas de informação complexos em subpartes independentes, mas interligadas entre si por pontos de conformidade, denominados pontos-de-vista (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Como descrito no capítulo de fundamentação conceitual, o modelo ODP define, de acordo com Linington, Milosevic, Tanaka, & Vallecillo, (2012), um framework de

trabalho e não uma metodologia, portanto a ordem que se aplica os pontos-de-vista fica à critério dos Arquitetos do Sistema, entretanto, uma pratica comum é iniciar o desenvolvimento do sistema pela especificação do ponto de vista empresa e na sequencia preparar as especificações do pontos-de-vista informação e computação, e finalmente os pontos-de-vista engenharia e tecnologia.

Figura 3.4 – Especificação UML4ODP para o contact center



Fonte: Autor

As especificações em cada ponto-de-vista podem colocar restrições nos demais pontos de vista do modelo ODP que são revisados e refinados conforme cada ponto de vista é desenvolvido num processo iterativo e contínuo pela busca de maturidade do sistema. Na Figura 3.4 observamos a contribuição da divisão em pontos-de-vista do RM-ODP e os pontos-de-correspondência para a especificação do sistema da central de atendimento.

3.4.1 Ponto de vista empresa

O primeiro ponto a ser levantado antes do desenvolvimento do sistema ODP é os requisitos e as restrições do negócio, que são conhecidos como os requisitos

funcionais e não funcionais do sistema (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Os requisitos funcionais e não funcionais utilizados no nosso modelo ODP foram baseados no contexto e novos requisitos em que está inserido o *contact center* (item 3.1), na arquitetura proposta de *contact center* (item 0), e nos elementos necessários do *contact center* expostos no item 2.1.

Figura 3.5 – Alguns dos requisitos identificados na Arquitetura Aberta de Central de Atendimento proposta por (GONÇALVES, 2001) e dos novos requisitos

item	Requisito	Ponto de Vista									
		EV	Tipo	IV	Tipo	CV	Tipo	NV	Tipo	TV	Tipo
1	O Sistema deve ajudar a satisfazer e fidelizar o cliente.	✓	RNF	✗	-	✗	-	✗	-	✗	-
3	O Sistema deve fornecer serviços a um custo compatível com o contexto corporativo em que será inserido.	✓	RNF	✗	-	✗	-	✗	-	✗	-
5	O Sistema deve permitir que um Cliente possa obter informações do banco de dados corporativos de forma autônoma dos seus dados particulares, produtos e chamados, desde que autenticado e autorizado pelo	✗	-	✓	RF	✗	-	✓	RF	✓	RF
7	O Sistema deve permitir a transferência da ligação do pré-atendimento (ura) a um atendente através da fila de atendimento.	✗	-	✓	RF	✗	-	✓	RF	✓	RF
13	O Sistema deve apresentar ao atendente um roteiro de atendimento, em forma de questionário, montado de acordo com o perfil do cliente, de forma a agilizar e otimizar o atendimento.	✗	-	✗	-	✓	RF	✓	RNF	✓	RNF
26	O Sistema deve informar a estimativa de tempo de espera aos diversos usuários da central de atendimento.	✗	-	✓	RF	✓	RF	✓	RNF	✓	RNF
27	O Sistema deve categorizar os clientes em perfis através da análise dos dados obtidos e armazenados no banco de dados.	✓	RNF	✓	RNF	✓	RF	✓	RNF	✓	RNF
28	O Sistema deve continuar o atendimento com um cliente em particular do ponto em que houve o último contato com ele, independente do canal de comunicação utilizado.	✓	RF	✓	RF	✗	-	✓	RNF	✓	RNF
41	O Sistema deve oferecer suporte aos diversos protocolos e APIs de telefonia disponíveis no mercado.	✗	-	✗	-	✗	-	✓	RF	✓	RF
42	O Sistema deve oferecer suporte aos diversos protocolos de rede abertos no mercado, independentemente da camada OSI.	✗	-	✗	-	✗	-	✓	RF	✓	RNF
53	O Sistema deve monitorar o comportamento dos clientes, da marca e de seus produtos nos sites de mídias sociais.	✗	-	✓	RF	✓	RF	✓	RF	✓	RF

Legenda: (EV) Ponto de vista Empresa, (IV) Ponto de vista Informação, (CV) Ponto de vista Computação, (NV) Ponto de vista Engenharia, (TV) Ponto de vista Tecnologia | (RF) Requisitos Funcionais, (RNF) Requisitos Não Funcionais.

Fonte: Autor

Através da leitura e análise da proposta de arquitetura aberta de central de atendimento proposta por (GONÇALVES, 2001) conseguimos identificar os requisitos funcionais e não-funcionais para uma central de atendimento, e acrescentar os novos requisitos do *contact center* conforme pode ser observado na Figura 3.5 alguns desses requisitos, para a relação completa consulte o anexo ao final do trabalho.

Com a relação dos requisitos identificados realizamos o mapeamento desses requisitos com os cinco pontos de vista do RM-ODP. Notadamente nem todos os requisitos pertencem a todos os Pontos de Vista do Modelo ODP, alguns requisitos, como por exemplo o requisito de número 5, que identifica a necessidade do sistema

prover meios para que o cliente consulte as suas informações pessoais, informações sobre chamados e de produtos, pertence aos ponto de vistas responsáveis por armazenar e disponibilizar esses dados: informação, engenharia e tecnologia, mas por não manipular e transformar os dados, não pertence ao Ponto de Vista Computação.

Da mesma forma alguns requisitos dizem como o sistema deve se comportar, descrevendo suas funcionalidades e serviços, sendo estes classificados como requisitos funcionais do sistema, outros requisitos, entretanto, impõem regras e restrições aos processos ou ao sistema como um todo, e estes são classificados como requisitos não funcionais.

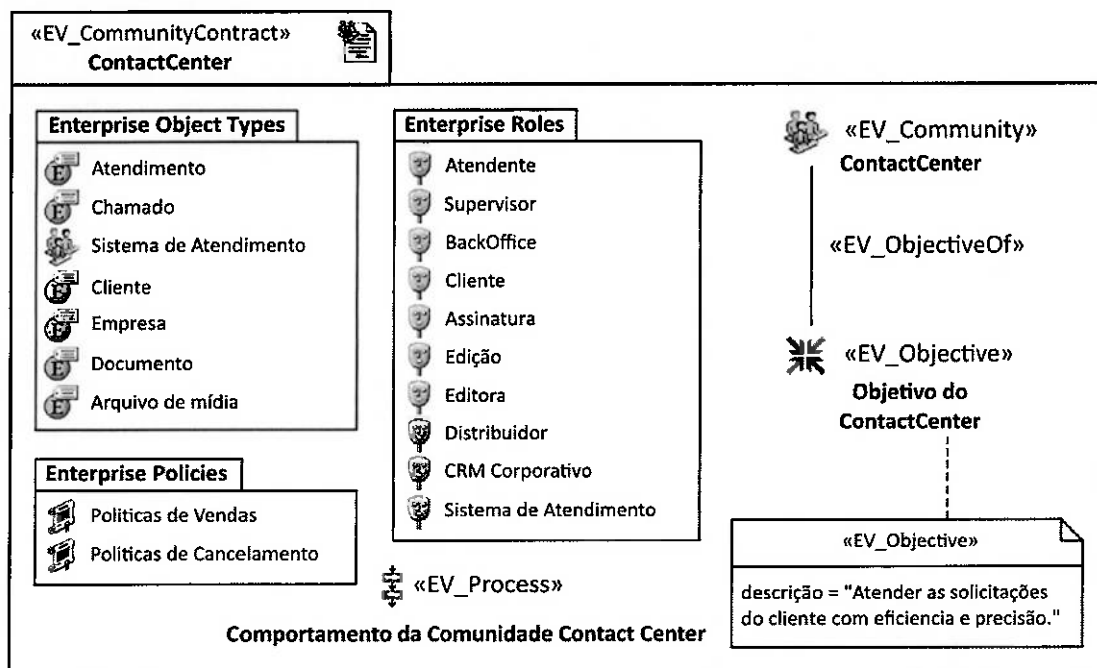
A partir da definição dos requisitos funcionais, extraídos da proposta de Gonçalves (2001) e complementados com os requisitos adicionais do novo contexto a ser implementado, começamos a modelagem ODP do sistema através do ponto de vista empresa. Este ponto de vista, conforme visto no item 2.3.1.1, observa como a corporação alvo do sistema está organizado, e através do comportamento observado procura definir todos os objetos pertencentes ao sistema, suas responsabilidades e funcionalidades. Dessa forma determinamos em alto nível de abstração, e em linguagem comum e de fácil compreensão a executivos, stakeholders e desenvolvedores um entendimento comum do que o sistema deve realizar.

Esse ponto-de-vista adota um principio estrutural mais flexível, pois processos de negócio são mais flexíveis e menos claros do que processos de software. O modo correto de um sistema de negócio operar, muitas vezes, é determinado pela quantidade de conjunto de regras que se sobrepõe; Algumas restrições vem do próprio negócio, outras de normas organizacionais, de acordos com parceiros comerciais, ou até mesmo de restrições legais (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Esses conjuntos de regras precisam ser fundidos e para isso é necessário basear a especificação deste ponto-de-vista em um conjunto inter-relacionado de *comunidades*. Cada comunidade possui um *contrato* que descreve as regras de negociação em que a comunidade opera, quais seus objetivos, o seu comportamento e papeis dentro do sistema, e as politicas e objetos locais (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Uma comunidade define no seu escopo como um grupo de participantes deve se comportar para atingir um determinado objetivo. As regras são expressas em forma de interações entre papéis dentro da comunidade de forma que a sua definição fique separada dos recursos disponíveis e das responsabilidades em efeito em um determinado período no tempo, como podemos observar na Figura 3.6 (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Figura 3.6 - Especificação da Comunidade contact center

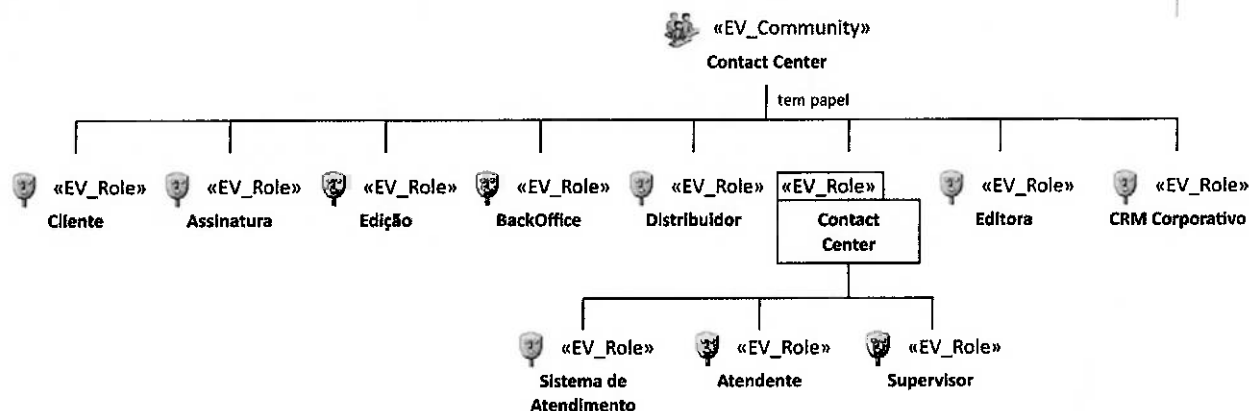


Fonte: Autor

Dessa forma, para a elaboração da arquitetura do contact center em questão, partimos do modelo de arquitetura aberta proposta por Gonçalves (2001), e dos componentes pertencentes ao nível corporativo do Sistema Aberto de Automação, que é composto do CRM corporativo e do sistema de gerenciamento de conhecimento da corporação; e levando em consideração o contexto e os requisitos funcionais e não funcionais exigidos por esse novo contexto, é realizada a primeira transformação arquitetural do modelo de Central de Atendimento para o modelo de *contact center* no ponto de vista empresa, especificando o seu 'contrato de comunidade', Figura 3.6. Neste diagrama é possível observar todo o escopo de atuação do contact center, seus objetos internos, seus papéis e relacionamentos externos, suas políticas e regras de conduta, seus processos e sua missão. Cada um desses elementos da comunidade será detalhado nos parágrafos seguintes.

Com a especificação do conjunto global da comunidade, e seguindo com a estratégia de transformação arquitetural, iniciamos o detalhamento do ponto de vista empresa.

Figura 3.7 - A Estrutura Organizacional da Comunidade do Contact Center



Fonte: Autor

Na Figura 3.7 é especificado a comunidade do contact center, ou seja, os participantes da comunidade descritos pelos papéis que exercem dentro da comunidade, a primeira camada da estrutura organizacional descreve o posicionamento como negócio do *contact center* no ambiente em que opera, observa-se que o *contact center* como um todo está representada como um papel nessa camada junto com papéis para seus principais 'players' externos com qual o *contact center* se relaciona. Refinando o papel *contact center*, temos os seus papéis participantes, dessa maneira temos os seguintes papéis no sistema:

- **Papel Cliente:** Representa a pessoa física ou jurídica que tem um relacionamento contratual com a editora, e é o responsável legal pelo pagamento das contas.
- **Papel Assinatura:** Representa a assinatura do periódico do Cliente sendo atendido.
- **Papel Edição:** Representa uma única edição que faz parte da assinatura do cliente.
- **Papel BackOffice:** Representa o departamento corporativo responsável por, além das atividades usuais de processamento e atendimento da carteira de assinantes, analisar e resolver as solicitações não resolvidas pelo *contact center*.

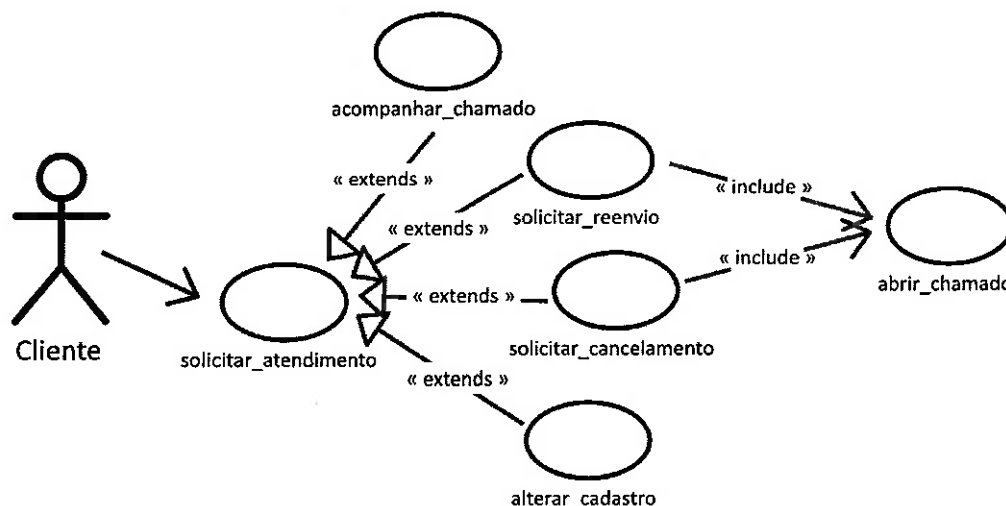
- Papel Distribuidor: é outra comunidade, podendo ser um departamento dentro da empresa ou terceirizado responsável pela distribuição das revistas e periódicos.
- Papel Editora: Representa a empresa cliente do *contact center*.
- Papel CRM Corporativo: representa o sistema de CRM da empresa como um todo. incluindo Data Warehouses, Data Marts, Data Mining, Bases de Conhecimento, Banco de Dados Corporativos, Sistemas de Apoio a Decisão entre outros sistemas que sejam necessários ao cumprimento da missão da empresa.
- Papel BackOffice: representa uma equipe de especialistas que tem conhecimento técnico para resolução de chamados mais específicos.
- Contact center: descreve o foco principal deste trabalho que é a unidade corporativa responsável pelo contato com o cliente, e contem os seguintes papeis:
 - Papel Atendente: é a pessoa responsável por interagir com o cliente no primeiro contato para resolução e acompanhamento dos chamados em andamento. Podendo ser o atendimento escalonado para um nível de atendimento especializado.
 - Papel Supervisor: é a pessoa ou grupo de pessoas responsável por monitorar e garantir a qualidade dos serviços prestados pelos atendentes e especialistas da Central de Atendimento.
 - Papel Sistema de Atendimento: Representa o sistema de atendimento ao cliente do *contact center*.
 - Papel CTI: representa o equipamento responsável por integrar o sistema telefônico com o sistema computacional da empresa, além disso é responsável pelas interfaces por voz da URA, pelo gerenciamento das filas de distribuição automática de chamadas, pelos sistemas de monitoramento e sistemas de estatísticas do *contact center*.

Através da definição dos papeis que interagem dentro da comunidade do Contact Center, definimos os casos de usos que cada papel está submetido, conforme podemos ver nas Mas o comportamento de cada comunidade não diz respeito somente as ações executadas pelos atores de cada papel, pois os papeis estão preocupados em conectar os comportamentos desta comunidade com uma visão

mais abrangente do sistema, a comunidade pode incluir objetos empresa adicionais que se preocupam somente com sua especificação interna e, portanto, tem o escopo local (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012). Esse objetos poderiam ser declarados como papéis, mas isso tornaria a visão externa da comunidade mais complexa sem trazer nenhum benefício extra a comunidade e ao sistema.

Figura 3.8, Figura 3.9 e Figura 3.10. Mas o comportamento de cada comunidade não diz respeito somente as ações executadas pelos atores de cada papel, pois os papéis estão preocupados em conectar os comportamentos desta comunidade com uma visão mais abrangente do sistema, a comunidade pode incluir objetos empresa adicionais que se preocupam somente com sua especificação interna e, portanto, tem o escopo local (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012). Esse objetos poderiam ser declarados como papéis, mas isso tornaria a visão externa da comunidade mais complexa sem trazer nenhum benefício extra a comunidade e ao sistema.

Figura 3.8 – Diagrama de Casos de Uso para o Papel Cliente



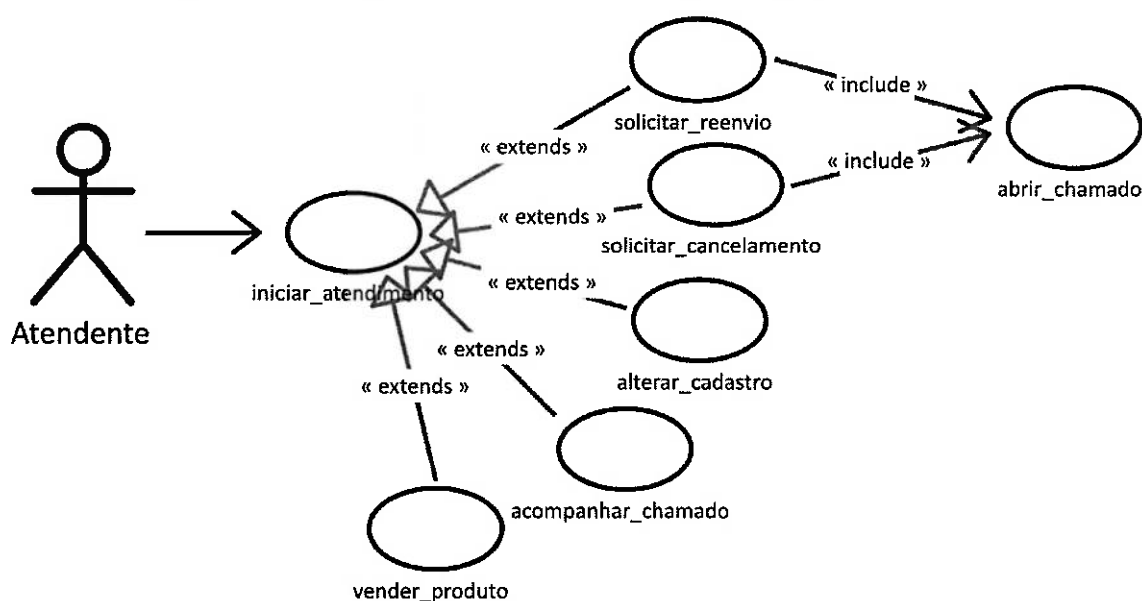
Fonte: Autor

Na Figura 3.8, podemos observar os possíveis casos de uso para o papel Cliente dentro da comunidade do ponto de vista empresa do *contact center*. Esses casos de usos são expressos independente do canal de comunicação utilizado, canais estes que são definidos no ponto de vista tecnologia (que não será abordado neste trabalho), e dependendo do canal utilizado pelo cliente nem todos os casos de uso estarão disponíveis para o Cliente, por exemplo: O Cliente pode fazer a solicitação

de cancelamento de uma assinatura e o acompanhamento desse chamado pessoalmente pela Internet, mas devido as limitações de uso do meio telefônico, essa operação necessita do auxílio de um atendente.

Na Figura 3.9, observamos os casos de uso para o papel Atendente, que compartilha dos mesmos casos de usos do papel cliente com a adição da possibilidade de realização de venda de produtos ao cliente, seja de uma assinatura, ou de uma única edição. É importante notar que nos casos de uso do cliente (Figura 3.8), a possibilidade de compra de uma assinatura pelo papel cliente pode ocorrer dentro da comunidade do *contact center* somente com o auxílio de um atendente, a aquisição através de site na Internet ocorre diretamente dentro do CRM corporativo, que não é parte integrante do *contact center*, mas faz parte da comunidade porque há o relacionamento entre o *contact center* e o CRM.

Figura 3.9 – Diagrama de Casos de Uso para o papel Atendente



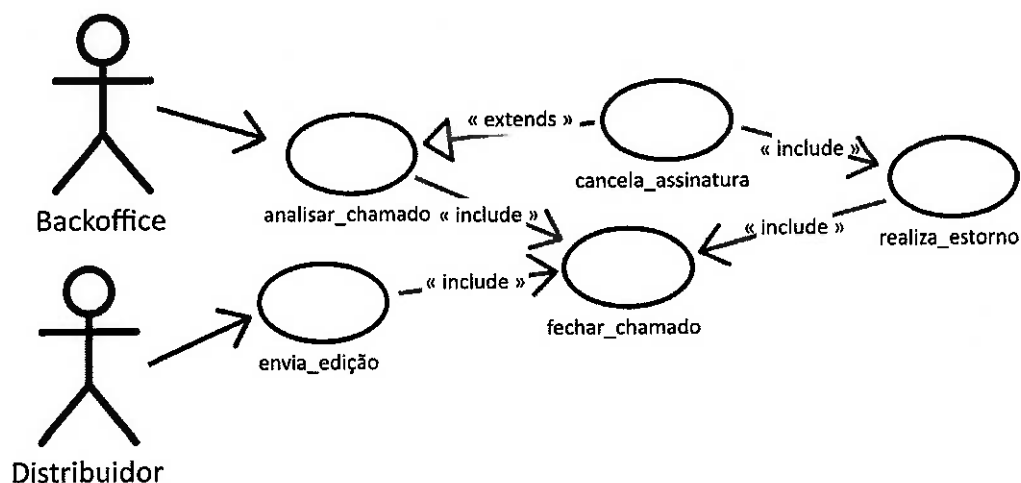
Fonte: Autor

E finalmente na Figura 3.10, observamos os casos de usos para os papéis BackOffice e Distribuidor, que recebem os chamados abertos pelos papéis Cliente e Atendente e agem de acordo com suas responsabilidades dentro do sistema.

Mas o comportamento de cada comunidade não diz respeito somente as ações executadas pelos atores de cada papel, pois os papéis estão preocupados em conectar os comportamentos desta comunidade com uma visão mais abrangente do sistema, a comunidade pode incluir *objetos empresa* adicionais que se preocupam

somente com sua especificação interna e, portanto, tem o escopo local (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012). Esse objetos poderiam ser declarados como papeis, mas isso tornaria a visão externa da comunidade mais complexa sem trazer nenhum benefício extra a comunidade e ao sistema.

Figura 3.10 – Diagrama de Casos de Usos para os Atores BackOffice e Distribuidor



Fonte: Autor

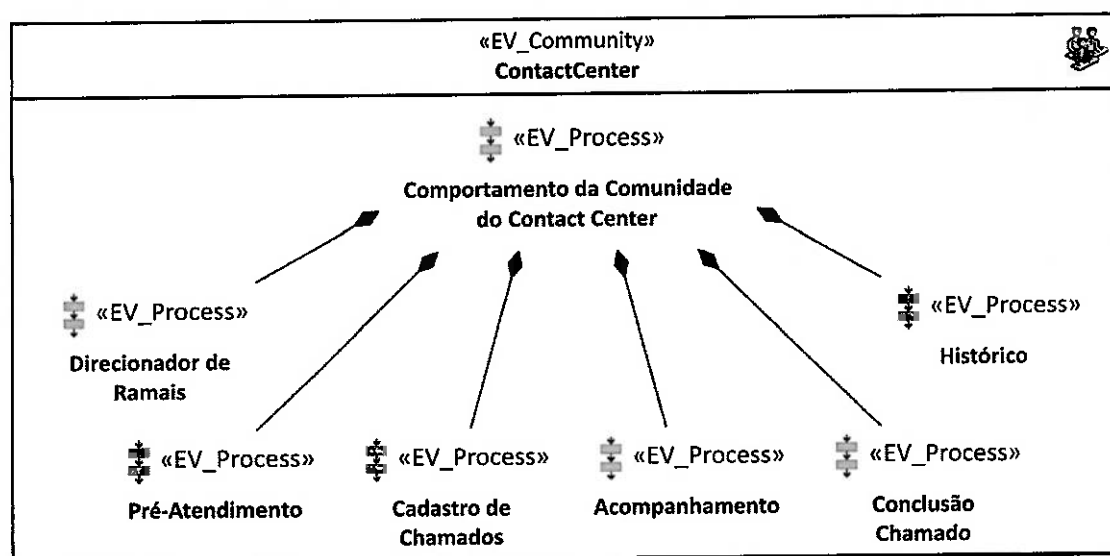
Papeis ou objetos empresa locais podem estar envolvidos no comportamento da comunidade de várias formas. Uma delas é a distinção permitida pela linguagem do ponto de vista empresa do modo como os objetos empresa participam em interações no sistema: Se o objeto empresa participa executando uma ação, ele é denominado de ator. Se o objeto empresa é essencial para a execução de uma ação, requerendo sua alocação ou consumo, então é denominado como recurso. Se o objeto empresa é mencionado em uma interação, mas não é um participante ativo desta, então, para está ação, esse objeto é denominado artefato (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Outra maneira, é através da descrição do comportamento por serviços de negócios, que foca na funcionalidade ou capacidade provida por um participante para outros participantes que podem utilizar os serviços para satisfazer suas necessidades de negócio. Esses serviços podem ser providos por um único papel na comunidade ou pode envolver diversos papeis, também podem ser declarados publicamente pela comunidade a que pertencem, sendo disponibilizados para comportamentos de comunidades diferentes. Os serviços de negócios também

podem ser suportados por um ou mais serviços técnicos, descritos no ponto de vista computacional (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Um processo é uma forma específica de comportamento expressa de forma sequencial ou ordenada em passos, cada um desses passos consome e produz informação, e dessa forma o processo envolve um ou mais papéis da comunidade que realizam ações associadas aos passos do processo. Cada processo almeja a satisfação de um objetivo, e em muitos processos podem ser definidos vários processos cada qual com o seu objetivo específico (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Figura 3.11 – Os processos do contact center descritos como comportamentos



Fonte: Autor

Interações são os elementos básicos de comportamento da comunidade, onde cada interação indica quais os papéis envolvidos, e qual parte atua nele. Múltiplas interações podem ser compostas em sequencias ou atividades concorrentes, formando partes estendidas de comportamento, assim, simples interações podem ser usadas para construir controle de fluxos (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

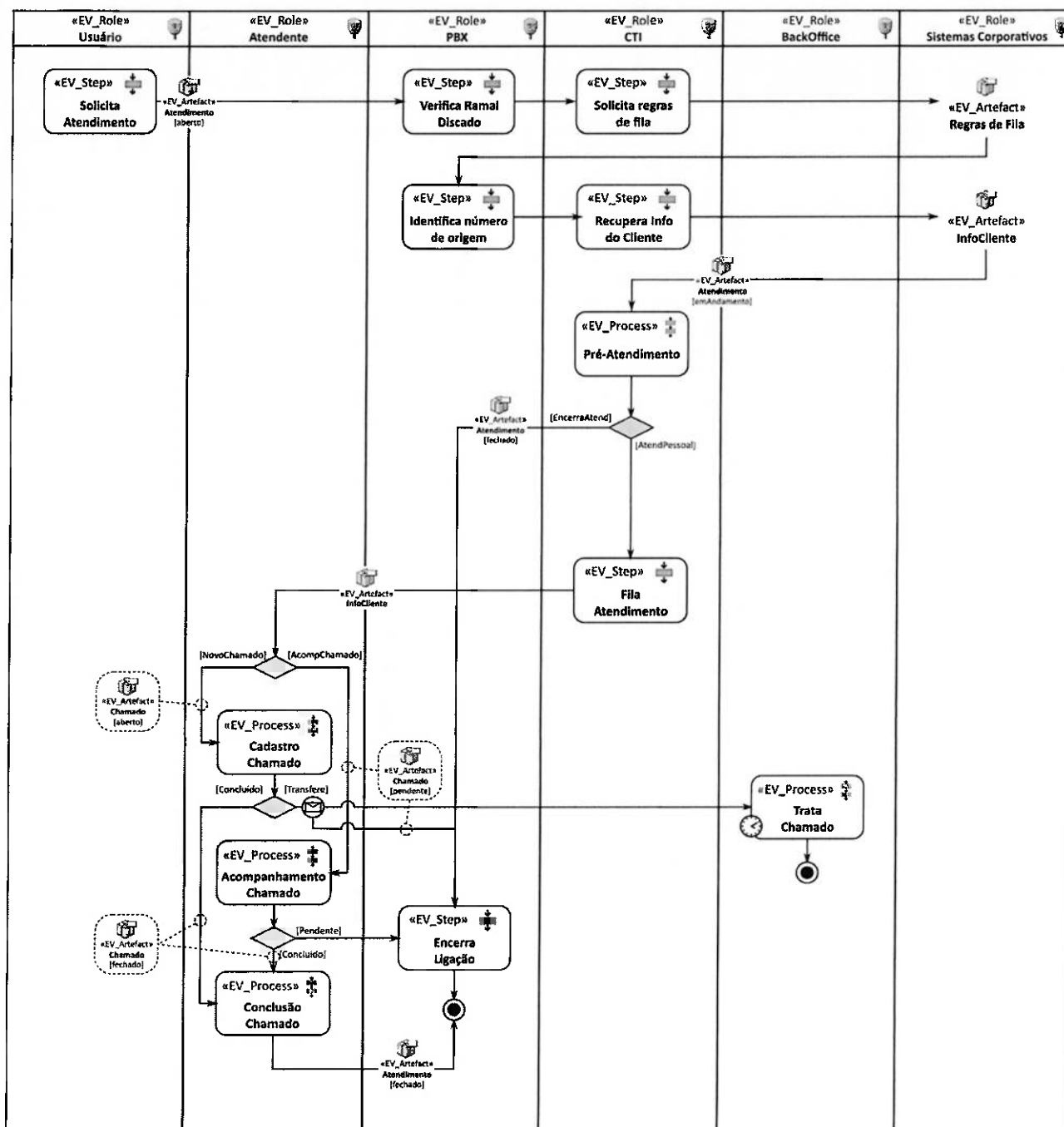
Todos os comportamentos da comunidade podem geralmente ser decompostas em um numero distinto de processos, como pode ser observado na Figura 3.11 os processos definidos por Gonçalves (2001) para a Central de Atendimento e atualizados para a arquitetura do *contact center* dentro do contexto corporativo que almejamos.

Cada um dos processos definidos dentro da comunidade *contact center* tem seu conjunto de passos que realiza as funções esperadas pelo sistema. Na Figura 3.12 é possível observar o detalhamento do processo Direcionador de Ramais, este processo, retirado do trabalho de Gonçalves (2001) e atualizado de acordo com o modelo de transformação arquitetural proposto, inicia-se quando o cliente liga para o número da Central de Atendimento desejado, o PBX verifica o ramal discado, consultando o sistema CTI pelas regras de encaminhamento que se aplicam a esse ramal, o CTI por sua vez consulta o banco de dados corporativo para recuperar as regras que foram definidas pelo CRM da empresa. Na sequência o PBX identifica o número de origem, e mais uma vez, consulta o CTI que se encarrega de recuperar as informações básicas do cliente do banco de dados.

O CTI em posse das regras de fila e dos dados do cliente inicia o processo Pré-Atendimento (não detalhado na figura), onde o cliente interage com a URA, cujas regras e árvore de opções são definidas pela empresa e armazenadas no banco de dados corporativo da empresa. O Cliente estando satisfeito com as consultas realizadas na URA pode encerrar a ligação, e dessa forma o atendimento se encerra no processo de Pré-Atendimento, com o armazenamento da interação do cliente no histórico de atendimento. Mas, caso o cliente precise de um atendimento pessoal, a ligação será transferida para o DAC, ou seja, a fila de espera, mas antes o CTI avalia a importância do cliente para a empresa, realizando novas consultas aos sistemas corporativos e em caso positivo ele reavalia a posição de todos os clientes atualmente em espera, posicionando este cliente na fila de acordo com o seu nível de importância para a empresa.

O cliente, então é transferido para o ramal do atendente disponível e junto os dados do cliente são transferidos para a tela do atendente em uma operação conhecida como sincronismo de tela, desse ponto o atendente pode dar continuidade ao atendimento como um novo chamado ou acompanhar um chamado anterior, dependendo, para isso, do encerramento da interação anterior, ou do fornecimento de um número de protocolo. Caso seja um novo chamado ou um chamado pendente e o operador conseguir resolver o chamado, este chamado será encerrado, caso contrário, uma nova interação será gerada, que acionará um grupo de especialistas com conhecimento especializado para resolver a solicitação do cliente.

Figura 3.12 – Detalhamento do processo de negócio direcionador de ramais



Fonte: Autor

A linguagem usada neste ponto de vista também define uma série de ações que ajudam a distinguir como as responsabilidades no sistema evoluem. O comprometimento é uma ação que resulta nos objetos correspondentes estarem sujeitos a certas obrigações. Essas obrigações podem ser passadas adiante do seu atual portador através da delegação para um agente. Um objeto também pode tornar fatos públicos através de uma declaração, que pode ser resultado de uma análise

sobre informações conhecidas de forma a derivar informações mais genéricas através de uma avaliação, dessa forma um objeto pode executar uma ação que gera uma receita que restringe comportamentos futuros dentro da comunidade. Dessa forma é possível identificar o participante responsável pelas alterações nas regras (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Outro ponto a ser abordado neste ponto de vista é a necessidade de colocar restrições em vários aspectos de qualidade de execução dos processos, isso requer que o comportamento básico seja anotado com um conjunto de sequencias de eventos permitidos, com as declarações de quais variações são permitidas no desempenho.

Dessa maneira, temos no *contact center* uma série de restrições que devem ser aplicadas no processo Direcionador de Ramais, herdadas do modelo de Gonçalves (2001), como por exemplo tempos de espera máximo na fila de atendimento e tempo máximo para solução de chamados em aberto, de acordo com a importância do cliente e do assunto sendo tratado, que são definidos por regras estabelecidas pelo CRM da empresa.

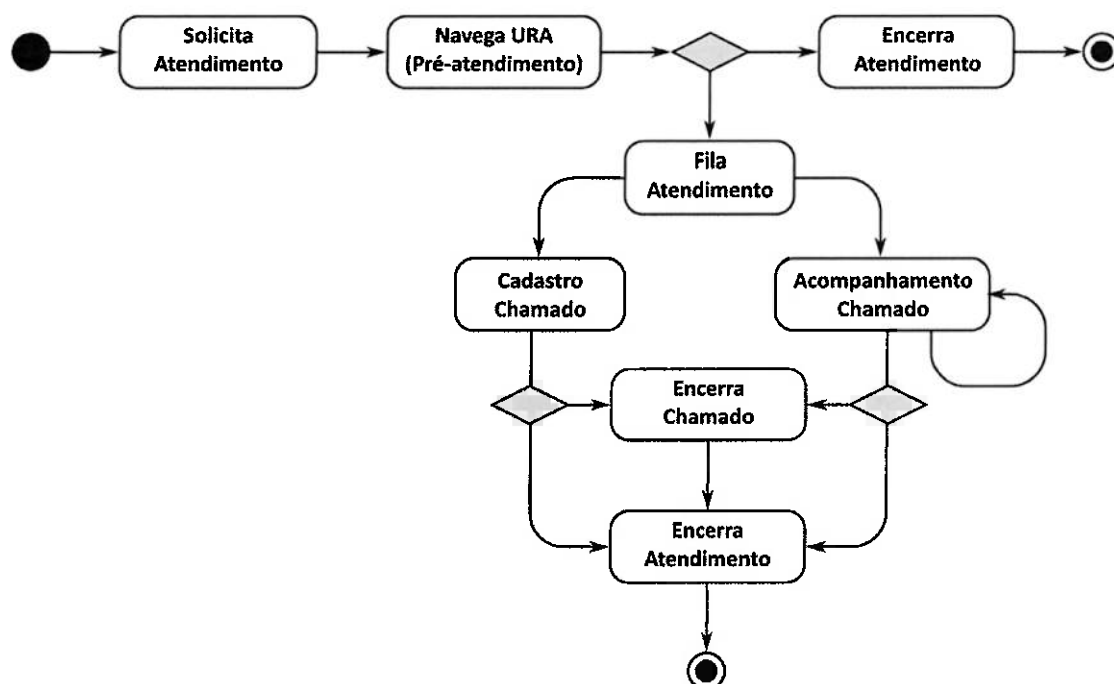
Em certas situações o cumprimento dessas políticas e níveis de serviços é dependente do cumprimento das obrigações pela comunidade e pelo ambiente em que este está inserido, por exemplo: No caso, da abertura de um chamado para o reenvio de uma edição não recebida de forma recorrente, isto é, quando o reenvio da edição superar a dois envios, as regras de negócio definem que novos reenvios devam ser realizados com 'aviso de recebimento', e dessa forma a resolução do chamado, no tempo acordado entre as partes, depende diretamente da presença do cliente no endereço para entrega da assinatura.

O último ponto abordado pelo ponto-de-vista empresa nesse trabalho é a identificação das interfaces de usuário do sistema. Como podemos observar na Figura 3.12, o processo Direcionador de Ramais possui interações entre os papéis Usuário, Atendente, PBX, CTI, Backoffice, Sistemas Corporativos (CRM e BD Corporativo). As interações podem ocorrer entre humano-humano, humano-maquina ou maquina-maquina. De acordo com Linington, Milosevic, Tanaka, & Vallecillo (2012) essas interações podem ser expressas de várias maneiras, mas a abordagem mais comum é a divisão das atividades de design em três aspectos separados, mas correlacionados:

- *Conteúdo* foca na informação persistente que são manipuladas pelo sistema, como dados brutos de entidades do sistema. Em sistemas de hipermídia também são incluídos entidades mais complexas, como imagens, vídeos, animações e trilhas de áudio;
- *Navegação*: cuida de como o conteúdo pode ser acessado pelos usuários do sistema e como o usuário pode mover entre os conteúdos, estabelecendo os caminhos para que isso ocorra;
- *Apresentação*: Lida com a visualização do conteúdo e os vários elementos interativos que suportam a funcionalidade do sistema (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Neste ponto-de-vista a preocupação principal é uma visão abstrata da navegação, outros aspectos da navegação são tratadas em outros pontos de vista; os elementos conteúdo são abordados no ponto-de-vista informação, no ponto-de-vista computacional é definido os fluxos de navegação e a apresentação é descrito no ponto-de-vista engenharia, com a definição de estilos e templates. E por último o ponto-de-vista tecnologia impõe algumas restrições na apresentação, incluindo padrões de usabilidade e acessibilidade (LININGTON, MILOSEVIC, TANAKA, & VALLECILLO, 2012).

Figura 3.13 - Máquina de estados para a navegação do processo direcionador de ramais



Fonte: Autor

A especificação da navegação neste ponto-de-vista está preocupada com a disponibilidade de funções para obter e manipular dados, e as ações que o sistema vai tomar como resultado.

Dessa forma, a especificação da navegação pode ser obtida a partir dos modelos de especificação empresa através da observação das interações entre os pares de papéis identificados anteriormente (ver Figura 3.12) e a partir da extração dessa observação dos estados de máquina que descrevem essas interações. Cada estado de máquina presente no diagrama da Figura 3.13 corresponde a um grupo de páginas ou formulários, que é apresentado para o usuário na qual este interage para o cumprimento da tarefa em mãos.

3.4.2 Ponto de vista informação

O foco do ponto de vista informação conforme visto no capítulo de fundamentação conceitual (item 2.3.1.2), é a de representar a informação compartilhada que é manipulada pelo sistema, desconsiderando como os dados são representados, implementados e distribuídos. Também é independente das interfaces computacionais e funções que manipulam esses dados ou da tecnologia utilizada para armazenar esses dados.

As informações que trafegam pelo sistema são representadas por esquemas que especificam o estado e a estrutura do objeto: o esquema invariante restringem os possíveis estados e a estrutura do objeto informação, ao longo do seu ciclo de vida. Os esquemas dinâmicos são representados pelos métodos implementados nas classes, e constitui toda ação que cause uma mudança no estado do objeto, e por último o esquema estático captura o estado e estrutura dos objetos Informação em um instante particular do tempo (Linnington, Milosevic, Tanaka, & Vallecillo, 2012).

Em primeiro lugar é necessário especificar os tipos de dados dos objetos informação que são utilizados para modelar os dados do sistema da Central de Atendimento, para isso expressamos esses objetos através de classes UML estereotipados como <<IV_NomeDoObjeto>> e identificados com o ícone ⓘ, conforme pode ser visto no diagrama de esquema invariante na Figura 3.15.

Para a modelagem do *contact center* proposto, iniciamos a especificação dos objetos informação tomando como base o diagrama de classes e os sistemas do nível de coordenação do SAA da arquitetura proposta por Gonçalves (2001) e os

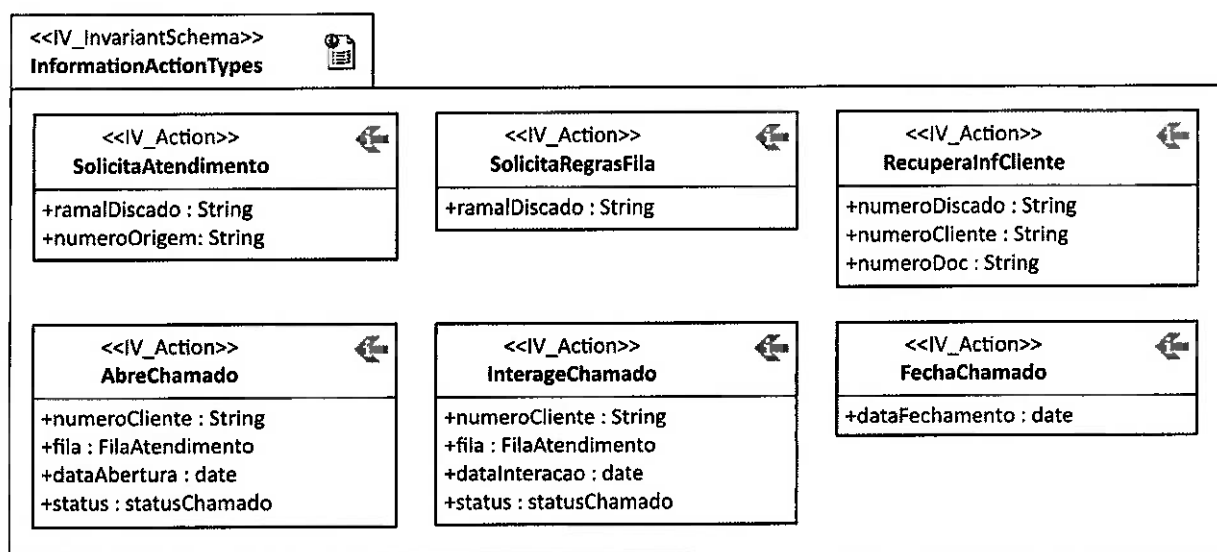
elementos arquiteturais do ponto de vista informação definidos anteriormente (item 3.4.1) e que já trazem incorporados as transformações arquiteturais do ponto de vista empresa para o *contact center*.

No caso da modelagem do *contact center* proposto no item 0 e para efeito de discussão serão definidos somente os objetos informação centrais do sistema, que são os objetos informação abaixo identificados:

- Objeto Informação Cliente <<IV_Cliente>>
- Objeto Informação Atendente <<IV_Atendente>>
- Objeto Informação Supervisor <<IV_Supervisor>>
- Objeto Informação Atendimento <<IV_Atendimento>>
- Objeto Informação Produto <<IV_Produto>>
- Objeto Informação CTI <<IV_CTI>>
- Objeto Informação CRM <<IV_CRM >>

Com a identificação dos objetos informação podemos definir os três esquemas em que é possível estruturar a especificação da informação de acordo com o seu comportamento (esquema dinâmico), as restrições que afetam os objetos e seus comportamentos (esquema invariante) e o estado da informação em determinado ponto no tempo (esquema estático).

Figura 3.14 - Alguns objetos ações-tipos do ponto de vista informação



Fonte: Autor


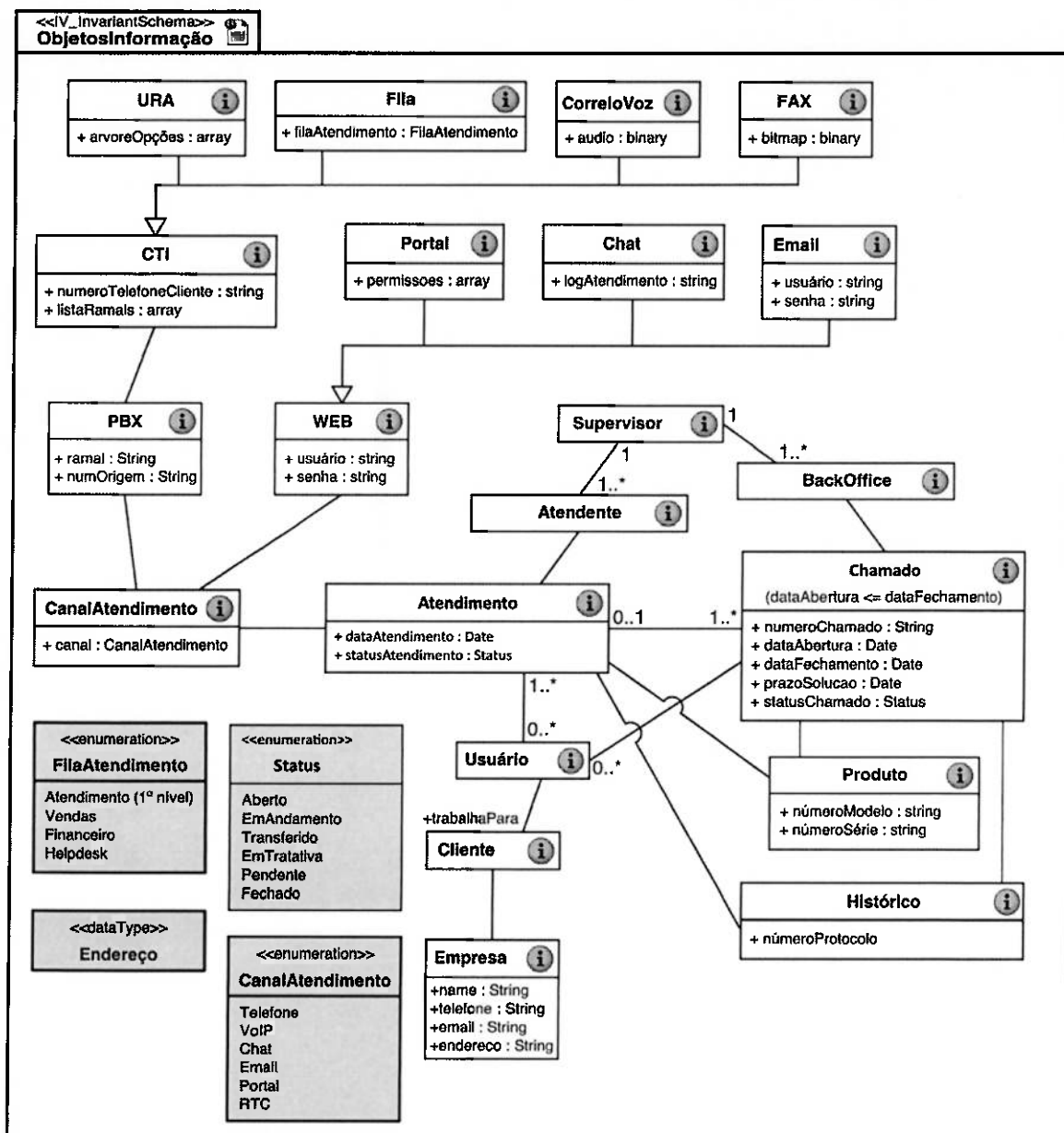
Mas antes de desenvolver os esquemas informação do sistema é necessário definirmos as ações-tipo, que expressam os eventos que causam a mudança de estado nos objetos informação dentro do pacote esquema invariante. As ações-tipo são representadas com a notação de sinal (i.e. signals) do UML, estereotipadas como <<IV_Action>> e identificados com o ícone , conforme pode ser observado na Figura 3.14.

Figura 3.15 – Esquema invariante do contact center (Diagrama de Classes)



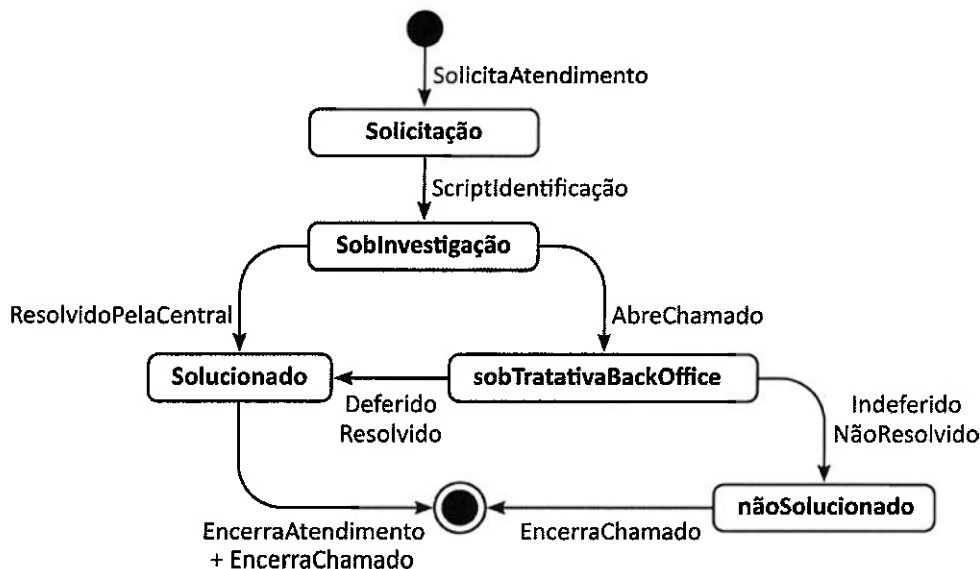
Fonte: Autor

É importante notar que os atributos desses sinais representam as informações transmitidas pelas interações entre esses elementos, e dessa forma, as instancias das classes que representam essas interações, serão expressas pelo envio ou recebimento desses sinais pela maquina de estados do objeto informação correspondente, como veremos no esquema dinâmico da Figura 3.16.

O diagrama de maquina de estado do UML pode ser utilizado para descrever o comportamento dos objetos informação, esse comportamento é disparado por sinais expressos pelas ações-tipo, definidos no passo anterior. A ocorrência de um sinal dentro do sistema altera o estado de todas as maquinas de estado que incluem uma transição desse sinal, demonstrando que uma interação ODP é um comportamento compartilhado, podendo causar a alteração de todos os objetos envolvidos.

Com os objetos informação e suas ações-tipo correspondentes definidos, desenhamos o esquema invariante de informação do sistema do *contact center*. Como pode ser observado na Figura 3.15, o esquema invariante descreve os objetos informação definidos para o sistema e o relacionamento entre eles, mas além dessas especificações, o esquema invariante é utilizado para especificar outros tipos de restrição nas informações manuseadas pelo sistema.

Figura 3.16 – Esquema dinâmico do objeto informação atendimento



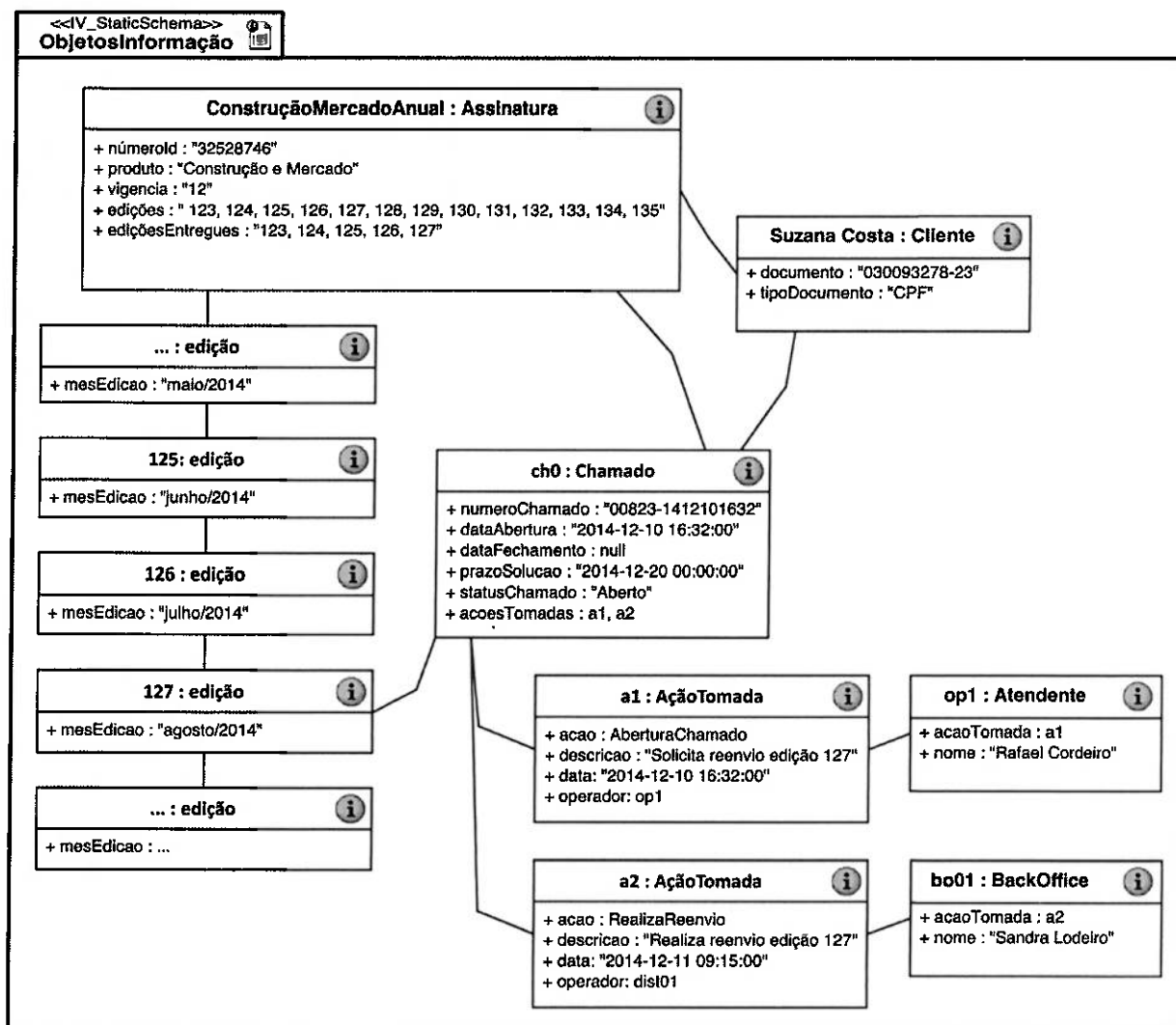
Fonte: Autor

Restrições entre objetos informação são demonstrados através da multiplicidade das associações entre eles, ou através de restrições OCL (“Object Constraint Language”) aos elementos do diagrama. Outras restrições OCL podem ser utilizadas

para pormenorizar as restrições do modelo, ou para expressar condicionais de integridade, definindo dessa forma, um conjunto de regras e restrições bem elaboradas para o modelo de dados compartilhado do sistema.

O esquema dinâmico Figura 3.16, por outro lado mostra as transformações ocorridas e as alternativas de fluxo da informação durante o seu ciclo de vida dentro do sistema através de um diagrama de estados do UML. Como pode ser observado existe uma correlação direta entre este diagrama e o diagrama de processo de negócios (Figura 3.12) elaborado no ponto de vista empresa. As ações neste diagrama são descritas pelas flechas que interligam as caixas, que por sua vez, descrevem as alterações promovidas nos objetos informações existentes no sistema.

Figura 3.17 – Exemplo de esquema estático capturado durante a resolução de um Chamado



Por ultimo, o esquema estático é utilizado para descrever a operação do sistema em determinado momento no tempo, capturando o estado das informações naquele momento. É muito utilizado para demonstrar o estado das informações na inicialização do processo dentro do sistema, ou até mesmo para demonstrar o estado das informações durante a operação do sistema para as partes interessadas (stakeholders), como pode ser observado na Figura 3.17, onde é demonstrado o estado do sistema durante a resolução de um chamado do *contact center*, esse diagrama se utiliza das classes definidas no esquema invariante e estão preenchidas com as informações fornecidas até o momento da captura do estado do sistema durante o atendimento em algum ponto do processo, conforme o diagrama da Figura 3.11.

3.4.3 Ponto de vista computação

O ponto de vista computação tem como principal objetivo a modelagem base das funcionalidades dos aplicativos e serviços oferecidos pelo sistema, e como esses serviços são organizados internamente com componentes e conectores, focando especificamente na funcionalidade do sistema e na arquitetura do software que é realizado por elas. Questões de como os componentes do software são distribuídas nos nódulos da plataforma e da tecnologia utilizada para implementar esses componentes são resolvidos nos pontos de vista engenharia e tecnologia. Dessa maneira a arquitetura de software desenvolvida aqui, torna-se naturalmente independente de distribuição e de plataforma, podendo ser reutilizada em qualquer outra plataforma e ser implementada de forma modular, fazendo com que o seu ciclo de vida não seja limitado pelas tecnologias utilizadas para implementá-la.

Nesse ponto de vista os arquitetos do sistema utilizam elementos que descrevem os objetos computacionais do sistema, as interações entre esses objetos através de suas interfaces, e os contratos que regem essas interações.

Esses objetos modelam os elementos funcionais básicos do sistema, descrevendo os serviços que oferece, bem como os serviços que consomem de outros objetos. Os objetos computacionais encapsulam parte do estado do sistema e sua funcionalidade permitindo que o desenho do sistema seja realizado de forma modular.

Os serviços oferecidos e requeridos por cada objeto computacional são definidos pelo conjunto de tipos de interfaces que o objeto implementa. Esses serviços, por

sua vez, são caracterizados pelos comportamentos envolvidos no seu uso e pela sintaxe de seus elementos, este último, expresso pela assinatura (i.e. signatures) de suas operações, fluxos e sinais que o suporta.

A especificação precisa das interfaces de cada objeto computacional permite aos usuários da corporação e demais sistemas de TI a explorar tanto os serviços que são oferecidos dentro dos sistemas do *contact center* e da corporação, quanto a maneira como devem ser invocados, bem como, auxilia na localização e descobrimento desse serviço em repositórios ou via web.

A definição do ponto de vista computacional usualmente é iniciado pela especificação em alto-nível da arquitetura de software necessária para alcançar os objetivos do sistema, através da definição dos componentes computacionais chaves do sistema e como eles interagem entre si.

A arquitetura de software define a estrutura e a organização do sistema em termos dos seus componentes e conectores de seus elementos de software, das propriedades visíveis externamente desses elementos e a relação entre eles, permitindo também extrair os princípios e diretrizes que regem como o sistema deve evoluir com o passar do tempo e provendo uma base para satisfazer tanto os requisitos funcionais quanto não-funcionais exigidos pelos stakeholders e pelo sistema..

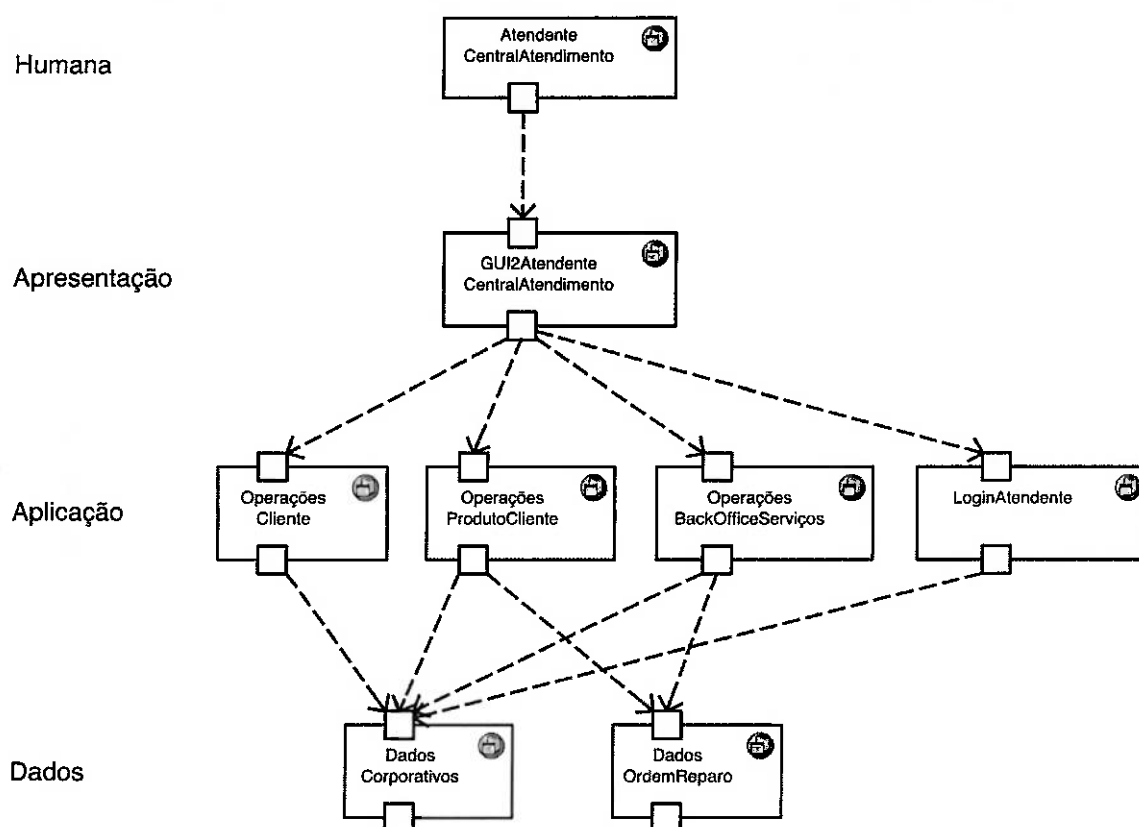
O primeiro passo para o desenho do ponto de vista computacional é de escolher qual estilo arquitetural (i.e. pipes-and-filter, client-server, multi-layer, blackboard, toaster, etc.) melhor se aplica ao sistema e aos requisitos dos usuários do sistema.

Para o modelo de Arquitetura Aberta de contact center, utilizaremos uma estrutura arquitetural simples de três camadas, onde a interface para os usuários, a lógica corporativa e o armazenamento de dados estão separados em camadas independentes, onde cada objeto de cada camada oferece serviços para as camadas imediatamente acima ou abaixo da camada sendo observada, organizando naturalmente as responsabilidades e a separação das preocupações do sistema.

Mas, entretanto, na especificação do ponto de vista empresa modelamos as interações de humanos atuando como atores com diversos papéis no sistema, como por exemplo: clientes, atendentes, supervisores e gerentes, e a modelagem deste ponto de vista precisa levar essa especificação em conta; e para isso é necessário acrescentarmos uma quarta camada para representar a interface humano-computador, como pode ser observado na Figura 3.18

No diagrama da Figura 3.19, mostramos uma parte maior da arquitetura de software do sistema proposto para o Contact center, agora organizados em pacotes apenas para ilustrar a divisão em camadas do sistema. Como pode ser observado nesta figura, são incluídos no diagrama serviços auxiliares, como por exemplo funções de login, pois estes provem um conjunto de serviços comuns para os demais objetos do sistema.

Figura 3.18 – Divisão em camadas do ponto de vista computacional



Fonte: Autor

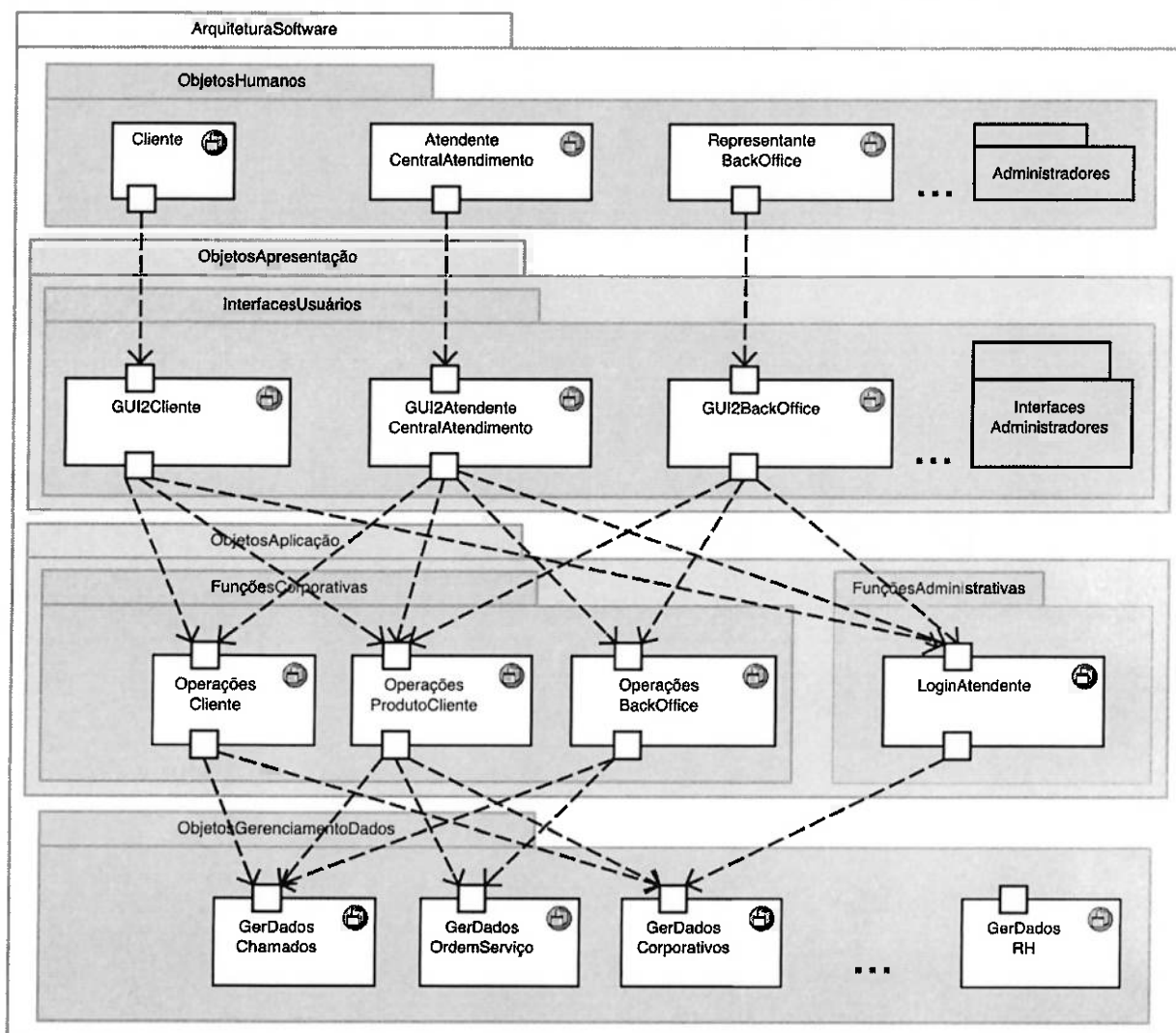
Continuando na Figura 3.19, as ligações entre cada objeto, representa, em alto-nível de abstração, as ligações necessárias para o funcionamento correto do sistema. As ligações entre esses objetos são realizadas através das interfaces projetadas em cada objeto e nas assinaturas de ações (i.e. action signatures) que definem os serviços providos ou consumidos de outros objetos do sistema, e especificados no próximo passo.

Na sequencia de projeto do ponto de vista computacional, detalhamos essas ligações estabelecidas na abstração em alto-nível, como pode ser observado na

Figura 3.20 onde estão definidas as interfaces, as assinaturas de operações dessas interfaces e como esses objetos se comportam durante o funcionamento do sistema.

As interfaces são representadas em UML4ODP através da notação de portas UML, e as assinaturas de cada objeto são especificadas utilizando a notação de interfaces do UML, as ligações entre objetos são representados com o uso da notação bola-e-soquete (i.e. ball-and-socket) do UML para representar ligações primitivas entre os objetos computacionais.

Figura 3.19 – Parte da Arquitetura de Software para o contact center



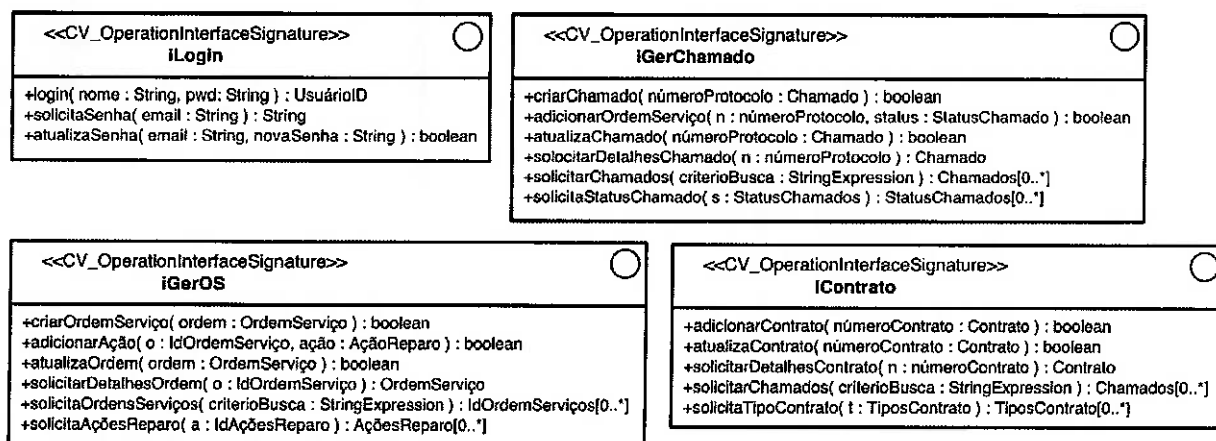
Fonte: Autor

Ainda na Figura 3.20, observamos a especificação detalhada de algumas assinaturas operacionais dos serviços fornecidos e requeridos de cada porta de

diversos objetos computacionais do sistema com a caracterização de algumas interações que ocorrem nessas interfaces. Como exemplo, é possível verificar que a interface para gerenciamento de chamados, nomeado como *iGerChamado*, apresenta diversas interfaces de operação por onde trafegam os dados dos objetos do ponto de vista informação, para essa interface, temos assinaturas específicas para cada tipo de ação desejada especificados de forma semelhante aos métodos das classes UML, descrevem os parâmetros necessários para sua execução e o tipo de dados que retorna.

Sendo assim, é possível verificar que a assinatura *+atualizaChamado()*, requer o dado *númeroProtocolo* e retorna o data tipo booleano para indicar o sucesso ou não da operação, e que a assinatura *+solicitarDetalhesChamados* da mesma interface, toma como parâmetro o *númeroProtocolo* e retorna a classe, do ponto de vista informação, *Chamado*.

Figura 3.20 – Especificação detalhada das assinaturas das interfaces



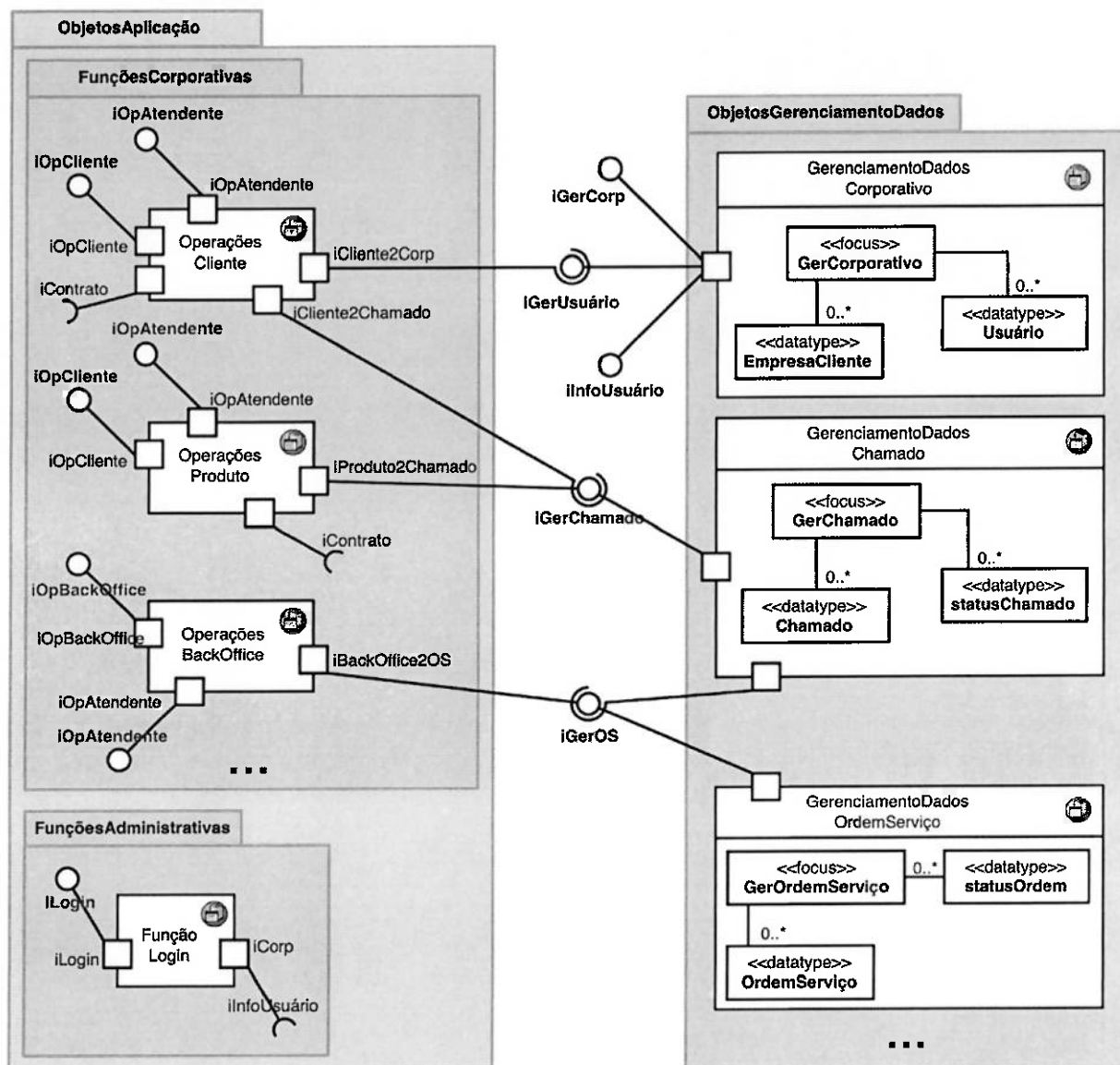
Fonte: Autor

Na Figura 3.21 podemos observar as ligações primitivas estabelecidas entre os objetos através do uso das interfaces apropriadas de acordo com a especificação de suas assinaturas.

Ainda na Figura 3.21 podemos observar um detalhamento maior de alguns componentes, como por exemplo os seus classificadores internos, que especificam o seu comportamento no sistema. Como exemplo, o componente GerenciamentoDadosChamado, gerencia as informações de todos os chamados no sistema, gerenciando e armazenando as informações relevantes sobre eles e dessa forma contendo um classificador GerChamado, estereotipado com a classe

<<focus>>, que detem a propriedade sobre as informações de chamados que são conhecido pelo sistema, ou seja, seus <<datatype>>.

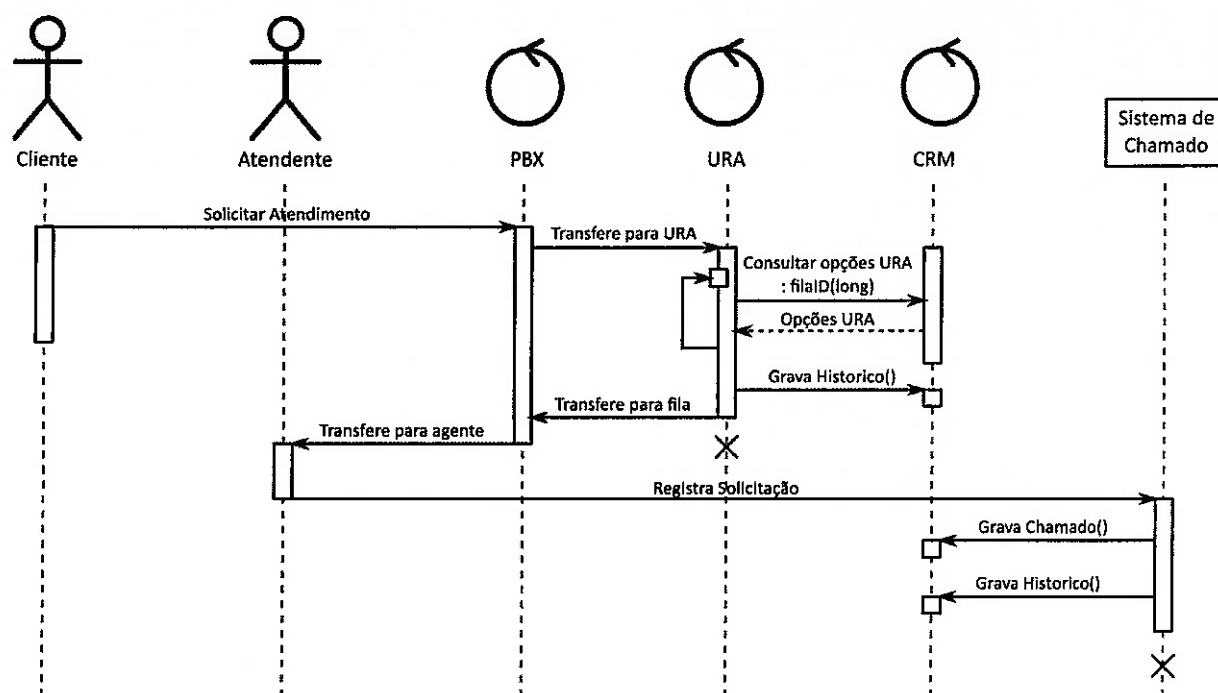
Figura 3.21 – Alguns componentes computacionais do sistema com suas interfaces expostas e os relacionamentos entre eles.



Fonte: Autor

Na Figura 3.22, verificamos o diagrama de sequencia para uma das atividades do *contact center* onde o cliente entra em contato com o *contact center* através do telefone para solicitar auxílio geral para a sua assinatura.

Figura 3.22 – Diagrama de sequencia de registro de solicitação geral.



Fonte: Autor

4 Conclusão

O principal objetivo de um *Contact center*, como vimos ao longo deste trabalho é auxiliar nas estratégias de CRM da empresa. Os Avanços tecnológicos, hoje mais do que nunca nas áreas de telefonia e computação permitem uma aproximação e disponibilidade ao cliente quase instantânea.

As diretrizes e arquitetura que Gonçalves (2001) traçou em seu inovador tratado sobre a Central de Atendimento, não só foram concretizadas, como foram extrapoladas neste trabalho com os novos conceitos de comunicação a qualquer tempo e lugar, confirmando a necessária alteração para *contact center*. A Alteração é conceitual e se serve de tecnologias de telefonia que permitem uma integração maior com os sistemas computacionais da empresa e aplicativos de mensageria instantânea que permitem que a empresa esteja disponível ao cliente a qualquer momento em que o mesmo necessite de auxílio ou informações sobre o seu relacionamento ou possível relacionamento com o cliente.

Nesta mesma direção, o surgimento e avanço das mídias sociais na internet como blogs, ferramentas de convivência social, i.e. *Facebook*, *Google+*, *Twitter*, entre outros e sites que permitem fazer a reclamação de forma pública, i.e. *ReclameAqui*, estão fazendo com que as empresas prestem atenção à construção do relacionamento com cada um dos seus clientes.

Dessa forma, o *contact center*, além de prestar o atendimento receptivo, seja pelos meios tradicionais de telefonia, seja pelos meios de comunicação de tempo real (mensageria instantânea), ou tempo não real de comunicação via Internet (e-mail, sms, WhatsApp) de forma eficiente e precisa, precisa capturar das mais diversas fontes de informação na Internet as mais diversas informações. Assim, de elogios às críticas e reclamações sobre a empresa, da divulgação de seus produtos e/ou serviços às respostas às pessoas que publicaram essas informações e onde as publicaram, é preciso atuar sobre essas interações de forma proativa, com o objetivo de estreitar a relação com atuais clientes e atingir novos clientes, almejando cada vez mais os objetivos do CRM.

Além disso, esses novos canais de relacionamento que a empresa estabelece com as redes sociais e sites permitem que a empresa armazene em sua base de dados corporativos ainda mais informações sobre os seus clientes, enriquecendo as bases de dados corporativas. E dessa forma, tornando as análises de dados

realizadas no *Data Warehouse*, *Data Mart* e *Business Intelligence* mais ricas. Propiciando um maior conhecimento dos seus clientes e do mercado para a corporação.

O uso do Architectural-Driven Modernization propicia técnicas de transformação arquitetural do sistema 'as-is' para o modelo 'to-be', fornecendo técnicas e ferramentas para descobrir e mapear sistemas já maduros em uma base de conhecimento, reduzindo o tempo de produção de um novo modelo arquitetural.

A transformação arquitetural permite que partes do sistema sejam recicladas e reutilizadas, evita que partes já existentes sejam reinventadas e permite aos projetistas a se concentrarem no contexto e nos novos requisitos do modelo arquitetural sendo desenvolvido.

Na mesma direção o RM-ODP permite a divisão das diversas responsabilidades do sistema e de seus componentes em áreas de atuação específicas facilitando a divisão do trabalho em equipes especializadas e permitindo que cada uma dessas partes do sistema avancem separadamente.

Dessa forma esse trabalho busca através do uso do ADM junto com o RM-ODP a atualização tecnológica de um modelo de central de atendimento ('as-is') para um modelo de *contact center* ('to-be') de maneira gradual e transparente a corporação e de seus usuários e espera ampliar as questões teóricas, metodológicas e técnicas, contribuindo para a continuidade dos avanços nesta área.

ANEXO A

REQUISITOS FUNCIONAIS E NÃO FUNCIONAIS

Bibliografia

AKSIN, Zeynep; ARMONY, Mor; Mehrotra, Vijay **The Modern Call Center: A multi-disciplinary perspective on Operations Management Research**. Nov/Dez 2007, Production and Operations Management Society (POMS) Vol. 16, No. 6, pp. 665-688

APTE, Uday M.; MASON Richard O. **Global disaggregation of information-intensive services**. Dallas, Texas, 1995, Management Science, Vol.41, No. 7, pp. 1250-1262.

Cisco Unified Communications System Release 9.0 SRND 2013, April 30 Cisco Systems, Inc. San Jose, CA

GONÇALVES, Ana Paula Cordeiro **Proposta de Arquitetura Aberta de Central de Atendimento**. São Paulo, 2001. 140p. Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo.

ITHAPE, Priya K., HIWALE, Dr. Prof. Anil S. **Multichannel Contact Center** International Journal of Research in Computer and Communication Technology, Vol 3, Issue 7, p. 739-742, July 2014

KHUSIDMAN, Dr. Vitaly; ULRICH, William **Architecture-Driven Modernization: Transforming the Enterprise** 12 Out. 2007

INTERNATIONAL CUSTOMER MANAGEMENT INSTITUTE - LEE, Howard **How to Modernize Your Contact Center in the Age of Siri**, 23 Maio 2014, disponível em: <http://www.lcmi.com/Resources/Technology/2014/05/How-to-Modernize-Your-Contact-Center-in-the-Age-of-Siri?p=1> . Acesso em 4 Jun 2014.

LININGTON, Peter F.; MILOSEVIC, Zoran; TANAKA, Akira; VALLECILLO, Antonio **Building Enterprise Systems with ODP: An introduction to Open Distributed Processing**, 2012 – CRC Press, Boca Raton, FL.

PEPPERS AND ROGERS GROUP **CRM Series Marketing 1 to 1: Um guia executivo para entender e implantar estratégias de Customer Relationship Management**. 1a. ed. Janeiro 2000

TAN, T. K.; RAGHUNATHAN A.; JHA, N. K. **Software Architectural Transformations: A New Approach to Low Energy Embedded Software**, 2003 - IEEE