

Madalena Pereira da Silva

**UM MODELO DE GERENCIAMENTO DA QUALIDADE DE  
EXPERIÊNCIA PARA A PROVISÃO DE SERVIÇOS CIENTES  
DE CONTEXTO**

Tese submetida ao Programa de Pós Graduação em Engenharia e Gestão do Conhecimento da Universidade Federal de Santa Catarina para a obtenção do Grau de Doutora em Engenharia e Gestão do Conhecimento.

Orientador: Prof. Dr. Mário Antônio Ribeiro Dantas.

Coorientador: Prof. Dr. Alexandre Leopoldo Gonçalves.

Florianópolis  
2017

Ficha de identificação da obra elaborada pelo autor,  
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Silva, Madalena Pereira da  
Um modelo de gerenciamento da qualidade de  
experiência para a provisão de serviços cientes de  
contexto / Madalena Pereira da Silva ; orientador,  
Mário Antônio Ribeiro Dantas, coorientador,  
Alexandre Leopoldo Gonçalves, 2017.  
305 p.

Tese (doutorado) - Universidade Federal de Santa  
Catarina, , Programa de Pós-Graduação em Engenharia e  
Gestão do Conhecimento, Florianópolis, 2017.

Inclui referências.

1. Engenharia e Gestão do Conhecimento. 2.  
Qualidade de Experiência. 3. Qualidade de Serviço.  
4. Experiência do Usuário. 5. Software-Defined  
Networking. I. Dantas, Mário Antônio Ribeiro . II.  
Gonçalves, Alexandre Leopoldo. III. Universidade  
Federal de Santa Catarina. Programa de Pós-Graduação  
em Engenharia e Gestão do Conhecimento. IV. Título.

Madalena Pereira da Silva

**UM MODELO DE GERENCIAMENTO DA QUALIDADE DE  
EXPERIÊNCIA PARA A PROVISÃO DE SERVIÇOS CIENTES  
DE CONTEXTO**

Esta Tese foi julgada adequada para obtenção do Título de Doutora e aprovada em sua forma final pelo Programa de Pós Graduação em Engenharia e Gestão do Conhecimento.

Florianópolis, 14 de Setembro de 2017.

\_\_\_\_\_  
Prof.<sup>a</sup> Gertrudes Aparecida Dandolini, Dra.  
Coordenador do Curso

**Banca Examinadora:**

\_\_\_\_\_  
Prof. Mário Antônio Ribeiro Dantas, Dr.  
Orientador  
Universidade Federal de Santa Catarina

\_\_\_\_\_  
Prof. José Leomar Todesco, Dr.  
Universidade Federal de Santa Catarina





---

Prof. Vinicius Medina Kern, Dr.  
Universidade Federal de Santa Catarina

---

Prof. Alex Sandro Roschildt Pinto, Dr.  
Universidade Federal de Santa Catarina

---

Prof. Alex Borges Vieira, Dr.  
Universidade Federal de Juiz de Fora

---

Prof.<sup>a</sup> Vera Maria Benjamim Werneck, Dr.<sup>a</sup>  
Universidade do Estado do Rio de Janeiro



Este trabalho é dedicado aos meus  
filhos Otávio e Lavínia, a razão da  
minha existência.



## AGRADECIMENTOS

Aos três amores da minha vida, meu esposo Júnior, meu filho Otávio e minha filha Lavínia. Obrigada pelo apoio, por esperarem pacientemente por mim. Amo vocês!

A minha sobrinha Vanusa por dedicar boa parte do seu tempo cuidando e ajudando na educação dos meus pequenos notáveis (Otávio e Lavínia). Sou eternamente grata pelo carinho que você tem por eles!

A minha mãe Ana Maria pela vontade de viver e permanecer na luta mesmo depois de tantas perdas e dificuldades. Ao meu saudoso pai Hermelino... queria tanto poder ver seus olhos brilharem por mais esta conquista e poder dar-lhe um abraço. Obrigada pela vida meus queridos!

Aos meus irmãos, irmãs, cunhados, cunhadas, sobrinhos e familiares, que acompanharam, choraram, riram, aconselharam, incentivaram e acreditaram na minha pessoa. Prometo que em breve poderei visitá-los e também recebê-los em minha casa.

A Sra Mariléia por ajudar cuidar das crianças nas férias e nos finais de semana para que eu pudesse estudar. Ao professor José Cé por preparar as deliciosas refeições.

Ao meu orientador, professor Mário Dantas, pelos ensinamentos, apoio, compreensão e paciência. Um verdadeiro mestre, um excelente orientador. Obrigada por acreditar que eu sou capaz!

Ao meu coorientador, professor Alexandre Leopoldo Gonçalves, pelas trocas de ideias, apoio e disponibilidade.

Aos professores do EGC pelos conhecimentos compartilhados e pela excelência em todas as atividades do PPGE/GC, em especial aos professores Mario Dantas e Alexandre Gonçalves, que sempre se mostraram dispostos em ouvir, sanar dúvidas e contribuir com suas sábias palavras. Estendo os meus agradecimentos ao professor Roberto Willrich, no INE-UFSC-FLS e ao Professor Alex Sandro Roschildt Pinto, do Departamento de Engenharia e Automação – DAS-UFSC-BNU.

Aos meus amigos que estiveram juntos nesta caminhada e que sempre foram solidários em diversas situações, extensivo a Sergio Murilo, Cleber Vinícius, Cristina Yamaguchi, Joaquim Rodrigo, Vinicius Barreto Klein e Marica Boell. Viajamos muitos quilômetros entre Lages e Florianópolis para estudar e, em função das necessidades financeiras sempre voltávamos “correndo” para trabalhar. Mas, valeu a pena!

Ao meu amigo Fabian pela disponibilidade em discutir e trocar

ideias para a produção deste documento.

Ao meu colega Felipe Volpato pelas longas e produtivas conversas via Skype, compartilhamento de materiais, apoio na programação e parceria na escrita dos artigos.

Aos irmãos de fé pelas orações e palavras de otimismo nos momentos de indecisões e aflições.

Aos membros de banca pelo aceite do convite e pelas brilhantes contribuições.

Ao Criador Maior – DEUS, pois sem ele eu nada seria.

É melhor tentar e falhar, que se preocupar e ver a vida passar. É melhor tentar, ainda que em vão, que se sentar fazendo nada até o final. Eu prefiro na chuva caminhar, que em dias tristes em casa me esconder. Eu prefiro ser feliz, embora louco, que em conformidade viver.  
(Martin Luther King).





## RESUMO

A QoE (*Quality of Experience*) refere-se a avaliação das percepções e expectativas dos usuários no uso do serviço. Essa avaliação pode ser influenciada por vários aspectos (e.g. funcionais, técnicos, humanos) que podem interferir na UX (*User Experience*). No entanto, em sistemas de comunicação de dados, a QoE tem sido estudada como uma extensão da QoS (*Quality of Service*), onde muitas vezes, apenas parâmetros técnicos relacionados com o desempenho da rede são usados para prever o nível de satisfação do usuário em relação ao serviço provido. Para uma gestão bem sucedida da QoE é necessário uma compreensão profunda e abrangente das múltiplas dimensões da percepção humana sobre a qualidade e dos fatores de influência contemplados nessas dimensões. Nesta tese é proposto um modelo de gerenciamento da QoE para prover serviços cientes de contexto, em sistemas de comunicação de dados, trazendo como resultados: (i) uma taxonomia multidimensional da QoE, pautada numa abordagem interdisciplinar; (ii) um modelo ontológico de representação do conhecimento da QoE; (iii) uma aplicação de gerenciamento de rede, com características semânticas e autonômicas, que aprende a UX e provê serviços cientes de contexto; (iv) uma arquitetura de provisão de serviços orientada ao usuário, projetada para a Internet do Futuro, com o uso de SDN (*Software-Defined Networking*). Para verificar a viabilidade do modelo proposto foi apresentado um cenário de provisão de serviços de *eHealth* entre um AAL (*Ambient Assisted Living*) e uma unidade de atendimento remota. Na avaliação foi verificado se os componentes do modelo proposto, quando comparados com os componentes da abordagem nativa, são capazes de: (i) prover serviços cientes do contexto e (ii) detectar, planejar e desencadear ações para restaurar a QoE, usando o conhecimento disponível na KB (*Knowledge Base*). A análise estatística realizada sobre os dados dos resultados experimentais permitiu evidenciar, com intervalo de confiança de 95%, que todos os serviços de *eHealth*, usando os componentes do modelo proposto, foram providos com qualidade superior quando comparados com os componentes da abordagem nativa.

**Palavras-chave:** Qualidade de Experiência, Qualidade de Serviço, Experiência do Usuário, Conhecimento, Ontologia, Rede Definida por Software, Ambiente de Vida Assistido.



## ABSTRACT

QoE (Quality of Experience) refers to the evaluation of users' perceptions and expectations about the use of a service. This evaluation can be influenced by several aspects (e.g. functional, technical, human) that may interfere with UX (User Experience). However, in data communication systems, QoE has been studied as an extension of QoS (Quality of Service), where often only technical network-performance parameters are used to predict the level of user satisfaction about the service provided. Successful QoE management requires a deep and comprehensive understanding of the multiple dimensions of human perception about quality in addition to the influencing factors inherent to such dimensions. In this doctoral thesis, a QoE management model is proposed to provide context-aware services in data communication systems, resulting in: (i) a multidimensional QoE taxonomy based on an interdisciplinary approach; (ii) an ontological model of QoE knowledge representation; (iii) a network management application, with semantic and autonomic features, that learns UX and provides context-aware services; (iv) a user-oriented service delivery architecture designed for Future Internet using Software-Defined Networking (SDN). In order to verify the feasibility of the proposed model, a scenario for eHealth service delivery was created between an AAL (Ambient Assisted Living) environment and a remote service unit. The evaluation verified whether the components of the proposed model, when compared to the components of the native approach, were able to: (i) provide context-aware services and (ii) detect, plan and trigger actions to restore QoE, using the knowledge available in the KB (Knowledge Base). The statistical analysis carried out with the data of the experimental results showed that, with a 95% confidence interval, all eHealth services using the components of the proposed model were provided with superior quality when compared to the components of the native approach.

**Keywords:** Quality of Experience, Quality of Service, User Experience, Knowledge, Ontology, Software-Defined Networking, Ambient Assisted Living.



## LISTA DE FIGURAS

Figura 1 - Diferentes níveis de abstração da QoS e QoE .....	34
Figura 2 - Aderência da Tese ao PPGEGC .....	41
Figura 3 - Modelo do Processo DSRM .....	45
Figura 4 - Espiral de criação de conhecimentos organizacionais .....	54
Figura 5 - Espectro semântico das abordagens usadas para modelar e representar o conhecimento .....	60
Figura 6 - Mapa conceitual com conceitos e ferramental para o desenvolvimento de aplicações semânticas .....	63
Figura 7 - Versões da linguagem OWL .....	68
Figura 8 - Simplificação visual de uma ontologia criada no Protégé ....	70
Figura 9 - Processo de desenvolvimento e ciclo de vida da Methontology .....	77
Figura 10 - Processo de desenvolvimento da metodologia On-to- Knowledge .....	78
Figura 11 - Processo de desenvolvimento do Ontology Development 101 .....	79
Figura 12 - Captura de imagem distorcida .....	98
Figura 13 - Diversidade de fatores de impacto na QoE .....	100
Figura 14 - Estrutura da KB para armazenar a experiência do serviço	115
Figura 15 - Propriedade de dados mapeados dos relatórios dos terminais para conceitos na KB .....	116
Figura 16 - Modelo geral da experiência do cliente .....	117
Figura 17 - Visão das classes da ontologia do domínio do cliente .....	118
Figura 18 - Visão das classes da ontologia do domínio do serviço .....	119
Figura 19 - Visão das classes da ontologia de negócios .....	119
Figura 20 - Arquitetura para a maximização da QoE em serviços multimídia .....	120
Figura 21 - Submodelo da ontologia da topologia da rede .....	121
Figura 22 - Submodelo da ontologia dos serviços disponíveis em uma rede de acesso .....	122
Figura 23 - Submodelo das capacidades de otimização da QoE e informações de configurações presentes e passadas de cada nodo .....	123
Figura 24 - Submodelo da ontologia do monitoramento dos serviços na rede de acesso .....	123
Figura 25 - Comparativo do modelo de hardware de rede com o modelo Programável .....	132
Figura 26 - Arquitetura SDN .....	133
Figura 27 - xemplo de rede com <i>switches</i> e roteadores <i>OpenFlow</i> .....	136
Figura 28 - Entrada na tabela de fluxos de um <i>switch OpenFlow</i> .....	137

Figura 29 - Procedimento de instalação de um novo fluxo.....	138
Figura 30 - Troca de mensagens do estado do fluxo entre o SDNC e <i>switch</i> .....	139
Figura 31 - Processo de monitoração da latência.....	144
Figura 32 - Abstração do modelo de gerenciamento da QoE .....	150
Figura 33 - Taxonomia das dimensões de influência na QoE.....	151
Figura 34 - Processo de desenvolvimento do modelo ontológico.....	158
Figura 35 - Classes base da ontologia de avaliação da QoE .....	163
Figura 36 - Ligação do conceito Time da ontologia proposta com a ontologia DAML Time .....	169
Figura 37 - Ligação da classe User com a classe Person da Ontologia FOAF .....	170
Figura 38 - Principais classes do vocabulário WGS84 .....	170
Figura 39 - Ligação da classe Measure com a classe UnitOfMeasurement da Ontologia MUO.....	171
Figura 40 - Arquitetura conceitual para a gerenciamento da QoE .....	182
Figura 41 - Componentes da camada de gerenciamento de serviços..	184
Figura 42 - Estrutura simplificada da KB usada na definição da política de QoS.....	185
Figura 43 - Processo de avaliação da QoE.....	186
Figura 44 - Simplificação da estrutura da KB usada para a avaliação da UX.....	188
Figura 45 - Arquitetura da aplicação de gerenciamento de QoE .....	190
Figura 46 - Aplicações de gerenciamento de recursos.....	193
Figura 47 - Algoritmo de gerenciamento do fluxo.....	196
Figura 48 - Componentes da camada de controle .....	199
Figura 49 - Infraestrutura utilizada nos experimentos .....	210
Figura 50 - IC do throughput: sinais ambientais (abordagem proposta) .....	223
Figura 51 - IC do throughput: sinais ambientais (abordagem nativa) .	223
Figura 52 - IC da latência: sinais ambientais (abordagem proposta) ..	224
Figura 53 - IC da latência: sinais ambientais (abordagem nativa) .....	224
Figura 54 - IC da perda de pacotes: sinais ambientais (abordagem proposta) .....	226
Figura 55 - IC da perda de pacotes: sinais ambientais (abordagem nativa) .....	226
Figura 56 - IC do throughput: sinais biomédicos (abordagem proposta) .....	227
Figura 57 - IC do throughput: sinais biomédicos (abordagem nativa)	228
Figura 58 - IC da latência: sinais biomédicos (abordagem proposta) .	229
Figura 59 - IC da latência: sinais biomédicos (abordagem nativa) .....	229

Figura 60 - IC da perda de pacotes: sinais biomédicos (abordagem proposta).....	230
Figura 61 - IC da perda de pacotes: sinais biomédicos (abordagem nativa).....	231
Figura 62 - IC do throughput: imagens médicas (abordagem proposta) .....	232
Figura 63 - IC do throughput: imagens médicas (abordagem nativa) .	232
Figura 64 - IC da latência: imagens médicas (abordagem proposta)...	233
Figura 65 - IC da latência: imagens médicas (abordagem nativa) .....	234
Figura 66 - IC da perda de pacotes: imagens médicas (abordagem proposta).....	235
Figura 67 - IC da perda de pacotes: imagens médicas (abordagem nativa).....	235
Figura 68 - IC do throughput: teleconsulta (abordagem proposta) .....	236
Figura 69 - IC do throughput: teleconsulta (abordagem nativa) .....	237
Figura 70 - IC da latência: teleconsulta (abordagem proposta).....	238
Figura 71 - IC da latência: teleconsulta (abordagem nativa).....	238
Figura 72 - IC da perda de pacotes: teleconsulta (abordagem proposta) .....	239
Figura 73 - IC da perda de pacotes: teleconsulta (abordagem nativa).	240
Figura 74 - IC na restauração da QoE, sem limpeza nos dados .....	255
Figura 75 - IC na restauração da QoE, com limpeza nos dados.....	256





## LISTA DE QUADROS

Quadro 1 - Exemplo de regra SWRL .....	71
Quadro 2 - Estrutura de uma consulta SPARQL.....	72
Quadro 3 - Conceitos de QoE em definições existentes.....	100
Quadro 4 - Regras de um <i>switch OpenFlow</i> .....	137
Quadro 5 - Estatísticas por fluxo, por porta e por fila .....	139
Quadro 6 - Conceitos da ontologia de avaliação da QoE.....	163
Quadro 7 - Consulta SPARQL da QC 1.....	173
Quadro 8 - Consulta SPARQL da QC 2.....	173
Quadro 9 - Consulta SPARQL da QC 3.....	173
Quadro 10 - Consulta SPARQL da QC 4.....	174
Quadro 11 - Consultas SPARQL das QC 5, 6, 7 e 8.....	174
Quadro 12 - Consulta SPARQL da QC 9.....	175
Quadro 13 - Consulta SPARQL da QC 10.....	175
Quadro 14 - Regra SWRL da QC 1.....	176
Quadro 15 - Regra SWRL da QC 2.....	177
Quadro 16 - Regra SWRL da QC 3.....	177
Quadro 17 - Regra SWRL da QC 4.....	177
Quadro 18 - Regras SWRL das QC 5, 6, 7 e 8.....	178
Quadro 19 - Regra SWRL da QC 9.....	178
Quadro 20 - Regra SWRL da QC 10.....	178
Quadro 21 - Regra SWRL da QC 11.....	179
Quadro 22 - Regra SWRL da QC 12.....	179
Quadro 23 - Regra SWRL da QC 13.....	179
Quadro 24 - Regra SWRL da QC 14.....	180
Quadro 25 - Regra SWRL da QC 15.....	180
Quadro 26 - Regra SWRL da QC16.....	181
Quadro 27 - Regra SWRL da QC 17.....	181
Quadro 28 - Regras SWRL das QC 18, 19, 20 e 21 .....	181



## LISTA DE TABELAS

Tabela 1 - Busca sistemática da literatura .....	39
Tabela 2 - Teses e dissertações do PPGEGC relacionadas com a proposta.....	42
Tabela 3 - Classificação de instrumentos de representação do conhecimento quanto ao uso .....	60
Tabela 4 - Comparativo das características entre algumas linguagens formais usadas para representação do conhecimento .....	61
Tabela 5 - Dimensões da qualidade em diferentes serviços .....	85
Tabela 6 - Dimensões da qualidade em serviços de comércio eletrônico .....	85
Tabela 7 - Requisitos de QoS em diferentes aplicações.....	91
Tabela 8 - Requisitos de QoS em protocolos de transmissão em sistemas de teletrauma .....	92
Tabela 9 - Requisitos de QoS em serviços de emergência em Telemedicina.....	92
Tabela 10 - Requisitos de QoS em serviços de e-Health.....	94
Tabela 11 - Comparativo dos resultados da busca sistemática.....	105
Tabela 12 - Diversidade de fatores de impacto na QoE .....	108
Tabela 13 - Representação do mapeamento entre Fator R e MOS.....	110
Tabela 14 - Modelos de estimativa de QoE por tipo de serviço.....	112
Tabela 15 - Parâmetros da QoC no sensoriamento de sinais vitais .....	126
Tabela 16 - Exemplo de política de QoS e QoC .....	184
Tabela 17 - Exemplos de instâncias da KB usadas para a predição do MOS .....	187
Tabela 18 - Ações de adaptações de políticas .....	192
Tabela 19 - Sensação térmica em função da temperatura do ponto de orvalho.....	208
Tabela 20 - Equações para obtenção dos valores das variáveis metereológicas.....	209
Tabela 21 - Terminologia, classificação e valores de referência dos Sinais vitais .....	209
Tabela 22 - Configuração das rotas dos <i>Links</i> usadas nos experimentos .....	212
Tabela 23 - Resumo dos requisitos de QoS dos serviços de e-Health. ....	213
Tabela 24 - Políticas de QoS para os serviços de eHealth.....	214
Tabela 25 - Estatística descritiva do throughput: sinais ambientais....	215
Tabela 26 - Estatística descritiva da latência: sinais ambientais .....	216
Tabela 27 - Estatística descritiva da perda de pacotes: sinais ambientais .....	216

Tabela 28 - Estatística descritiva do throughput: sinais biomédicos...	217
Tabela 29 - Estatística descritiva da latência: sinais biomédicos.....	218
Tabela 30 - Estatística descritiva da perda de pacotes: sinais biomédicos .....	218
Tabela 31 - Estatística descritiva do throughput: imagens médicas....	219
Tabela 32 - Estatística descritiva da latência: imagens médicas.....	219
Tabela 33 - Estatística descritiva da perda de pacotes: imagens médicas .....	220
Tabela 34 - Estatística descritiva do throughput: teleconsulta.....	220
Tabela 35 - Estatística descritiva da latência: teleconsulta .....	221
Tabela 36 - Estatística descritiva da perda de pacotes: teleconsulta ...	222
Tabela 37 - Resultado do Teste t do throughput: sinais ambientais ....	242
Tabela 38 - Resultado do Teste t da latência: sinais ambientais .....	243
Tabela 39 - Resultado do Teste t da perda de pacotes: sinais ambientais .....	244
Tabela 40 - Resultado do Teste t do throughput: sinais biomédicos...	245
Tabela 41 - Resultado do Teste t da latência: sinais biomédicos .....	246
Tabela 42 - Resultado do Teste t da perda de pacotes: sinais biomédicos .....	247
Tabela 43 - Resultado do Teste t do throughput: imagens médicas ....	248
Tabela 44 - Resultado do Teste t da latência: imagens médicas .....	249
Tabela 45 - Resultado do Teste t da perda de pacotes: imagens médicas .....	250
Tabela 46 - Resultado do Teste t do throughput: teleconsulta .....	251
Tabela 47 - Resultado do Teste t da latência: teleconsulta .....	252
Tabela 48 - Resultado do Teste t da perda de pacotes: teleconsulta ...	253
Tabela 49 - Informações estatísticas do tempo de restauração da QoE256	

## LISTA DE ABREVIATURAS E SIGLAS

API – Application Programming Interface  
AF – Assured Forwarding  
AAL – Ambient Assisted Living  
ARP – Address Resolution Protocol  
BD – Base de Dados  
BE – Best Effort  
BGP – Border Gateway Protocol  
CoS – Class of Service  
CRM – Customer Relationship Management  
DAML – DARPA Agent Markup Language  
DF – Default Forwarding  
DHCP – Dynamic Host Configuration Protocol  
DiffServ – Differentiated Services  
DNS – Domain Name System  
DS – Differentiated Service Field  
DSCP – Differentiated Services Code Point  
DSRM - Design Science Research Methodology  
EC – Engenharia do Conhecimento  
ECG – Eletrocardiograma  
ECN – Explicit Congestion Notification  
EF – Expedited Forwarding  
EHR – Electronic Health Record  
ETL – Extract, Transform and Load  
FEC – Forward Error Correction  
FIFO – First in, first out  
FOAF – Friend of a Friend  
GC – Gestão do Conhecimento  
GRE – Generic Routing Encapsulation  
GSQR – Generic Service Quality Reporting Protocol for Terminals  
HAN – Home Area Network  
HMI – Human-Machine Interface  
HIMS – Healthcare Information and Management Systems Society  
HTTP – Hypertext Transfer Protocol  
IC – Intervalo de Confiança  
ICMP – Internet Control Message Protocol  
IEEE – Institute of Electrical and Electronics Engineers  
IHC – Interação Humano Computador  
IntServ – Integrated Services  
IP – Internet Protocol  
IPTV – Internet Protocol Television  
ISO – International Organization for Standardization

ISP – Internet Service Provider  
ITUT – International Telegraf Union Telecommunication  
JDBC – Java Database Connectivity  
JSON – JavaScript Object Notation  
KB – Knowledge Base  
KPI – Key Performance Indicators  
LLDP – Link-Layer Discovery Protocol  
MAC – Media Access Control  
MAPE-K – Monitor-Analyze-Plan-Execute-Knowledge  
MC – Mídia do Conhecimento  
MCP – Major Configuration Parameters  
MOS – Mean Opinion Score  
MPLS – Multi-Label Switching Protocol  
MPLS-TP – Multiprotocol Label Switching - Transport Profile  
MQTT – Message Queuing Telemetry Transport  
MUO – Measurement Units Ontology  
NSP – Network Service Provider  
OF – OpenFlow  
OOMS – Object-Ontology Mapping Systems  
OMS – Organização Mundial da Saúde  
ONF – Open Network Foundation  
OntoKEM – Ontology for Knowledge Engineering and Management  
OSPF – Open Shortest Path First  
OVS – Open Virtual Switch  
OVSDB – Open vSwitch Database  
OWL – Web Ontology Language  
PPGEGC – Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento  
PBC – Perceived Behavioral Control  
QC – Questão de Competência  
QoC – Quality of Context  
QoE – Quality of Experience  
QoE-Control – Quality of Experience Control  
QoE-Eval – Quality of Experience Evaluation  
QoE-ManApp – Quality of Experience Management Application  
QoP – Quality of Perception  
QoS – Quality of Service  
OFQuality – Quality of Service Configuration Module  
RaaS – Robotics as a Service  
RBC – Raciocínio Baseado em Casos  
REST – Representational State Transfer  
RDF – Resource Description Framework  
RFC – Request For Comments

SAWSDL – Semantic Annotations for WSDL and XML Schema  
SLA – Service Level Agreement  
SBC – Sistema Baseado em Conhecimento  
SE – Sistema Especialista  
SDN – Software Defined Networking  
SDNC – Software Defined Networking Controller  
SGBD – Sistema de Gerenciamento de Banco de Dados  
SNMP – Simple Network Management Protocol  
SP – Service Provider  
SPARQL – Simple Protocol and RDF Query Language  
SWRL – Semantic Web Rule Language  
TAM – Technology Acceptance Model  
TCP – Transmission Control Protocol  
TIC – Tecnologia da Informação e da Comunicação  
ToS – Type of Service  
TPB – Theory of Planned Behavior  
TRA – Theory of Reasoned Action  
UCUM – Unified Code for Units of Measure  
UDP – User Datagram Protocol  
UR – Umidade Relat  
URI – Uniform Resource Identifier  
UX – User Experience  
VLAN – Virtual Local Area Network  
VoIP – Voice over Internet Protocol  
VLAN – Virtual Local Area Network  
VPN – Virtual Private Network  
XMPP – Extensible Messaging and Presence Protocol  
XP – Extreme Programming  
WSN – Wireless Sensor Networks





## SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>33</b>
1.1 Contextualização.....	33
1.2 Problema de Pesquisa.....	35
1.3 Objetivos.....	37
1.3.1 <i>Objetivo Geral.....</i>	37
1.3.2 <i>Objetivos Específicos.....</i>	37
1.4 Ineditismo e Contribuições da Proposta.....	38
1.5 Aderência do Tema ao PPGE GC.....	40
1.6 Metodologia da Pesquisa.....	44
1.7 Publicações da Tese.....	46
1.8 Organização do Trabalho.....	48
<b>2 ENGENHARIA DO CONHECIMENTO.....</b>	<b>51</b>
2.1 Conhecimento Organizacional.....	51
2.1.1 <i>Processo de Criação do Conhecimento Organizacional.....</i>	52
2.2 Evoluções e Concepções da Engenharia do Conhecimento.....	56
2.3 Processo de Apropriação e Formalização do Conhecimento.....	58
2.3.1 <i>Aquisição do Conhecimento.....</i>	58
2.3.2 <i>Modelagem do Conhecimento.....</i>	59
2.3.3 <i>Representação do Conhecimento.....</i>	61
2.3.4 <i>Validação do Conhecimento.....</i>	62
2.4 Ontologia como Representação do Conhecimento.....	65
2.4.1 <i>Construtos Ontológicos.....</i>	65
2.4.2 <i>Representação de Ontologias.....</i>	66
2.4.3 <i>Regras Semânticas em Ontologias.....</i>	70
2.4.4 <i>Consultas em Ontologias.....</i>	71
2.4.5 <i>Metodologias para Construção de Ontologias.....</i>	73
2.4.6 <i>Tipos de Ontologias.....</i>	79
2.5 Considerações do Capítulo.....	80
<b>3 QUALIDADE DO SERVIÇO E QUALIDADE DE EXPERIÊNCIA.....</b>	<b>83</b>
3.1 Qualidade do Serviço em Geral.....	83
3.1.1 <i>Dimensões da Qualidade do Serviço em Geral.....</i>	84
3.2 Qualidade do Serviço em Sistemas de Comunicação.....	86
3.2.1 <i>Parâmetros de Qualidade do Serviço em Sistemas de Comunicação.....</i>	86
3.2.2 <i>Qualidade Associada a Classes de Serviços.....</i>	89
3.2.3 <i>Principais Soluções de Qualidade do Serviço.....</i>	89

3.2.4 Exemplos de Aplicações com Restrições de QoS.....	90
3.2.5 Requisitos de QoS para Aplicações de e-Health.....	91
3.3 Qualidade de Experiência .....	97
3.3.1 Relações entre Qualidade do Serviço e Qualidade de Experiência .....	97
3.3.2 Dimensões de Influência na Qualidade de Experiência .....	99
3.3.3 Métodos para Avaliação da Qualidade de Experiência .....	108
3.3.4 Representação do Conhecimento da Qualidade de Experiência .....	114
3.4 Qualidade do Contexto.....	125
3.4.1 Parâmetros e Mensuração da Qualidade do Contexto.....	126
3.5 Considerações do Capítulo.....	128
<b>4 REDE DEFINIDA POR SOFTWARE.....</b>	<b>131</b>
4.1 Rede Definida por Software e Arquitetura TCP/IP.....	132
4.2 Arquitetura SDN .....	133
4.2.1 Camada de Aplicação.....	134
4.2.2 Camada de Controle.....	134
4.2.3 Camada de Dados.....	135
4.3 Protocolo OpenFlow .....	135
4.3.1 Troca de Mensagens entre Controlador e Switch OpenFlow....	138
4.3.2 Suporte a Qualidade do Serviço com o Protocolo OpenFlow... ..	140
4.3.3 Mensuração das Métricas de QoS com o Protocolo OpenFlow .....	141
4.4 Considerações do Capítulo.....	148
<b>5 MODELO DE GERENCIAMENTO DA QUALIDADE DE EXPERIÊNCIA DO USUÁRIO .....</b>	<b>149</b>
5.1 Abstração do Modelo Proposto.....	149
5.2 Taxonomia das Dimensões de Influência na Qualidade de Experiência .....	151
5.2.1 Fatores Humanos.....	151
5.2.2 Fatores de Conteúdo.....	153
5.2.3 Fatores de Contexto.....	154
5.2.4 Fatores da Qualidade do Serviço .....	155
5.3 Competência de Estudo das Dimensões da Qualidade de Experiência .....	156
5.4 Modelo de Representação do Conhecimento da Qualidade de Experiência .....	157
5.4.1 Metodologia para a Construção do Modelo de Conhecimento ..	158
5.4.2 Modelo de Conhecimento para a Representação da QoE.....	159
5.4.3 Consultas na Base de Conhecimento .....	172
5.4.4 Regras de Raciocínio na Base de Conhecimento.....	176
5.5 Arquitetura de Gerenciamento da Qualidade de Experiência .....	182

5.5.1 Camada de Gerenciamento do Contexto.....	183
5.5.2 Camada de Gerenciamento do Serviço.....	183
5.5.3 Camada de Gerenciamento SDN.....	190
5.5.4 Camada de Controle.....	198
5.5.5 Camada de Dados SDN.....	203
5.6 Considerações do Capítulo.....	204
<b>6 AVALIAÇÃO DO MODELO DE GERENCIAMENTO DA QOE</b> .....	<b>207</b>
6.1 Cenário Experimental.....	207
6.1.1 Correlações da Temperatura e Umidade Relativa do Ar com Crises Asmáticas.....	207
6.1.2 Correlação da Temperatura do Ponto de Orvalho como Indicador de Desconforto Térmico Humano.....	208
6.1.3 Parâmetros e Valores de Referência dos Sinais Vitais.....	209
6.2 Ambiente Experimental.....	210
6.2.2 Provedor do Serviço e Provedor de Rede.....	211
6.2.3 Protocolos de Criptografia e Roteamento.....	212
6.3 Resultados Experimentais.....	214
6.3.1 Análise Descritiva dos Dados.....	215
6.3.2 Inferência Estatística: Intervalo de Confiança.....	222
6.3.3 Inferência Estatística: Teste de Hipóteses.....	240
6.3.4 Tempo Médio para Restaurar a QoE.....	254
6.4 Considerações do Capítulo.....	257
<b>7 CONSIDERAÇÕES FINAIS</b> .....	<b>259</b>
7.1 Conclusões.....	259
7.2 Trabalhos Futuros.....	263
<b>REFERÊNCIAS</b> .....	<b>267</b>
<b>APÊNDICES</b> .....	<b>295</b>



# 1 INTRODUÇÃO

O presente capítulo apresenta a proposta da pesquisa e a forma com que a mesma foi conduzida. Inicialmente é realizada a contextualização do tema evidenciando os preceitos conceituais e técnicos necessários para a condução da pesquisa. Na sequência são apresentados o problema de pesquisa, os objetivos, o escopo e a delimitações da pesquisa, o ineditismo, a aderência do tema ao PPGEHC (Programa de Pós Graduação em Engenharia e Gestão do Conhecimento), a abordagem metodológica e a organização do trabalho.

## 1.1 CONTEXTUALIZAÇÃO

Até meados da década de 1950, a indústria de transformação era a que mais se destacava no cenário político e econômico mundial (MARTINS; LAUGENI, 2006). A partir de então, o setor terciário passou a assumir um papel preponderante no crescimento econômico de qualquer sociedade (FITZSIMMONS; FITZSIMMONS, 2005).

Para manterem-se competitivas, as organizações provedoras de serviços procuram maximizar a qualidade para que a percepção do cliente ao usar o serviço corresponda com suas expectativas. Porém, considerando as características peculiares dos serviços, a qualidade em serviços torna-se difícil de ser avaliada quando comparada com a qualidade dos bens manufaturados. Para facilitar nesta tarefa foram identificados fatores ou dimensões que auxiliam a qualificar o serviço prestado (JOHNSTON, 1995).

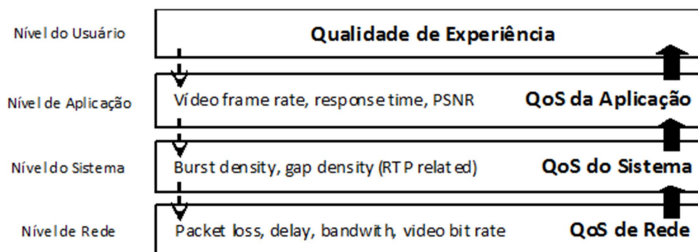
Embora muitos fatores determinantes (confiabilidade, rapidez, acessibilidade, disponibilidade, empatia...) possam ser utilizados para avaliar a qualidade de vários serviços, sempre haverá um conjunto específico de fatores determinantes da qualidade por categoria de serviço, conforme pode ser constatado nos exemplos apresentados a seguir.

No caso de um serviço físico, oferecido presencialmente, quando um cliente vai a um restaurante, por exemplo, ele tem expectativas quanto à qualidade dos alimentos, a variedade do cardápio, a limpeza e decoração do ambiente, ao preço, ao atendimento, entre outros. No uso de um serviço Web, por exemplo, é altamente desejável que o serviço tenha alta disponibilidade e excelente tempo de resposta. Em um sistema de comunicação com aplicações multimídia e de tempo real, por exemplo, é inadmissível perdas de sinais, atrasos e variações de atraso no envio dos dados. Enfim, são muitos os fatores determinantes usados

na tentativa de mensurar a qualidade. Para cada um dos exemplos citados, os parâmetros podem variar, são dependentes do domínio de aplicação, do contexto de uso do sistema e do usuário, bem como das inúmeras expectativas criadas pelo usuário em relação ao serviço.

Em sistemas de comunicação de dados, foco do estudo desta tese, o conceito de QoS (*Quality of Service*) foi introduzido pela ISO (1994) para mensurar a qualidade agregada aos serviços que uma rede de comunicação oferece, ou seja, refletir o quanto a rede é capaz de atender às expectativas dos seus usuários através dos serviços que ela disponibiliza. Inicialmente, este conceito foi focado na rede de comunicação de dados, posteriormente evoluiu para uma noção mais ampla contemplando as múltiplas camadas da interação rede/sistema/aplicação/usuário (Figura 1).

Figura 1 - Diferentes níveis de abstração da QoS e QoE



Fonte: Adaptado de Wu et al. (2009)

As camadas da Figura 1<sup>1</sup> contém parâmetros de QoS de aplicações multimídia e de tempo real (VoIP, IPTV, iTV) usados para mensurar a qualidade do serviço oferecida pelo sistema de comunicação. De acordo com o tipo de aplicação que se pretende avaliar, os parâmetros das camadas (rede, sistema, aplicação) são combinados e mapeados para predizer o nível de satisfação do usuário, ou seja, inferir a QoE (*Quality of Experience*). No entanto, a QoE, vai muito além de parâmetros técnicos, refere-se a avaliação das percepções e expectativas dos usuários no uso do serviço. Essa avaliação pode ser influenciada por vários aspectos (e.g. funcionais, técnicos, humanos) que podem interferir na qualidade percebida pelo usuário.

Embora o conceito de QoS tenha evoluído para uma noção mais ampla da qualidade (Figura 1) o que se observa na literatura

<sup>1</sup> A Figura 1 apresenta apenas um exemplo de parâmetros de QoS aplicáveis ao contexto de aplicações multimídia e de tempo real.

(GIACOBBE et al., 2016; LAGHARI et al., 2011) é que a maioria das propostas de prestação de serviços existentes preocupa-se com a qualidade sob o ponto de vista tecnológico e poucas focam na qualidade centrada nas necessidades dos usuários. Além disso, em função da grande quantidade de recursos computacionais no interior das instalações dos provedores, a gestão da QoS tem sido feita de modo uniforme, com base em parâmetros padronizados, muitas vezes considerando apenas a dimensão tecnológica. No entanto, os usuários têm percepções diferentes de qualidade baseadas em suas experiências. Em outras palavras, considera-se relevante investigar a prestação de QoS com base em uma abordagem mais complexa, onde a UX (*User Experience*) também é levado em conta. Mas,

- quais aspectos são importantes considerar para prover serviços cientes da UX?
- quais são as bases conceituais que amparam as propostas para a avaliação da UX?
- como a UX é obtida?
- como o conhecimento da UX é representado?

Estas são perguntas que oportunizam exaustivas pesquisas em busca de respostas, notoriamente em função de dois aspectos: qualidade do serviço (ponto de vista tecnológico) e a qualidade da experiência (ponto de vista do agente humano).

## 1.2 PROBLEMA DE PESQUISA

Para Silva, Dantas e Gonçalves (2014), as pesquisas que envolvem a UX devem ir além de aspectos tecnológicos e conhecimento do seu perfil. Para Fadel (2014) a UX deve ser avaliada de forma mais ampla, incluindo as necessidades cognitivas, estéticas e hedônicas. Essa avaliação só é possível se trabalhada de forma interdisciplinar e modelada como um constructo multidimensional (SILVA; DANTAS e GONCALVES, 2014). Contudo, o que se observa na literatura é que a maioria das propostas de QoE são construídas sem levar em consideração os diferentes tipos de conhecimentos do capital de clientes, não existe preocupação em modelar conjuntamente o conhecimento, o contexto de uso e o modelo de comunicação.

A grande maioria das propostas de QoE utiliza apenas os parâmetros de QoS como ponto de referência para mensurar tanto a qualidade do serviço quanto a satisfação do cliente (ALRESHOODI; WOODS, 2013; TSOLKAS et al., 2017). Foi observada uma excessiva

preocupação em saber se o meio físico, a rede, e os elementos envolvidos na propagação dos sinais estão entregando o serviço com qualidade, sem a preocupação de entregar o serviço de acordo com as necessidades estéticas e hedônicas dos usuários. Com base nessa realidade esta pesquisa visa contribuir para promover mudanças, incorporando outras dimensões na gestão da QoE, notoriamente aquelas relacionadas a fatores humanos.

Como ponto inicial, as dimensões de influência na QoE foram definidas a partir da compreensão de seu conceito. A partir de alguns conceitos extraídos da literatura (ETSI, 2010; FIEDLER; KILKKI; REICHL, 2009; ITU-T, 2008) pôde-se inferir que para ofertar serviços orientada a UX devem ser considerados o conhecimento do usuário, da rede que provisiona o serviço, do conteúdo que é ofertado e do contexto de uso do serviço. E aqui se evidenciou um grande desafio da Engenharia do Conhecimento, modelar conjuntamente o contexto, os tipos de conhecimentos e o modelo de comunicação. Esses três elementos formam a base conceitual para a modelagem de sistemas de conhecimento.

Para uma gestão bem sucedida da QoE é necessário uma compreensão profunda e abrangente das múltiplas dimensões da percepção humana sobre a qualidade e dos fatores de influência contemplados nessas dimensões (BARAKOVIĆ; SKORIN-KAPOV, 2013). A fim de compreender a QoE para um determinado serviço e como instrumentalmente estimá-la, é preciso levar em consideração os diferentes fatores e parâmetros que podem impactar a qualidade do serviço em questão e mapeá-los, no entanto, existe uma série de dificuldades associadas ao mapeamento:

- A lista de potenciais fatores e parâmetros é extensa para cada tipo de serviço.
- Alguns fatores relacionados com o contexto do usuário e uso do sistema são difíceis e, em alguns casos, impossível, de medir instrumentalmente.
- A QoE é um constructo multidimensional, assim o mapeamento dos fatores e parâmetros é uma proposta complexa.

Além disso, as arquiteturas de provisão de serviços são orientadas a tecnologia, usando uma abordagem da rede para o usuário, enquanto o foco deveria ser do usuário para a rede. Para tentar contemplar ao máximo a UX no uso de um serviço, é necessário prover mecanismos na rede de comunicação para que os serviços sejam oferecidos cientes da



UX. Com base nisso, evidenciou-se a SDN (*Software-Defined Networking*), ONF (2012), como uma abordagem flexível para gestão dos recursos da rede.

A SDN tem sido usada como infraestrutura de rede capaz de ser programada sob demanda para gerir a QoE no uso do serviço, como por exemplo, na avaliação do serviço de IPTV num sub sistema multimídia IP (HUONG-TRUONG, 2013); na gestão de falhas, com o propósito de validar um modelo cognitivo para a gestão autônômica da rede (KIM, 2013); na gestão dos centros de dados da google (JAIN et al., 2013), para prover serviços com qualidade aos usuários. No entanto, as pesquisas de SDN centradas na UX ainda são incipientes, prova disso são os resultados da nossa busca sistemática na literatura (Tabelas 1 e 2). A flexibilidade da SDN permite a programação da rede sob demanda, de acordo com o contexto (SILVA, et al., 2016d). Mas como o contexto deve ser modelado, está fora do escopo da SDN. Desta forma é necessário que o conhecimento usado pelos mecanismos da SDN (aplicações de controle e gerenciamento) seja representado univocamente para ser interpretado por máquina.

### **1.2.1 Pergunta de Pesquisa**

Diante de todo o cenário apresentado formulou-se a seguinte pergunta de pesquisa:

- Como prover serviços diferenciados aos usuários considerando que as entidades que gerenciam e provisionam os serviços estão situadas em diferentes níveis de abstração?

## **1.3 OBJETIVOS**

Como resposta à pergunta de pesquisa é apresentado o objetivo geral do trabalho, seguido dos objetivos específicos.

### **1.3.1 Objetivo Geral**

Elaborar um modelo de gerenciamento da QoE para a provisão de serviços cientes do contexto.

### **1.3.2 Objetivos Específicos**

- a) Selecionar uma abordagem semântica para a representação do conhecimento da QoE.

- b) Identificar os fatores de influência na QoE, formas de mensuração e representação do conhecimento, gerando uma taxonomia das dimensões a partir das pesquisas existentes.
- c) Especificar um modelo de representação do conhecimento da QoE, com base na taxonomia gerada.
- d) Elaborar um mecanismo que faz uso do modelo, projetando-o numa arquitetura orientada a UX, com o uso da SDN.
- e) Idealizar um cenário para a avaliação da abordagem proposta num Ambiente de Vida Assistido (AAL - *Ambient Assisted Living*).

#### 1.4 INEDITISMO E CONTRIBUIÇÕES DA PROPOSTA

Para verificar a originalidade da abordagem proposta, ou ainda, o ineditismo da mesma, no sentido de averiguar trabalhos relacionados ou similares ao proposto, foi realizada uma busca sistemática da literatura. O propósito era encontrar publicações interdisciplinares com o termo Qualidade de Experiência, procurando evidenciar:

- o domínio da aplicação da QoE;
- as bases conceituais das propostas existentes;
- as formas de mensuração;
- quais dimensões foram consideradas;
- a representação do conhecimento;
- se as propostas foram projetadas para a arquitetura SDN.

A busca sistemática foi feita em três bases de dados (*Scopus, Web of Science e IEEE Xplorer*), em datas distintas. Alguns filtros e combinações (termos da pesquisa) foram aplicados e os resultados refletindo os filtros estão sumarizados na Tabela 1. Inicialmente foram analisados as pesquisas da primeira busca sistemática, realizada em Junho de 2013.

A análise dos trabalhos, descrita na seção 3.3, permitiu evidenciar que apenas treze artigos (Tabela 10, seção 3.3) consideram a QoE não apenas no escopo das TIC (Tecnologia da Informação e Comunicação), mas agregam outras áreas do conhecimento.

A partir das análises, observou-se, ainda, que apenas três propostas usam abordagens semânticas para representar o conhecimento da QoE (FALLON; O’SULLIVAN, 2012; LAGHARI et al., 2010; LATRÉ et al., 2008), sendo que apenas numa delas (LAGHARI et al., 2010) a QoE é considerada como muito além da QoS.

Em Latré et al. (2008) é apresentada uma abordagem semântica

para otimização autônoma da QoE em redes de acesso multimídia. Contudo a abordagem foca apenas na QoS e não utiliza SDN, como forma de flexibilizar e prover uma arquitetura centrada nas necessidades do usuário.

De todos os trabalhos analisados, uma única proposta que apareceu nas duas bases de dados (KASSLER et al., 2012) utiliza SDN para atribuição de caminhos para prover QoE em serviços multimídia. Na proposta de pesquisa em andamento, Kassler et al. (2012) idealizaram um sistema que quando implementado irá permitir a negociação de parâmetros de comunicação do serviço e da rede entre usuários finais e, atribuir os caminhos de rede que são usados para fornecer fluxos multimídia, de acordo com a configuração do serviço contratado.

Tabela 1 - Busca sistemática da literatura

Base de Dados	TERMOS DA PESQUISA INICIAL – Junho de 2013							
	QoE <sup>2</sup> (Todos os anos)	QoE (2009 a 2013)	QoE Interdisciplinary	QoE and User Context	QoE, objective and subjective measures	QoE and subjective measures	QoE and usability	QoE and SDN <sup>2</sup>
<i>Scopus</i>	284	229	3	68	3	13	17	1
<i>Web of Science</i>	176	119	1	186	8	18	7	0
<i>IEEE Xplorer</i>	976	865	0	18	13	33	13	1
Base de Dados	TERMOS DA PESQUISA FINAL – Maio de 2017							
	QoE <sup>2</sup> (Todos os anos)	QoE (2009 a 2017)	QoE Interdisciplinary	QoE and User Context	QoE, objective and subjective measures	QoE and subjective measures	QoE and usability	QoE and SDN <sup>2</sup>
<i>Scopus</i>	651	582	4	173	7	29	22	72
<i>Web of Science</i>	473	416	1	550	12	33	13	98
<i>IEEE Xplorer</i>	1.231	1.186	0	18	13	33	13	35

Fonte: Base de dados IEEE Xplorer, Web of Science e Scopus

Ao longo da pesquisa desta tese, outros trabalhos foram pesquisados e apresentados no decorrer dos capítulos 3, 4 e 5. Contudo não foram encontradas trabalhos para a gestão da QoE com todas as características da abordagem proposta. Desta forma, a pesquisa realizada contempla uma lacuna identificada na literatura, trazendo tanto contribuição científica como contribuição tecnológica, a saber:

<sup>2</sup>Os termos QoE e SDN foram pesquisados, respectivamente, por *Quality of Experience* e *Software Defined Network*.

- Elaboração e avaliação de um modelo de gerenciamento da QoE para prover serviços em sistemas de comunicação de dados, trazendo como resultados:
  - Uma taxonomia multidimensional da QoE, pautada numa abordagem interdisciplinar;
  - Um modelo ontológico de representação do conhecimento da QoE;
  - Uma aplicação de gerenciamento de rede, com características semânticas e autonômicas, que aprende a experiência do usuário e provê serviços cientes de contexto;
  - Uma arquitetura de prestação de serviços orientada ao usuário, projetada para a Internet do Futuro.

Para manter a pesquisa atualizada em maio de 2017 uma nova busca sistemática foi realizada. Conforme pode ser observado na Tabela 1, praticamente não houve pesquisas explorando a QoE como um constructo interdisciplinar. Mas a mesma começou a ser estudada e associada ao contexto do usuário em SDN, em alguns trabalhos (HUONG-TRUONG, 2013; KIM, 2013; SILVA et al., 2015b, 2015c, 2016a, 2016c, 2016d), embora o foco na QoE ainda seja prematuro e precise evoluir em vários aspectos. Os demais trabalhos resultantes da busca sistemática são, oportunamente, referenciados ao longo da escrita deste documento.

## 1.5 ADERÊNCIA DO TEMA AO PPGEGC

Esta tese trata de um Modelo de Gerenciamento da QoE para a provisão de serviços cientes do contexto. Como tal, está inserida na área de concentração da Engenharia do Conhecimento (EC) e na linha de pesquisa Teoria e Prática em Engenharia do Conhecimento. No EGC, esta linha aborda metodologias e tecnologias da Engenharia do Conhecimento e da Inteligência Computacional e suas relações com a Gestão do Conhecimento (GC) e com a Mídia do Conhecimento (MC).

As atividades de pesquisa, formação e desenvolvimento da área de EC encontram sinergia com as áreas de GC e MC nos seguintes aspectos:

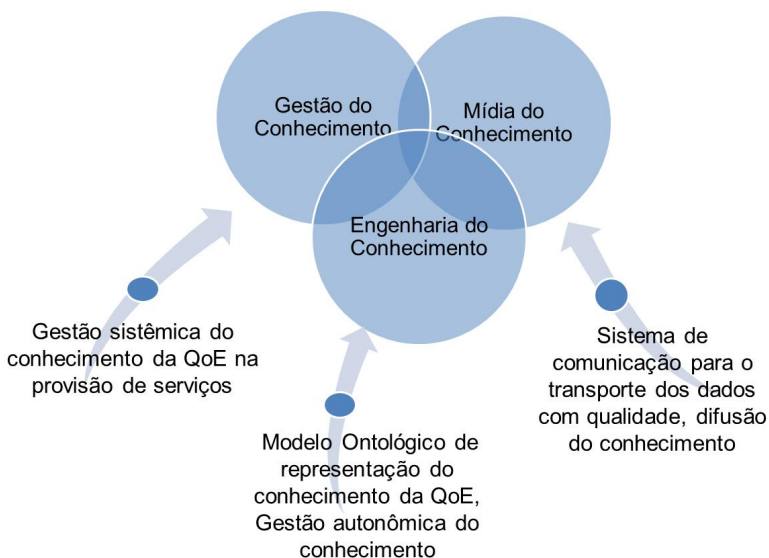
- i. Pesquisa e desenvolvimento de metodologias de identificação, representação e gestão de conhecimento;
- ii. Aplicação de sistemas de conhecimento à gestão do conhecimento organizacional (formalização, memória e tomada de decisão);

- iii. Aplicação de sistemas de conhecimento à interação homem-máquina, como suporte aos trabalhadores de conhecimento, inclusive na educação;
- iv. Aplicação de sistemas de conhecimento em mundos virtuais interativos visando à melhoria da eficácia e eficiência dos processos de treinamento e capacitação.

Considerando a sinergia entre as áreas descritas acima, constatou-se que esta tese está articulada tanto com a GC quanto com a MC. O alinhamento entre as três áreas do PPGEHC é apresentado na Figura 2.

Na Figura 2, enquanto a GC realiza a gestão sistêmica do conhecimento da QoE na provisão do serviço (e.g., cocriação do conhecimento; fidelização do cliente; diferenciação, avaliação e personalização do serviço), a EC preocupa-se em modelar, representar, armazenar e recuperar o conhecimento da QoE na KB, bem como gerar novos fatos a partir da aprendizagem de máquina, da avaliação do serviço e da captura do contexto da QoE. E para que o serviço possa ser provido e o conhecimento difundido, a MC provê recursos e artefatos tanto para a comunicação sintática (ponto de vista do agente humano) quanto para a comunicação semântica (ponto de vista dos agentes tecnológicos).

Figura 2 - Aderência da Tese ao PPGEHC



O contexto desta tese considera o “conhecimento como conteúdo

efetivado por agentes humanos ou artificiais em atividades de geração de valor científico, econômico, social ou cultural” (PACHECO, 2016, p. 21), considerado no EGC e, mais precisamente na área de EC, na noção de conhecimento cognitivista. Assim, em que pese a combinação de saberes das áreas de Engenharias, Ciência da Computação e Gestão de Serviços tenha espaço em cursos de pós-graduação que combinam as abordagens das disciplinas (Engenharia de Ontologias, Inteligência Artificial, Redes de Computadores, Qualidade da Informação e Gestão da Qualidade), no EGC, o modelo proposto se diferencia por considerar o conhecimento da QoE como recurso vital e dinâmico de agregação de valor organizacional; focar a QoE como um constructo multidimensional e interdisciplinar; preocupar-se tanto com o nível semântico quanto sintático do sistema de comunicação.

No histórico do EGC ainda não há tese ou dissertação com o tema Qualidade de Experiência, contudo foram encontradas 02 dissertações e 07 teses de doutorado que guardam alguma afinidade com a proposta deste trabalho, seja com algum assunto relacionado ao tema, aplicação em contexto similar e/ou abordagem metodológica. Na Tabela 2 são destacados os trabalhos considerados de contexto mais próximo ao desta tese.

Tabela 2 - Teses e dissertações do PPGEGC relacionadas com a proposta

<i>Título</i>	<i>Tema, Aplicação e/ou Metodologia</i>	<i>Autores</i>	<i>Área</i>	<i>Tipo</i>
Ontologia para representação semântica de indicadores de desempenho considerando aspectos de vaguidade, temporalidade e relacionamento entre indicadores	- Design Science Research	Freitas Junior e Gonçalves (2016)	EC	T
Identificação de critérios para avaliação de ideias: um método utilizando folksonomias	- Design Science Research	Rochadel e Souza (2016)	EC	D
Framework baseado em conhecimento para análise de rede de colaboração científica	- Design Science Research	Bordin e Gonçalves (2015)	EC	T
Cuida: um modelo de conhecimento de qualidade de contexto aplicado aos ambientes ubíquos internos em domicílios assistidos	- Qualidade de Contexto - Provisão de Serviços em AAL	Nazário e Dantas (2015)	EC	T
Modelo conceitual para criação, aplicação e avaliação de jogos educativos digitais	- Design Science Research	Jappur e Forcellini (2014)	GC	T
Avaliação de jogos voltados para a disseminação do conhecimento	-Experiência do Usuário	Savi e Ulbrich (2011)	MC	T
A Inovação em serviços e o processo de criação do conhecimento : uma proposta de método para o design de serviço	-Inovação em serviços - Processo de criação do	Manhães e Varvakis (2010)	GC	D

<i>Título</i>	<i>Tema, Aplicação e/ou Metodologia</i>	<i>Autores</i>	<i>Área</i>	<i>Tipo</i>
	conhecimento - Design Science Research			
Modelo de conhecimento para mapeamento de instrumentos da gestão do conhecimento e de agentes computacionais da engenharia do conhecimento baseado em ontologias	- Modelo de conhecimento - Ontologia para Engenharia e Gestão do Conhecimento	Rautenberg e Todesco (2009)	EC	T
Modelo de recuperação e comunicação de conhecimento em emergência médica com utilização de dispositivos portáteis	- Provisão de serviços de emergência médica	Manica e Dantas (2009)	EC	T

A sumarização da Tabela 2 permite evidenciar que há trabalhos que têm contexto de aplicação no mesmo segmento da presente tese, que foram estruturados com base em objetivos semelhantes e que disponibilizaram um processo metodológico aplicado na engenharia de ontologias nesta tese.

Os trabalhos de Nazário e Dantas (2015) e Manica e Dantas (2009) exploram a provisão de serviços cientes de contexto, aplicáveis em ambientes similares ao cenário de uso desta tese. Em Savi e Ulbrich (2008) é avaliada a experiência do usuário em jogos voltados para a disseminação do conhecimento, embora aplicado em contexto diferente, muitos aspectos da UX independem da aplicação. Em Manhães e Varvakis (2010), o tema aborda a inovação de serviços e o processo de criação do conhecimento, um dos desafios da tese aqui proposta consistiu, justamente, em prover um modelo de conhecimento para prover serviços projetados para a Internet do Futuro. E no que se refere aos procedimentos técnicos da pesquisa, os trabalhos de Freitas Junior e Gonçalves (2016), Rochadel e Souza (2016), Bordin e Gonçalves (2015), Jappur e Forcellini (2014) e Manhães e Varvakis (2010), utilizam a *Design Science Research Methodology* (PEFFERS et al. 2007), mesma abordagem metodológica adotada nesta tese. E por último, Rautenberg e Todesco (2009) propuseram e disponibilizaram um processo metodológico para a construção de ontologias, sendo o mesmo usado no desenvolvimento do modelo ontológico da QoE proposto neste documento.

Observando-se o histórico de trabalhos do EGC nota-se que a presente tese traz como contribuição específica um modelo de gerenciamento da qualidade de experiência para a provisão de serviços cientes de contexto e projetados para a Internet do Futuro. Nesse sentido, a mesma pode vir a ser subsídio para novas pesquisas na

inovação de qualidade em serviços e cocriação do conhecimento ciente da experiência do usuário, que podem ser instigados a partir deste estudo.

## 1.6 METODOLOGIA DA PESQUISA

Esta pesquisa caracteriza-se por ser tecnológica, no entanto, também possui um viés científico, uma vez que são investigadas as dimensões, os fatores e parâmetros de influência da QoE na provisão de serviços em sistemas de comunicação de dados. Desta forma são adotados os quatro critérios de classificação da pesquisa científica: quanto à natureza, quanto à forma de abordagem, quanto aos objetivos e quanto aos procedimentos técnicos (SILVA; MENEZES, 2005).

A ênfase da pesquisa consiste na análise quantitativa (TRIVINÕS, 1987), uma vez que são mensurados os aspectos da qualidade do serviço em diferentes níveis de abstração. Ao passo que a natureza da pesquisa é aplicada, pois “objetiva gerar conhecimentos para aplicação prática, dirigidos à solução de problemas específicos” (SILVA e MENEZES, 2001, p.20). No que se refere aos objetivos, a pesquisa é exploratória (WAZLAWICK, 2009), haja vista que são explorados e mapeados os diversos fatores e parâmetros de impacto relacionados com QoE. E quanto aos procedimentos técnicos, a pesquisa é bibliográfica e experimental, pois além da investigação do tema, é proposto um modelo para gerir a QoE na provisão de serviços, sendo o mesmo experimentado e avaliado em laboratório. Destaca-se, ainda, a utilização da *Design Science Research Methodology* (DSRM), Peffers et al. (2007), adotado como procedimento metodológico da pesquisa.

A DSRM é composta de seis etapas que seguem uma sequência nominal ou procedural, com ações de *feedback* entre elas, conforme apresentado na Figura 3(a).

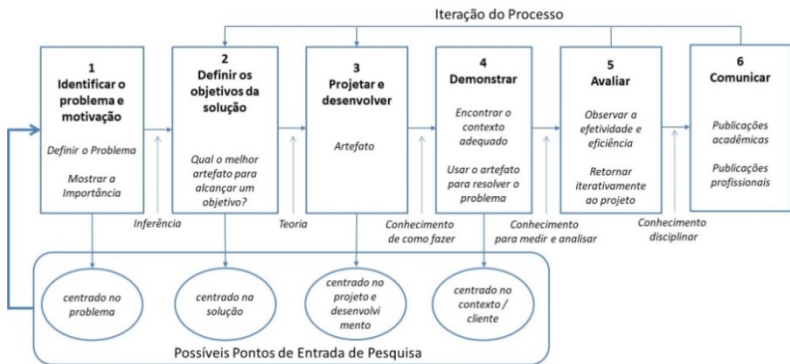
A Figura 3(a) sintetiza as 6 etapas da DSRM e os quatro possíveis pontos de entrada, respectivamente, sobre as quatro primeiras etapas do modelo. Os pontos de entrada permitem que a pesquisa possa ser iniciada centrada no problema; na solução; no projeto e desenvolvimento ou no contexto do cliente. O ponto de entrada da pesquisa desta tese é o centrado no problema, conforme apresentado na Figura 3(b).

Inicialmente foi realizada a definição do tema e através da busca sistemática foram definidos o problema e os objetivos da pesquisa. Na sequência foi realizada a revisão sistemática da literatura com os assuntos relacionados ao tema (Engenharia do Conhecimento,

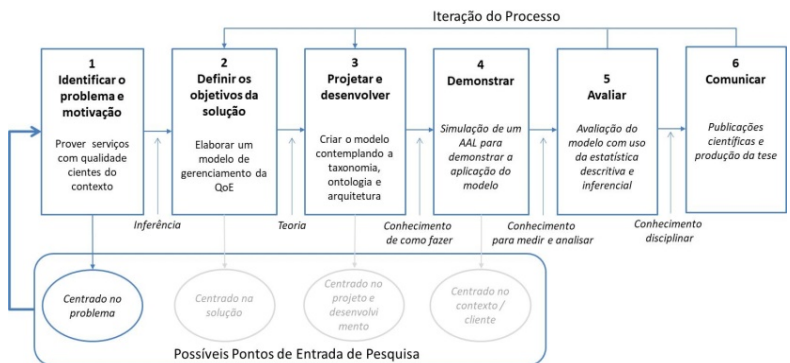


Qualidade do Serviço, Qualidade da Experiência, Qualidade de Contexto, Rede Definida por Software). Após, foi elaborado o modelo de gerenciamento da QoE, composto pela taxonomia com as dimensões e fatores de influência na QoE, o modelo ontológico de representação do conhecimento e a arquitetura de provisão e entrega de serviços cientes do contexto. Uma vez definido o modelo e caracterizado todos os seus componentes, foi idealizado um cenário para a provisão de serviços em um AAL. Os experimentos foram realizados em laboratório e os dados foram analisados usando as técnicas de estatística descritiva e inferencial. As publicações da pesquisa foram realizadas em várias etapas, conforme apresentação da seção a seguir.

Figura 3 - Modelo do Processo DSRM



(a)



(b)

Fonte: Peffers et al. (2007, p. 48)

## 1.7 PUBLICAÇÕES DA TESE

Nesta seção são apresentadas as publicações, como forma de elucidar a trajetória das pesquisas realizadas ao longo do desenvolvimento desta tese, embora todas as publicações estejam disponíveis no Apêndice A, separadas por tipo de publicação e com o respectivo estrato Qualis.

Assim que o tema foi definido iniciaram-se as pesquisas sobre os fatores de influência na QoE. Com base nos trabalhos existentes, foi realizado um mapeamento das dimensões de impacto na QoE, incluindo, os fatores humanos, conteúdo, contexto e tecnologia (SILVA; DANTAS, 2013). Dando sequência nesse levantamento e com o objetivo de explorar a QoE como um constructo multidimensional, fundamentado numa abordagem interdisciplinar, foi publicado um capítulo de livro, Silva et al. (2014). Como resultado é apresentado um modelo de QoE baseado na relação causa-efeito entre QoS e QoE. Um conjunto de dimensões, a partir do modelo proposto é analisado considerando um cenário multimídia, onde um serviço de vídeo sob demanda é fornecido, com base na experiência do usuário, adotando a flexibilidade do paradigma de SDN.

Para fornecer um embasamento teórico mais abrangente ao modelo, partiu-se para a investigação das bases conceituais das propostas de QoE existentes (SILVA et al., 2015a) e como resultados constatou-se que a maioria das propostas prioriza, principalmente, os aspectos tecnológicos, com isso negligenciando outras dimensões que interferem na QoE, notoriamente àquelas relacionadas a fatores humanos e contexto.

A hipótese inicial era que o modelo de QoE (SILVA et al., 2014) ao ser inserido numa plataforma de prestação de serviços seria capaz de fornecer subsídios para que algum mecanismo de inteligência pudesse associar aspectos hedônicos e pragmáticos do usuário com aspectos de QoS, conteúdo e contexto. Com base nisso duas contribuições foram idealizadas (arquitetura e a ontologia da QoE), Silva et al. (2015b), e assim surgiu a primeira versão da arquitetura proposta nesta tese. O objetivo da pesquisa consistiu no gerenciamento da QoE para a prestação de serviços ciente da experiência do usuário usando SDN. A arquitetura foi enriquecida com um motor semântico, com características autonômicas capaz de detectar e executar ações de adaptação de políticas em caso de degradações da QoE.

O artigo (SILVA et al., 2015b), selecionado entre os três melhores do evento, recebeu indicação a ser publicado num periódico

Francês (SILVA et al., 2016a). A arquitetura foi mantida, no entanto, a ontologia de QoE foi refinada e o motor semântico foi aprimorado com o ciclo de gerenciamento MAPE-K (*Monitor-Analyze-Plan-Execute over a shared Knowledge*), que juntamente com uma aplicação de controle SDN, contribuíram para a gestão da QoE com fina granularidade.

Em Silva et al. (2015c) a arquitetura original foi revisitada e novos componentes foram propostos para aprimorar a modelagem e o gerenciamento da qualidade de experiência. O mecanismo semântico foi capaz de aprender a experiência do usuário no uso de um serviço, detectar violações de métricas de QoS e fornecer informações para o controlador executar ações nos elementos da rede SDN, de acordo com políticas de QoS pré-definidas. Os resultados experimentais demonstraram que a proposta é viável e funcional e que o tempo entre a detecção da degradação na QoE e a adaptação de políticas nos recursos de rede não influencia na qualidade percebida pelo usuário.

Um dos objetivos do trabalho consiste em idealizar cenários para a avaliação do modelo num AAL. Assim, um ambiente real experimental foi configurado de forma que pudesse simular um AAL (SILVA et al., 2015d). No artigo foi investigada a viabilidade do uso de WSN (*Wireless Sensor Networks*) em saúde e proposto a implementação do IoT (*Internet of Things*) com o *ZigBee* para monitoramento das condições ambientais em um ambiente AAL ubíquo. Os dados obtidos a partir dos sensores (temperatura e umidade) foram combinados de modo a obter a temperatura do ponto de orvalho, usada como indicador de desconforto térmico dos usuários com pré-disposição a problemas respiratórios. As leituras das variáveis monitoradas foram transmitidas através da Internet e armazenadas no sistema supervisor ScadaBR a serem consumidas por aplicativos de cuidadores da saúde.

Ainda com relação a idealização de cenários em AAL configuramos um ambiente experimental, com o estudo de caso aplicado na Biometereologia Humana, Silva et al. (2016b). Com base nos trabalhos relacionados referenciados (SILVA et al., 2016b), constatou-se que a maioria das pesquisas sobre a Biometeorologia Humana usa dados de provedores de dados da saúde e do clima, para correlacionar e inferir os impactos das variáveis climáticas sobre as morbidades. Embora possam gerar indicadores para a gestão da saúde, as pesquisas precisam ser refinadas no sentido de considerar as leituras das variáveis meteorológicas no ambiente interno e, disponibilizar esses dados a serem avaliados juntamente com os dados clínicos dos pacientes. Visando contribuir com a Biometeorologia Humana foi proposta a

implementação da IoT, com o uso da *ZigBee*, para monitorar as condições ambientais de um AAL ubíquo, e armazenar os dados na nuvem, para serem consumidos por cuidadores da saúde.

Nos ambientes experimentais de AAL, a grande maioria dos dados é proveniente dos sensores, logo, é necessário avaliar a qualidade da informação do contexto fornecida por estes sensores, também chamada de QoC (*Quality of Context*). Partimos do pressuposto que a QoC também deve ser considerada para otimizar a experiência do usuário no consumo do serviço, pois se os sinais sensoriais não forem atuais, precisos e completos, por exemplo, a experiência do usuário pode ser afetada ou até mesmo induzir à decisões equivocadas pelos cuidadores da saúde. Com base nisso, um modelo de conhecimento de QoC foi proposto em Nazário et al. (2015). O modelo estrutura, organiza e integra o conhecimento encontrado na literatura, através de ontologia, permitindo o compartilhamento e reutilização do conhecimento. O modelo serviu de subsídios para o gerenciamento e distribuição do contexto nas pesquisas publicadas em Silva et al. (2016c, 2016d).

Outros trabalhos de coautoria, relacionados com assuntos desta tese, foram publicados em Assuiti et al. (2016); Martins, Silva e Dantas (2016); Vanelli et al. (2017a, 2017b). Assim como os artigos, Volpato et al. (2017a, 2017b, 2017c), com contribuições funcionais tecnológicas para a arquitetura proposta. Todos os trabalhos estão descritos no apêndice A.

## 1.8 ORGANIZAÇÃO DO TRABALHO

Nesta seção é descrita a organização do trabalho e os assuntos que nortearam a escrita de cada capítulo.

No Capítulo 2 é dada uma visão geral da evolução da Engenharia do Conhecimento, evidenciando as metodologias adotadas para capturar, modelar, representar e disponibilizar o conhecimento de modo a permitir sua apropriação e reuso em sistemas ou processos. Considerando que o Engenheiro do Conhecimento irá trabalhar na concepção, modelagem, representação e implementação de sistemas baseados no conhecimento, inicialmente é feito um resgate do processo de criação do conhecimento, tipos de conhecimentos e formas de conversão do conhecimento. Na sequência são apresentadas as etapas de apropriação do conhecimento e por último é enfatizada a representação do conhecimento com o uso de ontologias.

No capítulo 3 são apresentados os conceitos de qualidade em

serviços, incluindo a qualidade dos serviços em geral e qualidade em sistemas de comunicação de dados. A partir de então são evidenciados os diferentes conceitos da QoE; os fatores que podem influenciar na qualidade percebida pelo usuário e as diferentes abordagens para a mensuração e predição da QoE. Na sequência é investigado como o conhecimento da QoE tem sido representado e quais são as bases conceituais para a sustentação das propostas existentes.

No capítulo 4 é abordado o sistema de comunicação de dados usado para a disseminação do conhecimento e para a comunicação entre as entidades que provisionam e gerenciam a qualidade do serviço. Neste viés é apresentado o paradigma SDN, uma arquitetura inovadora para a provisão de serviços na Internet do Futuro.

No capítulo 5 é elaborado o modelo de gerenciamento da QoE, composto: (i) pela taxonomia com as dimensões e fatores de influência na QoE; (ii) pelo modelo ontológico, como forma de representação do conhecimento e (iii) pela arquitetura para a provisão e entrega de serviços ciente do contexto, utilizando o paradigma de SDN.

No capítulo 6 é realizada a avaliação do modelo proposto, sendo apresentados os componentes usados nos experimentos, o cenário e os resultados experimentais.

No capítulo 7 são descritas as conclusões e as contribuições da pesquisa. O capítulo também relaciona os objetivos propostos *versus* alcançados, como forma de evidenciar o cumprimento dos objetivos e metas do trabalho. Por último são apresentadas as pretensões de trabalhos futuros e algumas publicações já realizadas visando a extensão do trabalho quanto a melhoria no processamento e persistência do conhecimento.



## 2 ENGENHARIA DO CONHECIMENTO

Neste capítulo é apresentada uma visão geral da evolução da Engenharia do Conhecimento, evidenciando as metodologias adotadas para capturar, modelar, representar e disponibilizar o conhecimento de modo a permitir sua apropriação e reuso em sistemas ou processos. Considerando que o Engenheiro do Conhecimento irá trabalhar na concepção, modelagem, representação e implementação de Sistemas Baseados em Conhecimento, inicialmente é feito um resgate do processo de criação do conhecimento, tipos de conhecimentos e formas de conversão do conhecimento. Na sequência são apresentadas as etapas de apropriação do conhecimento e por último, é enfatizada a representação do conhecimento com o uso de ontologias.

### 2.1 CONHECIMENTO ORGANIZACIONAL

O estudo do conhecimento humano é tão antigo quanto a própria história do homem e tem sido o tema central da filosofia e da epistemologia desde o período grego. Para North (2010) a primeira ideia que vem junto com o conhecimento é a noção de contexto, desenvolvidos em diferentes contextos disciplinares, a saber:

“Na psicologia, o conhecimento é entendido, sobretudo, como conteúdo da memória, dando-se mais atenção à construção do modelo de conhecimento mediante redes semânticas associadas a uma pessoa. Na sociologia, o saber está associado, com maior amplitude, a instituições e grupos sociais. No mundo do trabalho não se tem um conceito unificado do conhecimento; não obstante, prevalece uma ótica individual, que o aproxima da psicologia, enquanto que na informática, fundamentalmente na área de inteligência artificial, o conhecimento se refere ao aspecto cognitivo da pessoa nos processos eletrônicos. Na economia o conhecimento se integra às doutrinas de gestão como fator de produção. Nessa disciplina, o foco não está tanto no saber individual, mas no coletivo, usualmente denominado conhecimento organizacional, aplicável à configuração de processos e de um quadro de referência que permita maior identificação e melhora no

desenvolvimento, distribuição e utilização dos conhecimentos por parte de uma empresa ou instituição” (NORTH, 2010).

Para Davenport (1998), o conhecimento é uma mistura fluida de experiência condensada, valores, informação contextual e *insight* experimentado. Essa mistura proporciona uma estrutura para a avaliação e a incorporação de novas experiências e informações. O conhecimento é aplicado na mente dos conhecedores, da qual se origina. Nas organizações, costuma estar embutido não só em documentos ou repositórios, mas também em rotinas, processos, práticas e normas organizacionais. O autor ressalta ainda que o conhecimento deriva da informação, da mesma forma que a informação deriva dos dados. Para que a informação se transforme em conhecimento, os seres humanos precisam fazer virtualmente todo o trabalho.

Segundo descrito em Fireston e McElroy (2001) o conhecimento organizacional é definido como a informação que passou por testes e avaliações em processos com o objetivo de eliminar os erros e de buscar a verdade. Isso é reforçado pelo registro e pela experiência com o processo de validação. Assim, o conhecimento organizacional é um subconjunto de informação.

### **2.1.1 Processo de Criação do Conhecimento Organizacional**

De acordo com os autores (NONAKA; TAKEUCHI, 1997) a estrutura conceitual básica sobre as formas de administração do processo de criação do conhecimento possui duas dimensões: dimensão ontológica e epistemológica.

Na dimensão ontológica, o conhecimento só pode ser criado por indivíduos, (NONAKA; TAKEUCHI, 1997). Uma organização, por si só, não pode criar conhecimento, seu escopo é apoiar os indivíduos e lhes proporcionar condições para tal. A existência do conhecimento organizacional é possível a partir de interações que permitem sua criação de forma individual e a disseminação para a organização em seu conjunto.

De acordo com a dimensão Epistemológica, há dois tipos de conhecimentos: explícito e tácito (NONAKA; TAKEUCHI, 1997).

O conhecimento explícito é objetivo e facilmente captado, codificado e compartilhado. Trata-se de um conhecimento transmissível em linguagem formal e sistemática, formalizado em textos, gráficos, tabelas, figuras, desenhos, esquemas, diagramas, etc., facilmente



organizado em bases de dados e em publicações em geral, tanto em papel quanto em formato eletrônico.

O conhecimento tácito é o que reside essencialmente na cabeça das pessoas. É um saber subjetivo, baseado em experiências pessoais e específicas ao contexto, e por tal motivo é mais difícil de ser formulado e comunicado; corresponde às habilidades inerentes a uma pessoa; é um sistema de ideias, percepção e experiência, difícil de ser formalizado, transferido ou explicado a outra pessoa.

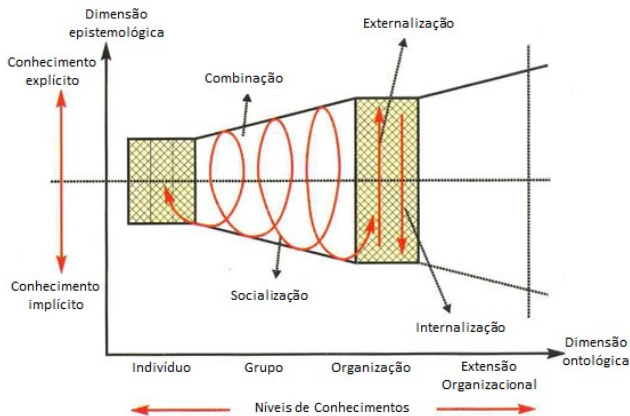
Para North (2010) a dimensão do conhecimento é disposta pelo conhecimento tácito e explícito. O conhecimento tácito representa o conhecimento pessoal de um indivíduo e se baseia em ideias, valores e sentimentos de cada pessoa. Pode ser visto como “juízo subjetivo e intuição que dão corpo ao conhecimento tácito que está profundamente ancorado nas ações e experiências de cada um” (NORTH, 2010). Esse tipo de conhecimento é muito difícil de formular e de transmitir já que está incorporado em cada pessoa. O conhecimento explícito é metódico, sistemático e existe de forma articulada, “é armazenado de forma impessoal, nos meios de comunicação, podendo ser gravado, cruzado e armazenado com a ajuda das Tecnologias da Informação e da Comunicação” (NORTH, 2010).

Um terceiro tipo de conhecimento, chamado de conhecimento cultural é apresentado em Popadiuk e Choo (2006). Esse conhecimento refere-se às crenças utilizadas para descrever e explicar a realidade, bem como convenções e expectativas empregadas para atribuir valor e significado a novas informações. Trata-se de um conhecimento não codificado, mas que é difundido através das relações em grupos. Embora Nonaka (1995) não tenha se referido ao conhecimento cultural, ele elaborou uma distinção entre o conhecimento coletivo e o individual.

Por último, Yakhlef (2005) descreve o conhecimento como acumulativo, construído sobre e a partir de conhecimentos criados anteriormente. Diferentemente de outras atividades organizacionais, a criação de conhecimento não precisa estar localizada em certo lugar e tempo e nem ser monitorada. Ideias criativas e insights não necessariamente são criados durante as atividades executadas. Em contraste com recursos físicos, ideias são transferidas e ao mesmo tempo não se perdem, pois o indivíduo as compartilha com outros e continua possuindo-as. Com base nessa distinção, os atores (NONAKA; TAKEUCHI, 1997) afirmam que há um pressuposto de que o conhecimento humano é criado e expandido através da interação social entre os conhecimentos tácito e explícito, o que se chama “conversão do conhecimento”, que acontece nos processos de socialização,

externalização, internalização e combinação (Figura 4).

Figura 4 - Espiral de criação de conhecimentos organizacionais



Fonte: North (2010, p. 49)

#### 2.1.1.1 Socialização do Conhecimento

O processo de socialização ocorre na conversão de parte do conhecimento tácito entre duas pessoas. “Esse modelo básico trata de uma geração limitada de conhecimento, já que esse conhecimento não se converte em explícito e, portanto não está à disposição da organização” (NORTH, 2010).

Na concepção de Rossato (2002), a socialização do conhecimento, normalmente, ocorre quando:

- Existe diálogo frequente e comunicação “face a face”.
- Os brainstorming, insights e intuições são valorizados, disseminados e analisados (discutidos) sob várias perspectivas (por grupos heterogêneos).
- Valoriza-se o trabalho do tipo “mestre-aprendiz” por meio da observação, imitação e prática acompanhada por um tutor.
- Há compartilhamento de experiências e modelos mentais via trabalho em equipe.

#### 2.1.1.2 Externalização do Conhecimento

O processo de externalização ocorre na conversão de parte do conhecimento tácito do indivíduo em algum tipo de conhecimento

explícito. “Nesse modelo básico se conseguem novos conhecimentos novos e valiosos para a organização, já que os conhecimentos tácitos são documentados mediante a chamada externalização” (NORTH, 2010).

O autor Rossatto (2002) afirma que a externalização transforma um conhecimento pessoal e subjetivo em outro transmissível e articulável. Normalmente, esse conhecimento conceitual acontece por meio de:

- Representação simbólica do conhecimento tácito através de modelos, conceitos, hipóteses, etc., construídos por meio de metáforas/analogias ou dedução/indução, em que se faz uso de toda a riqueza da linguagem figurada para tentar externalizar a maior fração possível do conhecimento tácito.
- Descrição de parte do conhecimento tácito, por meio de planilhas, textos, imagens, figuras, regras (por exemplo, nos sistemas especialistas), scripts, design history, etc.
- Relatos orais e filmes (gravação de relatos orais e imagens de ocorrências/ações).

#### 2.1.1.3 Combinação do conhecimento

O processo de combinação ocorre na conversão de algum tipo de conhecimento explícito gerado por um indivíduo e que será agregado ao conhecimento explícito da organização. Esse tipo de conversão também é abordado pelas teorias ligadas ao processamento da informação. “Com esse modelo básico o conhecimento total da organização não aumenta, já que só se resume ou se expressa de outra maneira o que é conhecido” (NORTH, 2010). No entanto, as informações estão a disposição da organização.

Para Rossatto (2002), a combinação representa o agrupamento de conhecimentos explícitos em um sistema de conhecimento. Esse tipo de conversão envolve também as tarefas de disseminação de conhecimentos explícitos pela organização, as quais podem ocorrer por meio de documentos, manuais, reuniões, treinamentos teóricos e outras formas concretas de divulgação oral ou escrita. Normalmente, esse conhecimento sistêmico acontece por meio do agrupamento (classificação, sumarização) e do processamento de diferentes conhecimentos explícitos.

#### 2.1.1.4 Internalização do conhecimento

O processo de internalização ocorre na conversão de partes do conhecimento explícito da organização em conhecimento tácito do indivíduo. Tal conversão também é abordada pelas teorias ligadas à aprendizagem organizacional. Nesse modelo básico, “o processo está relacionado com a aprender fazendo, com a criação de rotinas de ação ou a aquisição de habilidades” (NORTH, 2010).

Para Rossato (2002) a internalização resulta da interpretação de conhecimentos explícitos presentes em livros, manuais, documentos, aulas e palestras. Normalmente, esse conhecimento operacional acontece por meio da:

- Visualização e estudo individual de documentos de diferentes formatos (i.g. textos, imagens).
- Prática individual (e.g. aprender fazendo)
- Reinterpretação/reexperimentação de forma individual, de vivências e práticas (e.g. práticas e lições aprendidas).

A Figura 4 ilustra que quando os ciclos de conversão do conhecimento passam várias vezes pelos quatro processos (socialização, externalização, internalização e combinação) formam um espiral que serve para analisar e entender os mais diversos casos de criação e disseminação do conhecimento, tendo cada caso suas particularidades ou especificidades.

## 2.2 EVOLUÇÕES E CONCEPÇÕES DA ENGENHARIA DO CONHECIMENTO

A primeira geração dos SBC (Sistemas Baseados em Conhecimento), denominados de Sistemas Especialistas (SE), foi proveniente da Engenharia do Conhecimento Clássica, cuja abordagem metodológica era centrada na transferência do conhecimento do especialista, para formalismos de representação e métodos de raciocínio (STUDER; BENJAMINS; FENSEL, 1998; SCHREIBER et al., 2002).

Para a construção de SBC, o conhecimento era explicitado pelo especialista, codificado usando regras heurísticas e prototipado. O desenvolvimento consistia em reproduzir o mecanismo de solução de problemas dos especialistas, ou seja, o sistema era construído para pensar e resolver problemas como os agentes humanos (SCHREIBER et al., 2002).

A abordagem de transferência do conhecimento do especialista

para a concepção de SBC tornou-se limitada quanto ao desenvolvimento de sistemas em grande escala, e por consequência dificuldade de manutenção, custo e “principalmente porque mecanismos humanos de resolução de problemas são fortemente baseados em ponderações de incertezas, além de acoplados e dependentes dos variados modelos mentais que lhe dão suporte” (ABEL; FIORINE, 2013, p.4). Os sistemas baseados em regras (BUCHANAN; SHORTLIFFE, 1984), redes semânticas (DUDA et al., 1978) e redes bayesianas (FLORES et al., 2001) são exemplos que adotaram essa abordagem. Para resolver os problemas apontados, era necessária uma evolução no processo de engenharia do conhecimento aplicado no desenvolvimento dos SBC.

A partir dos anos 90, várias pesquisas individuais contribuíram para a melhoria do processo de construção dos SBC (GRUBER, 1995, 1998; SCHREIBER, 1992). Assim surgiu a Nova Engenharia do Conhecimento, que através das lições aprendidas com a evolução da Engenharia do Conhecimento, migrou da abordagem humanística (de transferência do conhecimento) para uma abordagem de modelagem com inteligência de máquina.

Na abordagem baseada em modelagem, o Engenheiro do Conhecimento deve considerar várias fontes geradoras do conhecimento (incluindo, pessoas, processos, arquivos e sistemas) e criar um modelo computacional dotado de inteligência com capacidades de resolução de problemas comparáveis a de um especialista de domínio (STUDER; BENJAMINS; FENSEL, 1998). O modelo não necessita inspirar-se em modelos cognitivos de representação ou inferência, isto é, simular processos cognitivos de um especialista, mas criar um modelo que oferece resultados similares.

De acordo com Abel e Fiorine (2013), como reflexo desta mudança, os termos sistemas especialistas, que faziam referência à imitação de um especialista humano, e sistemas baseados em conhecimento, que eram dirigidos por conhecimento humano, foram substituídos pelo termo mais geral de sistemas de conhecimento (SCHREIBER et al., 2000).

Um sistema de conhecimento é definido por Schreiber et al. (2000) como “qualquer sistema de informação que gerencie, armazene e/ou aplique conhecimento organizacional explicitamente representado”. Ainda, segundo Schreiber et al. (2002), o Engenheiro do Conhecimento deve se questionar e obter respostas do por quê, o quê e como o sistema de conhecimento deve ser desenvolvido, ou seja:

- Por quê? Consiste em entender o contexto organizacional e o ambiente em que o sistema de conhecimento será inserido.

- O quê? Consiste em descrever conceitualmente o conhecimento a ser aplicado na tarefa que será modelada pelo sistema de conhecimento.
- Como? Consiste em descrever os aspectos tecnológicos dos elementos que deverão compor a implementação computacional do sistema de conhecimento.

Para a apropriação do conhecimento, visando às respostas dos questionamentos acima descritos, o Engenheiro do Conhecimento necessita conhecer o processo de engenharia do conhecimento.

## 2.3 PROCESSO DE APROPRIAÇÃO E FORMALIZAÇÃO DO CONHECIMENTO

Abel e Fiorine (2013) acreditam que a Engenharia de Conhecimento propõe metodologias que podem ser aplicadas em cada etapa do processo de apropriação do conhecimento, incluindo a aquisição do conhecimento, modelagem conceitual, representação do conhecimento e sua validação quanto aos objetivos da organização.

### 2.3.1 Aquisição do Conhecimento

Esta etapa refere-se a captura do conhecimento necessário para a resolução do problema. Nesse processo de aquisição, o conhecimento é obtido por fontes humanísticas (especialistas de domínio), fontes manuais (e.g. documentos impressos/manuais/procedimentos/padrões) e/ou fontes computacionais (e.g. base de dados, arquivos, texto, sensores).

Seguindo a ótica da Gestão do Conhecimento, o autor Matté (2009) considera diferenças entre os termos “aquisição”, “extração” e “elicitação”, conforme segue:

- Aquisição: quando o conhecimento é adquirido de qualquer fonte, incluindo fontes humanas e fontes não humanas, independente se a mesma é física ou lógica.
- Extração: quando o conhecimento é adquirido unicamente de fontes não humanas, especificamente de fontes provenientes da Tecnologia da Informação (TI).
- Elicitação: quando o conhecimento é adquirido exclusivamente do especialista humano.

Para Abel e Fiorine (2013) a etapa de aquisição do conhecimento é considerada uma das mais críticas, pelo fato de “não existir uma

metodologia confiável de aplicação que resulte num modelo do domínio e dos processos cognitivos de seus agentes”. Assim, foram criadas algumas técnicas de aquisição do conhecimento para auxiliar nesse processo (QUINTELA, 2005), incluindo técnicas manuais, semi-automáticas e automáticas.

- Técnicas manuais: são oriundas das áreas da Psicologia e Análise de Sistemas, tais como entrevistas estruturadas ou não estruturadas, análise de protocolo (registro de discursos, diálogos), análise por interrupção, análise dirigida pelo contexto, observação direta, entre outras.
- Técnicas semiautomáticas: essas técnicas são provenientes da Psicologia (AGR – Análise de Grades de Relatos), na reutilização de modelos (KIF – *Knowledge Interchange Format*), linguagem universal para expressar e transportar conhecimento entre bases de conhecimento), e em ontologias reutilizáveis (Ontolingua, OCML). Pode ajudar o especialista ou ajudar o engenheiro de conhecimento (editores, documentadores, etc.).
- Técnicas automáticas: são baseadas na aprendizagem automática, em que o uso do computador detém todo o processo de compreensão do domínio, tais como, extração de conhecimento em documentos, mineração de dados, RBC (Raciocínio Baseado em Casos), aprendizagem de máquina.

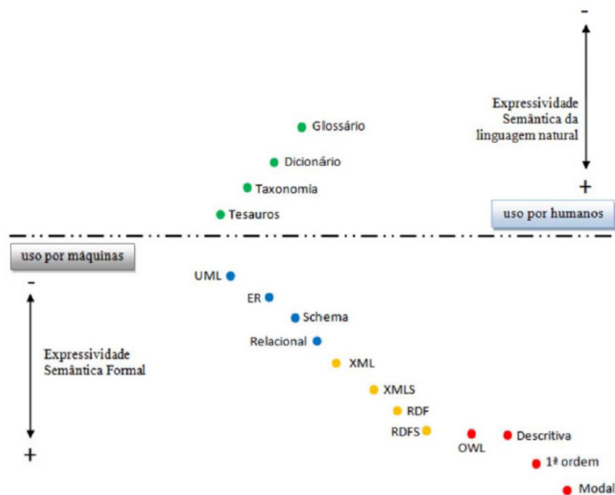
### 2.3.2 Modelagem do Conhecimento

A modelagem do conhecimento de domínio implica em capturar a estrutura estática de informações e tipos de conhecimentos, provenientes da etapa de aquisição do conhecimento. São obtidos os principais conceitos, instâncias, propriedades, relacionamentos, regras, procedimentos e métodos, incluindo os requisitos de sistemas. Essas informações darão origem ao modelo conceitual, no nível do conhecimento, que será posteriormente formalizado através de uma ou mais abordagens de representação, no nível simbólico (ABEL; FIORINE, 2013). Os atores descrevem ainda que, a modelagem conceitual se refere, portanto, a captura semântica do domínio, ou a construção da conceitualização ontológica.

Como não existe um padrão universalmente aceito para a engenharia de requisitos para a construção do modelo, o conhecimento pode ser modelado usando diferentes abordagens, com variações do grau

de expressividade semântica. Com o propósito de levantar as diferentes abordagens existentes para modelar e representar o conhecimento, os autores Almeida e Souza (2011), propuseram um espectro semântico de tais abordagens (Figura 5), ordenado de acordo com a expressividade semântica, separando o seu uso por humanos (linguagem natural) e por máquinas (semântica formal).

Figura 5 - Espectro semântico das abordagens usadas para modelar e representar o conhecimento



Fonte: Almeida e Souza (2011)

A partir da avaliação do espectro semântico de instrumentos para organização da informação proposta por Almeida e Souza (2011), foi possível sumarizar as abordagens, relacionando o seu uso e expressividade, conforme ilustrado na Tabela 3.

Tabela 3 - Classificação de instrumentos de representação do conhecimento quanto ao uso

<i>Instrumento</i>	<i>Uso</i>
Lógica (modal, 1ª. ordem, descritiva), OWL (lite, DL, full)	Web Semântica e representação do conhecimento para raciocínio automático de máquina
RDFS, RDF, XML schema, XML	Orientadas a web, modelos de dados semiestruturadas da Web
UML, entidade relacionamento, schema, Relacional	Orientados a sistemas de informação
Tesouro, taxonomia, dicionário, glossário.	Orientado a organização da informação pelos agentes humanos



### 2.3.3 Representação do Conhecimento

A Figura 5 demonstra a diversidade de abordagens que podem ser usadas para a representação do conhecimento, o que possibilita que os modelos conceituais possam ser criados sem qualquer ou com rico formalismo semântico, tanto em linguagem natural, quanto em linguagem formal. Se por um lado, as linguagens semânticas contribuem para o processamento automático dos agentes computacionais, por outro, inicialmente, os Engenheiros do Conhecimento tiveram dificuldade, pois o conhecimento não podia ser expresso usando a linguagem natural. Para este propósito, foram criadas as linguagens que disponibilizam primitivas de representação para capturar e estruturar conceitos de um determinado domínio, ao mesmo tempo em que retém a sua representatividade semântica (ABEL; FIORINE, 2013).

Considerando que os computadores necessitam ter acesso a coleções estruturadas de informações e de conjuntos de regras de inferência que o ajudem no processo de dedução automática para que seja administrado o raciocínio automatizado, é necessário o uso de ferramental criado para esta finalidade. Nesse viés, as ontologias têm sido utilizadas como forma de representação do conhecimento (seção 2.4), pois sobre elas podem ser criadas regras que permitem representar explicitamente a semântica dos dados. Através das ontologias é possível elaborar uma rede escalável de conhecimento humano, complementando o processamento da máquina e melhorando qualitativamente o nível dos serviços (BERNERS-LEE; HENDLER; LASSILA, 2012).

Conforme descrito por Almeida e Souza (2011), as principais linguagens de representação de conhecimento são baseadas em dois paradigmas dominantes, frames e lógica de descrição. No entanto iremos nos concentrar apenas em descrever a linguagem OWL, por ser a mais completa para formalismos ontológicos, conforme pode ser observado no Tabela 4.

Tabela 4 - Comparativo das características entre algumas linguagens formais usadas para representação do conhecimento

<i>Característica</i>	<i>RDF-S</i>	<i>SHOE</i>	<i>XOL</i>	<i>OML</i>	<i>OIL</i>	<i>DAML-OIL</i>	<i>OWL</i>
Conceito	S	S	S	S	S	S	S
Relações	S	S	S	S	S	S	S
Funções	?	N	S	?	S	S	S
Instâncias	S	S	S	?	S/N	S	S
Axiomas	?	N	N	S	N	N	S
Restrições de tipos	S	S	S	S	S	S	S

<i>Característica</i>	<i>RDF-S</i>	<i>SHOE</i>	<i>XOL</i>	<i>OML</i>	<i>OIL</i>	<i>DAML-OIL</i>	<i>OWL</i>
Restrições de Integridade	N	N	?	?	N	N	S

Fonte: Fleischmann e Dantas (2004)

A Tabela 4 apresenta um comparativo das características entre algumas linguagens usadas para representação do conhecimento. De acordo com os autores são usados “S”, “N”, “S/N”, “?” para indicar, respectivamente, quando a característica está presente na linguagem; quando a característica não está presente na linguagem; quando a característica não está presente na linguagem, mas pode ser simulada; nenhuma informação foi encontrada.

Através da Tabela 4 é possível observar que a linguagem mais completa para a representação do conhecimento é a OWL, pois além dos elementos necessários que compõem uma ontologia, oferece restrições de tipos e restrições de integridade.

### 2.3.4 Validação do Conhecimento

Neste trabalho está sendo considerado que a validação do conhecimento ocorre quando o conhecimento modelado, formalmente representado, é incorporado num sistema de conhecimento e colocado em produção na organização.

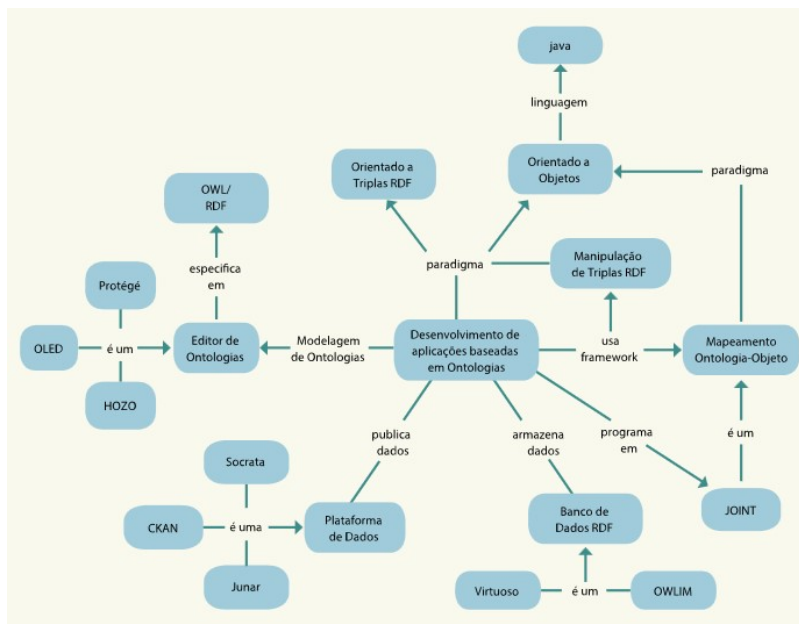
Quando o sistema de conhecimento é desenvolvido usando metodologias ágeis, tais como Scrum, XP (*Extreme Programming*), entre outras apresentadas por Pressmann (2010), versões incrementais do sistema são colocadas em produção para validação. Após o uso, as versões podem ser validadas. Contudo, o desenvolvimento de sistemas ou aplicações semânticas difere do desenvolvimento tradicional de sistemas de informação, pois ao invés da manipulação dos dados, agora, manipula-se o conhecimento.

Para a implementação de sistemas de conhecimento, cujo conhecimento esteja persistido em ontologias se faz necessário o uso de ferramentas específicas para o desenvolvimento de aplicações semânticas. Em Isotani e Bittencourt (2015), capítulo 5, é apresentado todo o ferramental existente para o desenvolvimento de aplicações semânticas, incluindo: Plataformas para publicação de dados; Frameworks para a manipulação de RDF; Bancos de dados RDF; Sistemas de mapeamento objeto-ontologia.

Para uma visão abrangente dos conceitos e principais ferramentas

de apoio ao desenvolvimento de aplicações baseadas em Ontologias, uma mapa mental (Figura 6) é apresentada por Isotani e Bittencourt (2015). Cada conceito é explorado pelos autores tanto com viés teórico quanto prático, objetivando esclarecer como desenvolver aplicações semânticas por meio do uso de dados abertos.

Figura 6 - Mapa conceitual com conceitos e ferramenta para o desenvolvimento de aplicações semânticas



Fonte: Isotani e Bittencourt (2015)

A Figura 6 sintetiza o esquema do processo de desenvolvimento de aplicações baseadas em ontologias, compreendendo:

- i. Modelagem da ontologia
- ii. Publicação dos dados
- iii. Armazenamento dos dados
- iv. Tipos de paradigmas de desenvolvimento
- v. Programação com suporte ao mapeamento ontologia-objeto
- vi. Framework para minupulação de triplas RDF e mapeamento ontologia-objeto

Para avaliar a abordagem semântica proposta nesta tese foi implementada uma aplicação semântica que manipula a ontologia

(módulo de gerenciamento da QoE). Para a implementação foram selecionadas algumas ferramentas apresentadas por Isotani e Bittencourt (2015):

- Escolha do Framework para manipulação do RDF (*Resource Description Framework*): nessa etapa deve ser feita a seleção de uma API para a manipulação de triplas RDF e a conexão com o banco de dados RDF. Essa API é equivalente ao uso do driver JDBC (*Java Database Connectivity*) no desenvolvimento de sistemas de informação objeto-relacional. Neste trabalho foi utilizado a API Sesame (BROEKSTRA et al., 2002) por ser considerada uma das mais completas.
- Escolha da Ferramenta para armazenar o RDF em algum banco de dados: nessa etapa deve ser feita a escolha de uma ferramenta para a persistência do banco de dados. Essas ferramentas são conhecidas como banco de dados RDF, triplestore, repositórios de ontologias, entre outros. No desenvolvimento de sistemas de informação essas ferramentas equivalem ao SGBD (Sistema de Gerenciamento de Banco de Dados). Porém, nem todas as ferramentas usam bancos de dados relacionais para armazenar as triplas em RDF, com exceção do Virtuoso (OPENLINKSW, 2015). O Virtuoso foi escolhido por ser um servidor multiprotocolo que provê acesso ao banco de dados relacional interno por meio de ODBC/JDBC, com adaptação para suportar o armazenamento em triplas RDF. No Virtuoso, o mapeamento das triplas RDF é feito em tabelas dentro do seu banco de dados relacional. Além disso, o Virtuoso oferece um motor de busca em SPARQL, convertendo as consultas em SPARQL feitas pelo desenvolvedor em SQL.
- Escolha da Ferramenta para o mapeamento objeto-ontologia: com uso de uma ferramenta com este propósito o desenvolvedor foca na lógica da aplicação, não sendo necessário escrever códigos para conexão com o banco de dados RDF, nem acessar os dados via triplas RDF, preservando as características de orientação a objetos da linguagem de programação. Um sistema de mapeamento objeto-ontologia cria classes na linguagem de programação correspondentes às entidades em uma ontologia. Portanto, instâncias dessas classes podem ser representadas na

aplicação como objetos. No desenvolvimento de sistemas objeto relacional o Hibernate é equivamente a uma ferramenta de mapeamento objeto-ontologia. Para a implementação da aplicação semântica proposta foi utilizado o JOINT, um *toolkit* Java de código aberto que provê uma série de funcionalidades para facilitar o desenvolvimento de aplicações baseadas em ontologias (HOLANDA et al., 2013). O JOINT permite manipular as ontologias em um repositório; consultar na linguagem SPARQL e manipular instâncias por meio do paradigma de orientação a objetos.

## 2.4 ONTOLOGIA COMO REPRESENTAÇÃO DO CONHECIMENTO

Na literatura a ontologia é descrita como “uma especificação formal explícita de uma conceitualização compartilhada”. Esse conceito foi definido gradativamente, por três autores. Gruber (1993) definiu ontologia como “uma especificação explícita de uma conceitualização”. Borst (1997) manteve o conceito original adaptando-o como “uma especificação formal de uma conceitualização compartilhada”. Studer, Benjamins e Fensel (1998) reuniu ambas as definições, explicando que “trata-se de uma especificação formal porque é legível para computadores; explícita, pois os conceitos, propriedades, relações, restrições e vocabulário estão explicitamente definidos em linguagem declarativa; compartilhada porque o conhecimento é consensual; e conceitualização, porque se trata de um modelo abstrato de algo que se deseja representar”.

Vale destacar que a palavra “conceitualização” não existe no português, sendo aplicada aqui para se referir a uma visão abstrata e simplificada do mundo que se deseja representar, considerando as entidades e relacionamentos nele presentes.

### 2.4.1 Construtos Ontológicos

Uma ontologia é composta por cinco construtos ontológicos (conceitos, instâncias, relações, funções, axiomas), Gruber (1993).

Os *conceitos*, também chamados de classe do domínio ou tarefas, representam os elementos de um domínio, definidas por um conjunto de atributos e seus possíveis valores. Exemplo: em uma ontologia usada para modelar pessoas, estudante e professor são entidades que representam os conceitos.

A *relação* é um tipo de interação (associação) entre os conceitos de um domínio. Exemplo: um estudante é uma subclasse de Pessoa; o nome do estudante é uma instância de estudante.

As *funções* são tipos especiais de relações que mapeiam um ou mais conceitos do domínio para um conceito único, assim, o n-ésimo elemento é único para os n-1 elementos precedentes. Exemplo: Pai de -, associando um ou vários filhos à seu pai.

Os *axiomas* são usados para a modelagem de sentenças que são sempre verdadeiras, sobre um domínio. Exemplos, se um estudante está matriculado na disciplina de “Estrutura de Dados” e a disciplina de “Lógica de Programação” é pré-requisito de “Estrutura de Dados”, então o aluno já cursou a disciplina de “Lógica de Programação”.

As *instâncias*, também chamados de indivíduos, representam elementos específicos. Exemplo: o nome da estudante é uma instância da classe Aluno. Os Indivíduos podem ter relações com outros indivíduos, ou com tipos primitivos (como strings ou números). Estas relações também são chamadas de propriedades.

Para Studer, Benjamins e Fensel (1998), muitas vezes a definição de ontologia é diluída, de forma que taxonomias são vistas como ontologias completas, entretanto, ontologias diferem de taxonomias em dois aspectos:

- Ontologias possuem uma estrutura interna rica e refletem o consenso de um grupo.
- Ontologias incluem todas as restrições relevantes entre classes, valores de atributos, instância e relações.

## 2.4.2 Representação de Ontologias

Existem duas maneiras de representar ontologias: através de representação formal e representação gráfica. Na representação formal a ontologia é especificada usando uma linguagem para ser interpretada por máquinas. Enquanto que na representação gráfica, a ontologia é descrita em linguagem natural para ser compreendido por agentes humanos.

### 2.4.2.1 Representação Formal de Ontologias

Para Isotani e Bittencourt (2015) na representação formal, normalmente, usa-se lógica de predicados, lógica descritiva ou linguagens baseadas em frames. A partir destas, são descritas outras

linguagens com rica expressividade semântica, conhecidas como linguagens de descrição de ontologias (HORROCKS et al., 2003). Os mesmos autores apontam que a linguagem mais difundida e mais completa para especificar ontologias é a OWL (*Ontology Web Language*). Essa afirmação confirma o estudo realizado por Fleischmann e Dantas (2004), sumarizado na Tabela 4.

A OWL (OWL, 2004) é a linguagem padrão da W3C para a codificação de ontologias. OWL tem características de várias famílias de linguagens de representação, incluindo principalmente lógica descritiva e frames. OWL é derivada do RDF (*Resource Description Framework*), e serializada em XML, provê construções para criar as classes, instâncias, relações entre classes ou instâncias, chamadas propriedades de objeto, relações entre classes/instâncias e tipos primitivos, chamadas propriedades de tipos de dados e axiomas — utilizados para definir o domínio e o alcance das propriedades. Todas estas construções podem ser denominadas genericamente de recursos, terminologia também usada em RDF.

A OWL é baseada em modelo de lógica que torna possível que os conceitos sejam descritos e bem definidos e permite que motores de inferência verifiquem se as declarações e as definições da ontologia são mutuamente consistentes e ajuda manter a hierarquia corretamente. Atualmente, existem duas versões de OWL sendo utilizadas, OWL 1.0 e OWL 2.0.

#### 2.4.2.1.1 Versão OWL 1.0

A OWL provê três perfis (OWL, 2004) para expressar significados e semântica, sendo eles OWL *lite*, OWL *DL* e OWL *full* (Figura 7).

O perfil OWL Lite abrange a expressividade de frames e lógica de descrição com algumas restrições. Por exemplo, a cardinalidade máxima ou mínima assume apenas os valores 0 ou 1. Apesar disso, a linguagem é dotada de riqueza semântica, sendo, por isto, ideal para usuários iniciantes e desenvolvedores que preferem frames à lógica de descrição. Trabalha hierarquia de classes e simples restrições (OWL, 2004).

O perfil OWL *DL* garante completude, decidibilidade e toda a expressividade da lógica de descrições, almejando satisfazer engenheiros de conhecimento familiarizados com esta tecnologia. Classes podem ser construídas por união, interseção e complemento, pela enumeração de instâncias e podem ter disjunções. Tipos são

mantidos cuidadosamente separados (por exemplo, uma classe não pode ser instância e propriedade ao mesmo tempo) (OWL, 2004).

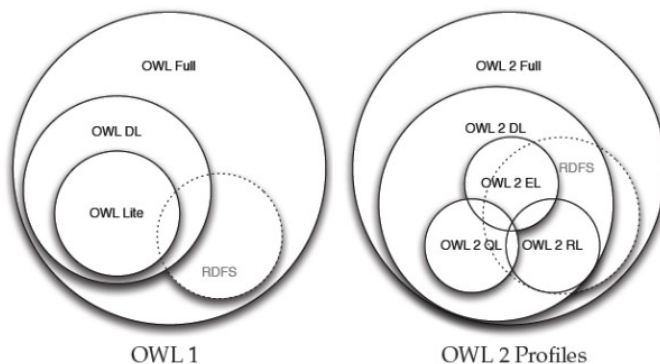
O perfil OWL *Full* fornece a expressividade de OWL e a liberdade de usar RDF, inclusive permitindo novas meta classes, já que elas são subclasses definidas em RDF Schema. Fazendo uso mais complexo, não há garantia de computabilidade com isso, há dificuldade de implementar software de “raciocínio” e inferência (OWL, 2004).

#### 2.4.2.1.2 Versão OWL 2.0

A OWL 2.0, adiciona novas funcionalidades em relação a OWL 1.0, incluindo, reforço das capacidades de anotação; cadeias de propriedade; tipos de dados mais ricos, intervalos de dados; qualificados restrições de cardinalidade; assimétrica, reflexiva e propriedades disjuntas e chaves. Foram adicionados três novos perfis, conforme ilustrado na Figura 7. Os perfis contemplam outras restrições, nomeadamente relacionadas com a especificação de quais elementos podem ser utilizados. Estas restrições visam tornar o perfil adequado para aplicações com um determinado conjunto de características.

De acordo com Motik et al. (2015), os perfis são independentes e foram projetados com diferentes níveis de expressividade a serem usados em diferentes cenários de aplicação. A escolha de qual perfil utilizar dependerá da estrutura das ontologias e das tarefas de raciocínio.

Figura 7 - Versões da linguagem OWL



Fonte: Bergman (2013)

O perfil OWL 2 EL foi projetado para uso em aplicações que empregam ontologias que contêm um grande número de classes e/ou



propriedades, que possuem objetos estruturalmente complexos e cujas inferências ocorrem principalmente ao nível terminológico (MOTIK et al., 2015).

O perfil OWL QL é particularmente adequado para aplicações que manipulam ontologias pequenas e cuja expressividade é reduzida, mas que possuem uma grande quantidade de indivíduos, tipicamente oriundos de uma base de dados relacional, e sobre os quais se pretende realizar consultas (*queries*). Este perfil é preferido quando a resposta da consulta é uma tarefa mais importante do que a do raciocínio. O perfil garante ainda que as respostas das consultas são válidas (*soundness*), completas (*completeness*) e obtidas de forma eficiente (MOTIK et al., 2015).

Perfil OWL 2 RL é destinado a aplicações que exigem raciocínio escalável sem sacrificar muito a expressividade (MOTIK et al., 2015). Esse perfil suporta tanto a semântica direta quanto a semântica baseada em RDF, sendo considerado o perfil ideal para o enriquecimento de dados especificados e conectados via RDF, pois pode ser implementado usando famílias de linguagens de regras (MOTIK et al., 2015), como por meio da utilização de RIF (*Rule Interchange Format*). Segundo Isotani e Bittencourt (2015), o perfil OWL 2 RL é o mais adequado para aplicações voltadas para dados abertos conectados.

OWL 2.0 traz novas funcionalidades já implementadas em várias ferramentas que trabalham com ontologias, tais como os motores de inferência FaCT++ (TSARKOV; HORROCKS, 2009), Pellet (SIRIN et al., 2007), APIs como a OWL API (SOURCEFORGE.NET, s.d.) e editores gráficos como o Protégé (BMIR, s.d.).

#### 2.4.2.2 Representação Gráfica de Ontologias

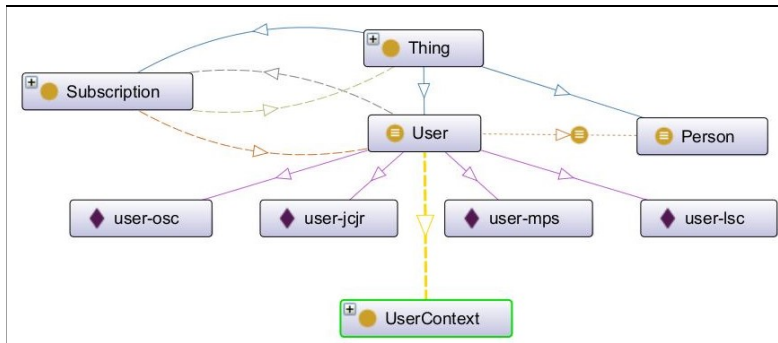
Em Isotani; Bittencourt (2015) é descrito que existem muitas maneiras de representar graficamente uma ontologia, uma vez que esta é composta principalmente por conceitos e suas relações. Algumas formas comuns para representar graficamente ontologias são grafos, UML, estrutura de árvore, além de outras (ISOTANI; BITTENCOURT, 2015).

Algumas ferramentas (e.g., Protégé, OntoEdit) permitem que as ontologias sejam especificadas em OWL 1.0 ou 2.0, através de componentes visuais. No caso do Framework Protégé, o código OWL é gerado de acordo com o perfil escolhido no momento da criação da ontologia.

A Figura 8 ilustra uma representação visual de uma ontologia criada no Protégé, com algumas classes (*User*, *Subscription*,

*ContextUser*) e instâncias da classe usuário. As classes *User* e *Subscription* se relacionam através das propriedades de objetos, “*hasSubscription*” e “*isSubscriptionOf*”. E a classe *User* se relaciona com a classe *Person* do vocabulário externo FOAF (BRICKLEY; MILLER, 2000) através da propriedade de dados “*hasPersonID*”.

Figura 8 - Simplificação visual de uma ontologia criada no Protégé



### 2.4.3 Regras Semânticas em Ontologias

As ontologias permitem ao agente computacional compreender a semântica do domínio, permitindo assim uma busca mais precisa do que uma abordagem sintática. As regras, no entanto, permitem o conhecimento mais rico do domínio, realizam a inferência de novos conhecimentos, agregam novas informações à base do conhecimento e fornecem um alto nível de flexibilidade.

As linguagens baseadas em regras tiveram a iniciativa centrada em uma linguagem de marcação baseada em XML. A XML pode ser utilizada para vários tipos de regras, como por exemplo, negócios e conversões, no entanto as regras especificadas em XML são carentes de formalismo semântico. Para isso, existe a SWRL (HORROCKS et. al., 2004), uma linguagem de regras que se tornou um padrão do W3C, projetada para a Web semântica, construída com a combinação da RuleML e OWL DL e OWL Lite (sub linguagens da OWL).

A SWRL propõe uma sintaxe abstrata em alto nível para definir cláusulas de Horn tanto em OWL DL quanto em OWL Lite, oferecendo assim uma maneira formal de se representar regras em ontologias OWL.

Uma ontologia OWL, em sua sintaxe abstrata, é formada por uma seqüência de axiomas e fatos. Axiomas podem ser de vários tipos, como

subclasses ou classes equivalentes. A SWRL propõe estendê-los para axiomas de regra, conforme sintaxe definida em Horrocks et. al. (2004).

A SWRL trabalha com *built-ins*, que são métodos definidos para serem usados nas sintaxes das regras. Os métodos contêm um conjunto de operadores matemáticos e rotinas para manipulação de string e dados. O prefixo convencional da SWRL é *swrlb*. A forma com que as regras são construídas e os métodos disponibilizados podem ser consultados em Horrocks et. al. (2004).

A representação de regras segundo a abordagem SWRL é feita da seguinte forma: antecedente (corpo)  $\rightarrow$  conseqüente (cabeça), onde tanto antecedente quanto conseqüente são expressos por conjunções de átomos representados por  $a1 \wedge \dots \wedge a_n$ . Existe uma implicação entre o antecedente e o conseqüente, de forma que quando as condições do antecedente são verdadeiras, então as condições do conseqüente também são verdadeiras.

Com base na ontologia da Figura 8, no Quadro 1 é apresentado um exemplo de regra SWRL para descobrir quais usuários são assinantes. No antecedente é verificado se existem instâncias de usuário e de assinatura e através do relacionamento entre as classes é averiguado se o usuário tem assinatura. Se as condições do antecedente são verdadeiras, então, o conseqüente é inferido como verdadeiro e o resultado do processo da inferência (instâncias de usuários assinantes) é feito na classe assinantes.

Quadro 1 - Exemplo de regra SWRL

```
User(?usr) ^ Subscription(?sub) ^ hasSubscription(?usr, ?sub) ^
swrlb:greaterThanOrEqual(?idade, 30) ^ foaf:age(?usr, ?idade) ^
swrlb:stringEqualIgnoreCase(?str, "female") ^ foaf:gender(?usr, ?str) ->
ux:Subscription(?usr)
```

#### 2.4.4 Consultas em Ontologias

Além da OWL API (2013), que fornece uma interface de programação para consultar e utilizar o conhecimento contido em ontologias OWL, existem outras formas de realizar consultas em ontologias. Uma das formas mais conhecidas e utilizadas é através do uso da SPARQL (*Simple Protocol and RDF Query Language*).

A SPARQL é o padrão proposto pela W3C (2008) para realizar consultas em grafos RDF, podendo ser utilizada em qualquer derivação de RDF, como por exemplo, em OWL. Uma tripla RDF (W3C, 1999) é definida por três componentes: sujeito, predicado e verbo, a saber:

- *Sujeito*: pode representar as instâncias (objetos) ou classes de uma ontologia, por exemplo.
- *Predicado*: pode representar as relações entre as classes (também chamadas de propriedades de objetos); as relações entre os indivíduos e os tipos primitivos de dados (também chamadas de propriedade de dados) ou algum tipo de recurso suportado na versão do RDF.
- *Objeto*: representam os valores das relações, incluindo as instâncias, classes ou tipos primitivos de dados.

O padrão SPARQL é composto de uma linguagem de consulta, um protocolo de consulta, e um formato de documento XML para reproduzir os resultados. A grande vantagem do uso de SPARQL é a sintaxe da linguagem de consulta, bastante familiar a SQL, com a exceção do uso de triplas RDF.

Um consulta em SPARQL é composta por alguns componentes básicos SELECT ... FROM ... WHERE { ... }, sendo necessário declarar o vocabulário e fonte de dados usadas nas consultas, além da recomendação do uso de prefixo para abreviação das URIs.

O Quadro 2 apresenta um exemplo de uma consulta SPARQL para buscar os usuários (instâncias) da classe User na porção ontológica apresentada na Figura 8. A estrutura da consulta contém o cabeçalho com os prefixos dos vocabulários (RDF, OWL, RDFS, XSD) e das ontologias utilizadas.

Através da relação entre as classes usuário e assinatura, feita pela propriedade de objeto *hasSubscription* são verificados os usuários que possuem assinatura, se os mesmos são do sexo feminino e possuem idade acima de 30 anos (reuso das propriedades de dados *gender* e *age* do vocabulário FOAF). O resultado é ordenado pelo nome das instâncias de usuário.

Quadro 2 - Estrutura de uma consulta SPARQL

```
#prefixo dos vocabulários utilizados na estrutura da consulta
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: http://www.w3.org/2001/XMLSchema#

#prefixo da ontologia que será consultada
PREFIX ux: <http://www.semanticweb.org/madalena/ontologies/ux/#>

#prefixo da vocabulário reutilizado
```

```

PREFIX foaf: <http://xmlns.com/foaf/0.1/#>

SELECT ?User
  WHERE {
    ?User ux:hasSubscription ?Subscription.
    ?Person foaf:gender "female"^^xsd:string.
    ?Person foaf:age ?age .
    FILTER (?age > 30)
  }
ORDER BY ?User

```

### 2.4.5 Metodologias para Construção de Ontologias

A construção de ontologias requer o uso de metodologias que auxiliem o Engenheiro do Conhecimento no processo de desenvolvimento. Atualmente existem algumas metodologias desenvolvidas para nortear os passos no desenvolvimento de ontologias.

No trabalho descrito por Silva, Souza e Almeida (2008), os autores apresentam uma comparação de metodologias e métodos para construção de ontologias e vocabulários controlados seguindo normas internacionais para construção de software e de vocabulários controlados.

Existe uma variedade de estratégias para desenvolvimento de ontologias, comprovando a hipótese de que grupos diferentes apresentam abordagens e características diversas, sendo direcionadas a diferentes propósitos e aplicações. (SILVA; SOUZA; ALMEIDA, 2008, p. 72).

No contexto das ontologias, algumas abordagens seguem um modelo de ciclo de vida, outras não. Nesse quesito, a que mais se destaca é a Methontology por ser praticamente completa em relação a um ciclo de desenvolvimento, não propondo apenas a fase de pré-desenvolvimento. (SILVA; SOUZA; ALMEIDA, 2008, p. 72).

Algumas abordagens dão mais ênfase em atividades de desenvolvimento, especialmente a implementação da ontologia (método Cyc e método 101), desconsiderando aspectos

importantes relacionados a gerenciamento do projeto, a estudo de viabilidade, à manutenção e à avaliação de ontologias. (SILVA; SOUZA; ALMEIDA, 2008, p. 72).

Em Rautenberg e Todesco (2009) é proposto um processo para a engenharia de ontologias, tirando proveito das melhores práticas das metodologias existentes, *Methontology* (GÓMEZ-PÉREZ; CORCHO; FERNÁNDEZ-LÓPEZ, 2004) *On-to-Knowledge* (SURE; STUDER, 2000) e *Ontology Development 101* (NOY; MCGUINNESS, 2008). De acordo com Rautenberg e Todesco (2009, p. 137), as metodologias são assim utilizadas:

*A Ontology Development 101* contribui com uma visão clara de como se dá um processo iterativo para o desenvolvimento de ontologias, Rautenberg e Todesco (2009, p. 137).

*On-to-Knowledge* contribui na especificação dos requisitos da ontologia, por meio do emprego de questões de competências como modo simples e direto para confirmar o propósito e o escopo de uma ontologia. Tal contribuição permite identificar antecipadamente, conceitos, propriedades, relações e instâncias, Rautenberg e Todesco (2009, p. 137).

*Methontology* contribui com alguns artefatos de documentação e na atividade de avaliação de ontologias, Rautenberg e Todesco (2009, p. 137).

Como resultado do estudo é apresentado e validado um processo para o desenvolvimento de ontologias, compreendendo as atividades de especificação, conceitualização, formalização, implementação e validação de ontologias e a utilização de ambientes computacionais (RAUTENBERG, TODESCO, 2009). O processo metodológico descreve quatro atividades, sendo cada atividade composta por:

### I. Especificação:

- a. Identificar o propósito da ontologia - identificar em que ambiente o modelo se insere, por que o modelo deve ser desenvolvido, entre outros;
- b. Identificar o escopo da ontologia - responder as questões gerais como “quem são os usuários”, “quais são as intenções de uso”, entre outras;
- c. Considerar o reuso de ontologias - verificar a existência de demais ontologias correlacionadas para fazer uso de conceitos já estabelecidos;
- d. Identificar as fontes de conhecimento - procurar por livros, artigos, dicionários, entre outras fontes, das quais se pode abstrair conceitualizações.

### II. Aquisição do conhecimento:

- a. Gerar as questões de competências – entrevistar especialistas de domínio na perspectiva que estes elaborem questões que a ontologia deva responder e que relacionem os termos, jargões e relacionamentos presentes no domínio;
- b. Listar os termos da ontologia - a partir das fontes de conhecimento e das questões de competências, enumerar termos comumente utilizados pelos especialistas de domínio;
- c. Agregar os elementos reutilizáveis - uma vez definidas as ontologias que tem aderência ao modelo proposto, capturar delas alguns elementos;
- d. Definir as classes - verificar na lista de termos, das fontes de conhecimento e dos elementos reutilizados se o entendimento de um termo remete a um conceito geral do domínio;
- e. Definir as propriedades das classes - verificar na lista de termos, das fontes de conhecimento e dos elementos 101 reutilizados se o entendimento de um termo remete a um dado necessário de algum conceito geral do domínio;
- f. Definir as relações entre classes - verificar na lista de termos, das fontes de conhecimento e dos elementos reutilizados se o entendimento de um termo remete a uma associação entre dois ou mais conceitos do domínio;
- g. Definir as restrições - para cada propriedade e relação de classes, verificar a existência de alguma regra que possa ser atribuída ao seu valor. Por exemplo, para a propriedade

idade, a restrição que esta não pode receber valores negativos;

- h. Criar as instâncias – para cada classe associar os termos tidos como instâncias, que caracterizam-se como exemplos concretos da classe em questão.

### III. Implementação:

- a. Valorar as propriedades das instâncias - definir os valores para cada propriedade de dados dos elementos da ontologia;
- b. Valorar as relações das instâncias – definir explicitamente os valores para cada relação entre as instâncias da ontologia;
- c. Valorar as restrições – para cada instância deve-se valorar as restrições presentes no domínio quanto aos valores possíveis para as suas propriedades de dados e para suas relações admitidas com a classe da ontologia.

### IV. Verificação:

- a. Verificação técnica da ontologia perante o domínio - verificar se a ontologia não expressa inconsistências em relação ao entendimento aceito sobre o domínio nas fontes de conhecimento;
- b. Verificação técnica da ontologia perante o framework de referência - visitar o propósito, o escopo e as questões de competências da ontologia para avaliar a consistência da ontologia frente os requisitos levantados;
- c. Verificação da ontologia - questionar os possíveis usuários da ontologia, quanto à utilidade, à precisão e a cobertura da ontologia na explicitação do conhecimento modelado.

Por considerar que o processo de Rautenberg e Todesco (2009) é adequado para o desenvolvimento do modelo proposto, a seguir são apresentadas as metodologias que serviram de base para a validação do processo.

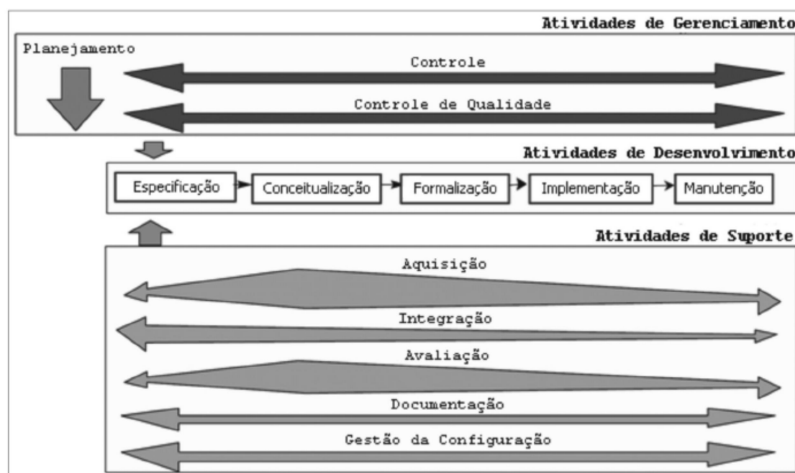
#### 2.4.5.1 Methontology

A *Methontology* (GÓMEZ-PÉREZ; CORCHO; FERNÁNDEZ-LÓPEZ, 2004), inspirada nas metodologias de Engenharia de Software e metodologias de Engenharia do Conhecimento, define um ciclo de vida e atividades de suporte para a construção da ontologia. O ciclo de vida é baseado na prototipagem de ontologias, de acordo com a evolução do



processo de desenvolvimento (especificação, conceitualização, formalização, implementação e manutenção). As atividades que compreendem a *Methontology* são destinadas ao gerenciamento, desenvolvimento e suporte do ciclo de vida de uma ontologia, conforme demonstra a Figura 9.

Figura 9 - Processo de desenvolvimento e ciclo de vida da Methontology



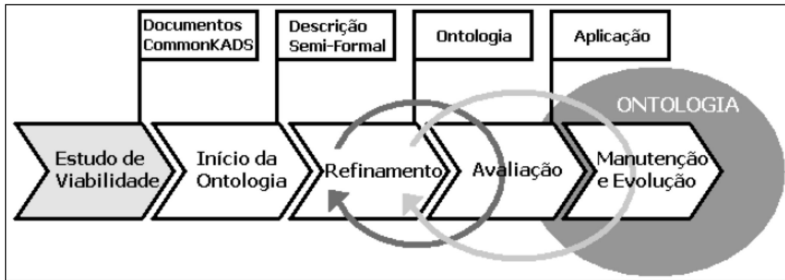
Fonte: adaptado de Gómez-Pérez, Corcho e Fernández-López (2004) por Rautenberg, Todesco e Gauthier (2009)

#### 2.4.5.2 On-to-Knowledge

*On-to-Knowledge* (SURE; STUDER, 2000) é uma metodologia que tem como propósito desenvolver ontologias para serem empregadas em Sistemas de Gestão do Conhecimento. O processo de desenvolvimento da ontologia no *On-to-Knowledge* contempla dois processos, denominados de meta processo de conhecimento e processo de conhecimento.

O primeiro processo, meta processo de conhecimento, engloba as cinco fases (estudo da viabilidade, inicialização, refinamento, avaliação da ontologia, aplicação e evolução da ontologia) apresentadas na Figura 10. Essas fases são descritas conforme apresentado em Rautenberg, Todesco e Gauthier (2009).

Figura 10 - Processo de desenvolvimento da metodologia On-to-Knowledge



Fonte: adaptado de Sure et al. (2000) por Rautenberg, Todesco e Gauthier (2009)

O estudo de viabilidade, apoiado pela metodologia CommonKADS (SCHREIBER et al., 2002), é realizado antes do início do desenvolvimento da ontologia. Este estudo visa identificar problemas e oportunidades de uma organização, objetivando mapear a real necessidade do desenvolvimento de uma ontologia. Após análise da viabilidade inicia-se o processo de construção da ontologia, onde todo o conhecimento é levantado e mapeado (e.g. domínio, objetivos, fontes de conhecimentos, atores, cenários, questões de competências, requisitos, ambiente de desenvolvimento da ontologia). Nessa fase, o conhecimento é semiformalizado. Segue-se para o refinamento, momento destinado ao desenvolvimento da ontologia a ser usada num sistema de gestão do conhecimento. Após o refinamento parte-se para a avaliação. Na avaliação é aferida a completude e a precisão da ontologia, mediante a documentação gerada durante o desenvolvimento da ontologia e um frame de referência, o qual pode corresponder às questões de competências enumeradas na fase “início da ontologia”. E por último, a fase de manutenção e evolução da ontologia, cuja competência e responsabilidade é da organização.

O segundo processo, processo de conhecimento, é um processo iterativo da gestão de conhecimento, aplicado na organização. Nele são realizadas as seguintes atividades, criação ou importação de dados; captura do conhecimento; recuperação e acesso ao conhecimento e utilização do conhecimento.

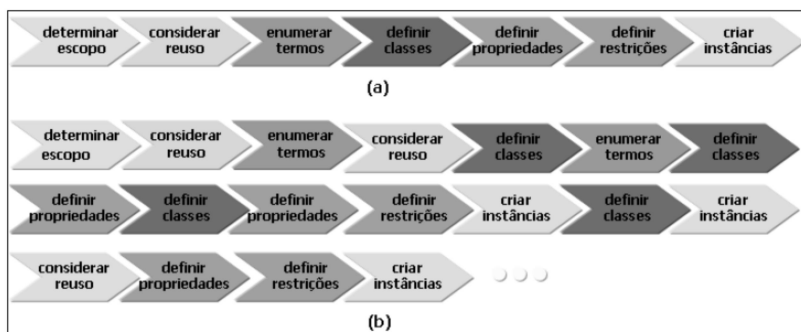
#### 2.4.5.3 Ontology Development 101

Por considerar que não existe uma metodologia adequada para o desenvolvimento de ontologias, os autores sugeriram e criaram o

processo *Ontology Development 101* (NOY; MCGUINNESS, 2008). O processo, graficamente representado pela Figura 11, consiste em um guia de passos interativos, os quais podem ser executados livremente na engenharia de ontologias.

A Figura 11 (a) apresenta os sete passos contemplados no processo e Figura 11 (b) ilustra um exemplo do emprego dos passos durante o desenvolvimento de uma ontologia.

Figura 11 - Processo de desenvolvimento do *Ontology Development 101*



Fonte: adaptado de Noy e Mcguinness (2008) por Rautenberg, Todesco e Gauthier (2009)

## 2.4.6 Tipos de Ontologias

As ontologias podem ser classificadas quanto ao grau de expressividade (leves e pesadas) e quanto a estrutura (alto nível, domínio, tarefa, aplicação).

De acordo com Isotani e Bittencourt (2015) na especificação das ontologias leves não há preocupação em definir detalhadamente cada conceito representado. São utilizados conceitos, relações e instâncias para definir a taxonomia que representa a relação hierárquica entre conceitos. Segundo os autores, esse tipo de ontologia vem sendo utilizado na Web para categorizar grandes quantidades de dados, principalmente em portais como Yahoo! e AOL.

Para Isotani e Bittencourt (2015) na especificação das ontologias pesadas são definidos detalhadamente cada conceito, a organização dos conceitos baseados em princípios bem definidos, uma definição formal da semântica entre os conceitos e suas relações, entre outros aspectos. São utilizados conceitos, relações, instâncias, axiomas e restrições para criação de bases de conhecimento reusáveis e compartilháveis.

Quanto a estrutura, as ontologias são classificadas em quatro tipos (GUARINO, 1998), descritas de acordo com Morais e Ambrósio (2007).

- Ontologias de alto nível: também chamada de ontologia de base, especificam os conceitos mais amplos (e.g., elementos da natureza, espaço, tempo, matéria, objeto, processos, ações, eventos), independentes de um problema específico ou domínio.
- Ontologias de domínio: especificam o vocabulário relacionado a um domínio específico (e.g., medicina, computação).
- Ontologias de tarefa: especificam tarefas genéricas, que podem contribuir na resolução de problemas, independente do domínio (e.g., diagnóstico, processo de vendas).
- Ontologias de aplicação: especificam conceitos tanto de um domínio particular quanto de uma tarefa específica, que muitas vezes são especializações de ambas as ontologias relacionadas (ontologia de domínio e de tarefa relacionadas com a aplicação).

Existem outras classificações de ontologias, tais como aquelas apresentadas por Morais e Ambrósio (2007).

## 2.5 CONSIDERAÇÕES DO CAPÍTULO

Neste capítulo foi apresentada a evolução da Engenharia do Conhecimento, iniciando com o processo de criação do conhecimento. Esta contextualização é necessária para que o Engenheiro do Conhecimento tenha uma visão abrangente dos tipos de conhecimentos existentes na organização e os diferentes processos de conversão do conhecimento, compreendendo a internalização, socialização, externalização e combinação.

Na sequência foram apresentadas as etapas de apropriação do conhecimento, contemplando a aquisição, a modelagem, a representação e a validação do conhecimento na organização. A ênfase foi dada para a representação do conhecimento com o uso de ontologias.

A construção de um modelo ontológico deve ser guiada por um processo, uma metodologia ou uma combinação de ambos. Desta forma foram descritos os processos e metodologias para o desenvolvimento de ontologias, permitindo que o Engenheiro do Conhecimento possa, inicialmente, conhecer o escopo e propósito da ontologia, os usuários, o

ambiente organizacional, as tarefas e os recursos tecnológicos disponíveis.

No capítulo, além da apresentação das metodologias e processos, foram apresentados os padrões OWL, SWRL e SPARQL, usados respectivamente para especificar o modelo; criar regras semânticas; realizar consultas. Outros aspectos foram apresentados, incluindo, formas de representações e tipos de ontologias.

O embasamento conceitual deste capítulo contemplou o primeiro objetivo proposto neste trabalho, uma vez que a documentação do mesmo possibilita, de modo objetivo, que o Engenheiro do Conhecimento se aproprie dos aspectos de construção, formalização e representação do conhecimento. O capítulo permitiu identificar que a ontologia é uma tecnologia chave para a representação do conhecimento, pelo fato do conhecimento formal ser compreensível por agentes computacionais, prover interoperabilidade e padronização dos conceitos em diferentes domínios administrativos (e.g., provedor de rede, provedor de serviço, provedor de conteúdo, provedor de dados), todas essas constatações são premissas necessárias para a modelagem do conhecimento da QoE.

O próximo capítulo fundamenta toda a abordagem do gerenciamento da qualidade de serviços, tanto do ponto de vista tecnológico quanto do ponto de vista do agente humano. Além disso, o capítulo fundamenta o funcionamento da arquitetura SDN, como alternativa para a provisão de serviços orientada a experiência do usuário.



### 3 QUALIDADE DO SERVIÇO E QUALIDADE DE EXPERIÊNCIA

Este capítulo aborda os conceitos de qualidade de serviço, incluindo a qualidade dos serviços em geral, QoS em sistemas de comunicação e QoE. A partir de então são evidenciados os diferentes conceitos da QoE e os fatores de influencia na qualidade percebida pelo usuário. Na sequência são apresentados as diferentes abordagens para a mensuração e predição da QoE. São demonstrados também, como o conhecimento da QoE tem sido representado e quais são as bases conceituais para a sustentação das propostas existentes.

#### 3.1 QUALIDADE DO SERVIÇO EM GERAL

Na literatura não existe um consenso entre os autores sobre o termo qualidade. Algumas definições focam na qualidade do produto, outras na definição da qualidade em serviços. Contudo Garvin (1992) contribui para uma compreensão mais ampla do termo, identificando e descrevendo as perspectivas do conceito de qualidade em cinco principais abordagens (transcendental, baseada no produto, foco na produção, Orientada ao valor, centrada no usuário).

- Transcendental: essa abordagem considera que a qualidade é uma característica de excelência, inata ao produto, estando a qualidade mais relacionada com a marca ou com a especificação do produto do que com o seu funcionamento.
- Baseada no produto: a qualidade é definida como um conjunto mensurável de atributos de um produto, que são mais facilmente identificados no caso de bens tangíveis do que no caso de serviços.
- Foco na produção: a qualidade é definida como conformidade com as especificações e padrões estabelecidos previamente, mesmo que essas especificações não correspondam às reais necessidades dos clientes.
- Orientada ao valor: relaciona a qualidade com o custo benefício do produto ou serviço, onde o valor para o cliente deverá ser maior que o preço.
- Centrada no usuário: o foco passa a ser satisfazer as necessidades do cliente, em que se procura conciliar às especificações do produto com as especificações do consumidor.

A pesquisa dessa tese está relacionada com a abordagem baseada no usuário. Esta abordagem contempla todas as demais, pois “quando uma empresa se preocupa com questões como qualidade, marca, conformidade com especificações de projeto, atributos desejáveis de um produto e valor oferecido maior que o preço, ela automaticamente está se preocupando com as necessidades do consumidor” (PALADINI, 1995).

Para Paladini (1995) a maioria das definições de qualidade será direcionada para a abordagem centrada no usuário, pois a qualidade percebida no uso do produto deve corresponder ou superar suas expectativas. Contudo, avaliar a qualidade baseada em expectativas é complexo (SLACK, et al., 1997), pois as expectativas e percepções dos usuários são diferentes e podem variar. No caso dos serviços, as percepções entre diferentes usuários podem variar ainda mais, devido ao fato dos serviços serem intangíveis. Além disso, a própria tendência de variabilidade dos serviços pode fazer que um mesmo usuário tenha diferentes percepções do mesmo serviço em diferentes ocasiões.

Para Parasuraman, Zeithaml e Berry (1988) qualidade em serviços é a relação entre as expectativas prévias do consumidor antes de comprar e experimentar um serviço e sua percepção posterior sobre a satisfação de tais expectativas. Assim, um serviço é considerado de qualidade se as expectativas do consumidor são atendidas ou superadas.

### **3.1.1 Dimensões da Qualidade do Serviço em Geral**

Para avaliar a qualidade em serviços, alguns autores têm definido fatores determinantes genéricos, que segundo eles podem ser aplicados a qualquer tipo de serviço. A partir da junção das propostas dos autores (PARASURAMAN; ZEITHAML; BERRY, 1985; JOHNSTON, 1995; GHOBADIAN; SPELLER; JONES, 1994; GIANESI; CORRÊA, 1994), é possível evidenciar os seguintes determinantes para mensurar a qualidade: tangibilidade, confiabilidade, presteza, garantia, empatia, rapidez, flexibilidade, acesso e disponibilidade. Embora boa parte desses determinantes possa ser aplicada a muitos serviços, há uma variabilidade muito grande de parâmetros específicos associados a serviços específicos, conforme pode ser observado na Tabela 5.

A Tabela 5 apresenta exemplos de serviços com parâmetros associados com a qualidade. Para cada um dos exemplos citados, os parâmetros podem variar, são dependentes do domínio de aplicação; das intenções, do contexto e do cenário de uso do utilizador.



Tabela 5 - Dimensões da qualidade em diferentes serviços

<i>Tipo de Serviço</i>	<i>Exemplos de parâmetros determinantes na qualidade</i>
Serviço de Restaurante	Variedade no cardápio, limpeza, decoração, atendimento...
Serviço de Hotéis	Localização, limpeza, atendimento, custo...
Serviço de Home Banking	Segurança, confiabilidade, autenticidade, disponibilidade...
Serviço Web	Disponibilidade, tempo de resposta...
Serviço Multimídia e de Tempo Real	Largura de banda, perda de sinais, atrasos, <i>jitter</i> , ruídos...

Há casos em que o próprio sistema prioriza os parâmetros de qualidade, como, por exemplo, um buscador de hotéis vai priorizar a localização ao invés do custo para encontrar hotéis de frente ao mar. Em outras circunstâncias só o usuário pode passar informações relevantes para a aplicação inferir serviços correspondentes com suas necessidades. Por exemplo, se o usuário informa ao buscador de hotéis que o hotel deve ficar próximo ao aeroporto, o conceito de próximo, depende de outras informações (forma de locomoção, idade), ou seja, depende do contexto, cenário e intenções de uso.

No contexto de serviços de comércio eletrônico, por exemplo, os autores têm evidenciado novas dimensões, conforme mostra a Tabela 6.

Tabela 6 - Dimensões da qualidade em serviços de comércio eletrônico

<i>Autor</i>	<i>Dimensões</i>
Dabholkar (1996)	Design do website; confiabilidade; entrega; facilidade de uso; divertimento; controle.
Zeithamal et al. (2002)	Segurança; comunicação; confiabilidade; capacidade de resposta; entrega.
Madu e Madu (2002)	Desempenho; elementos; estrutura; estética; confiabilidade; manutenção; segurança e integridade do sistema; confiança; capacidade de resposta; diferenciação e customização; política da loja online; reputação; garantia; empatia.
Surjadaja et al. (2003)	Segurança; interação; capacidade de resposta; informação; confiabilidade; entrega; e, customização.
Santos (2003)	Facilidade de uso; design do website; ligação com outros sites; estrutura; conteúdo; eficiência; confiabilidade; comunicação; segurança; incentivo; suporte.
Field et al. (2004)	Design do website; confiabilidade; segurança; portfólio de produtos.
Yang e Fang (2004)	Capacidade de resposta; confiabilidade; credibilidade; competência; acesso; cortesia; comunicação; informação; capacidade de resposta; design do website.
Fassnacht e Koesel (2006)	Qualidade gráfica; layout; atratividade da seleção; informação; facilidade de uso; qualidade técnica; confiabilidade; benefício funcional; benefício emocional.
Cristoal et al. (2007)	Design do website; serviço ao consumidor; garantia; gerenciamento de pedidos.

Fonte: Borges, Veiga e Teixeira (2014)

## 3.2 QUALIDADE DO SERVIÇO EM SISTEMAS DE COMUNICAÇÃO

Sempre que um serviço é oferecido usando uma estrutura da Internet ou de redes de computadores, algumas entidades fazem parte desse cenário, sendo elas: o usuário, o cliente, o provedor de serviços e o provedor de redes.

O provedor de serviços (SP - *Service Provider*) é a entidade que irá fornecer o serviço para o cliente. Por exemplo, num serviço de IPTV, o Provedor de Serviços é a entidade responsável pelo envio do serviço contratado. Para o transporte dos dados (dados, áudio, vídeo etc) é necessária o uso de tecnologias de acesso (e.g. Internet banda larga com fio ou sem fio) que prove acesso ao serviço, esse serviço é oferecido pelo Provedor de Serviços de Redes (NSP – *Network Service Provider*). O Cliente é uma entidade com permissões legais para se inscrever (se registrar) ao serviço oferecido pelo NSP. O usuário é uma entidade, autorizada pelo Cliente, que invoca/usa o serviço.

O roteamento é a principal forma adotada na Internet para a entrega de pacotes de dados entre *hosts*. Por definição, os dados são transmitidos sem QoS, ou seja, são de melhor esforço, isso significa que todos os dados transmitidos entre origem e destino recebem o mesmo tratamento nos roteadores do percurso. Havendo congestionamento, os roteadores colocam os pacotes em suas respectivas filas, respeitando a ordem de saída, conforme a ordem de chegada definida pela política FIFO (*First in, first out*). Como a capacidade da fila dos roteadores tem limitações, os pacotes são descartados, independente da criticidade dos dados. A simplicidade e robustez do modelo contribuíram para o sucesso da Internet, porém, a partir dele não é possível fornecer serviços diferenciados ou com algum grau de qualidade de acordo com as necessidades dos usuários.

### 3.2.1 Parâmetros de Qualidade do Serviço em Sistemas de Comunicação

Para prover garantias na transmissão de determinados fluxos de dados, a ISO (1994) introduziu o conceito QoS (*Quality of Service*). A QoS é usada para mensurar a qualidade do serviço oferecida por uma rede de comunicação, ou seja, refletir o quanto ela é capaz de atender às expectativas de seus usuários através dos serviços que ela disponibiliza ISO (1994).

O desempenho de um serviço provido pela rede é gerido através

de um conjunto de parâmetros de QoS (e.g., atraso, variação no atraso, perda de pacotes, largura de banda, taxa de bits), descritos a seguir.

- **Largura de banda (*bandwidth*):** a largura de banda é determinada pelo meio de transmissão que está sendo utilizado, pelos protocolos, pela distância entre os nós intermediários e pela velocidade de comutação nos nós intermediários.
- **Vazão (*Throughput*):** a capacidade de tráfego, também chamado de vazão de tráfego, é definida na RFC 5136 (CHIMENTO; ISHAC, 2008). Esta RFC contém várias definições para capacidade de tráfego em uma rede IP. Destas definições, a mais utilizada é a capacidade de tráfego em um enlace, que é definida como número máximo de bits na camada IP (que correspondem a oito vezes o número de octetos em todos os pacotes IP recebidos corretamente) que podem ser transmitidos corretamente entre dois hosts em um intervalo de tempo. Vale lembrar que o enlace corresponde a uma conexão direta entre dois hosts. Quando se fala de um caminho completo entre a fonte e o destino, a capacidade do caminho é a equivalente à capacidade do enlace de menor capacidade deste caminho, que pode variar com o tempo devido a variações no nível de congestionamento dos enlaces. A vazão em uma rede é a largura de banda efetiva ou a taxa de bits efetiva, ou seja, a quantidade de dados transmitidos com sucesso por unidade de tempo. Pode ser definida como sendo a diferença entre a taxa de bits do enlace e os vários overheads (sobrecarga). Em termos práticos, as aplicações geram vazões que devem ser atendidas pela rede. A vazão na maioria das redes sofre variações no decorrer no tempo. Em algumas situações, a vazão pode se alterar rapidamente devido às falhas nos nós da rede ou linhas ou devido ao congestionamento quando grandes fluxos de dados são introduzidos na rede.
- **Atraso (*delay*):** o atraso padronizado pela RFC 2679 (ALMES; KALIDINDI; ZEKAUSKAS, 1999a) é considerado um dos principais parâmetros de desempenho da rede. Denota o tempo decorrido para transmitir um bloco de dados de um emissor a um receptor. Na camada de aplicação, o retardo é a diferença de tempo (fim-a-fim) transcorrida entre a geração do dado no transmissor e a sua apresentação

no receptor. Essa diferença inclui parcelas referentes ao processamento nos nós intermediários (roteadores, swiches) e finais (*endpoints*), a disputa pelo acesso ao meio nos enlaces compartilhados e ao tempo de propagação no meio físico. Caso a conexão entre transmissor e receptor envolva múltiplos saltos, como é o comum em redes comutadas por pacotes, a soma de todos os retardos salto-a-salto mais o retardo de processamento deve ser igual ou inferior ao retardo fim-a-fim desejado. Normalmente deseja-se limitar algum parâmetro relativo à curva de distribuição do retardo, como um valor médio, máximo, ou um percentual.

- Variação do atraso (*Jitter*): corresponde a variação do atraso, padronizado pela RFC 3393 (DEMICHELIS; CHIMENTO, 2002). Em redes a pacotes, os fluxos são adicionalmente divididos em blocos de dados, e cada bloco é transmitido em sequência. Se a rede é capaz de enviar todos os blocos com uma sequência uniforme, então cada bloco deveria chegar ao destino após um atraso uniforme. Muitas redes não garantem um atraso uniforme para seus usuários. Variações em atraso são comuns. Os atrasos na transmissão são causados por muitos fatores, tais como: diferenças de tempo de processamento dos pacotes, diferenças de tempo de acesso à rede e diferenças de tempo de enfileiramento. Se as variações nos atrasos são devido às imperfeições do sistema na rede (software ou hardware), ou devido às condições de tráfego dentro da rede, estas variações são normalmente chamadas de *jitter*. O *jitter* é a variação observada no conjunto de valores de retardo de unidades de dados consecutivos. Como no caso do retardo, deseja-se limitar valores estatísticos da distribuição.
- Taxa de perdas de pacotes: padronizado pela RFC 2680 (ALMES; KALIDINDI; ZEKAUSKAS, 1999a) é a razão entre a quantidade de pacotes perdidos e a quantidade de pacotes enviados. Os pacotes são perdidos na rede por descarte nas filas dos nós intermediários, ou podem ser corrompidos por colisão com outros pacotes em enlaces compartilhados e ainda por variação no meio físico (tendo, nesse caso, relação com a taxa de erros de bits).

### 3.2.2 Qualidade Associada a Classes de Serviços

Um serviço pode ser contratado com garantias de QoS. Os provedores de maneira individual ou colaborativa podem prover serviços com diferentes níveis de QoS, utilizando classificações: ouro, prata e bronze, por exemplo.

Uma Classe de Serviço (CoS - *Class of Service*), também referida como CoS compreende uma classificação dada pelo provedor a aplicações com características de tráfego e desempenho semelhantes. Por exemplo, aplicações com requisitos de tempo real como chamadas de voz e outros serviços de telefonia podem ser classificadas usando a mesma CoS (BABIARZ; CHAN; BAKER, 2006). E os requisitos de cada CoS de um provedor, por sua vez, são declarados utilizando especificações de QoS. Os valores ideais dos parâmetros garantidos são escolhidos de modo que para o agregado de tráfego que a CoS representa a utilização do serviço seja satisfatória.

Os serviços podem ser oferecidos com diferentes níveis de QoS, de acordo com o valor que o consumidor está disposto pagar. No estabelecimento do SLA (*Service Level Agreement*), para cada requisito de QoS, um valor deve ser definido para ser usado como tolerável (valor exato, mínimo, máximo) durante a realização de um contrato eletrônico.

### 3.2.3 Principais Soluções de Qualidade do Serviço

Para implementar a QoS nos sistemas de comunicação de dados, geralmente são utilizados dois modelos, serviços Integrados (IntServ) e serviços diferenciados (DiffServ).

O modelo IntServ (BRADEN; CLARK; SHENKER, 1994) é baseado em reserva de recursos. A reserva é feita pelo protocolo de sinalização fim-a-fim RSVP (Resource Reservation Protocol) entre todos os *hosts* no caminho entre origem e destino. O modelo IntServ é caracterizado pela alocação de recursos para dois tipos de serviços: (i) serviços garantidos para aplicações que necessitam de um atraso constante, e (ii) serviços de carga controlada para aplicações que requerem segurança e destacam o serviço de melhor esforço (BE - *Best Effort*).

O modelo DiffServ (BLAKE et al., 1998) é a arquitetura mais difundida para prover QoS para aplicações multimídia e de tempo real. A implementação da QoS ocorre com base na definição de tipos de serviços sobre o tráfego da rede, cujo objetivo consiste em atender o nível de serviço desejado de cada aplicação, como por exemplo, voz,

vídeo e dados. Para prover diferenciação os pacotes são marcados de acordo com as classes de serviços pré-determinadas no campo DS (*Differentiated Service Field*) do cabeçalho IP.

O conjunto de CoS definidas por DiffServ, identificado pelo campo DSCP (*DiffServ CodePoint*) do pacote IP, é apresentado conforme descrito em (BABIARZ; CHAN; BAKER, 2006).

- *Default Forwarding* (DF): é a CoS padrão, correspondente ao serviço de melhor esforço ou Best Effort. Uma vazão mínima de tráfego é garantida, mas não há garantia de ordem ou entrega de pacotes.
- *Assured Forwarding* (AF): um conjunto de CoS, com diferentes níveis de garantia em atraso, variação de atraso, perda e vazão. Todos os pacotes que excedem os limites são descartados. Existem quatro classes, e três níveis de descarte para cada classe. Os serviços que as classes AF devem oferecer garantias, abrangem conferência multimídia (classe 4), streaming multimídia (classe 3), transações cliente-servidor (classe 2) e aplicações store and forward (classe 1).
- *Expedited Forwarding* (EF): é a CoS de maior prioridade, designada para serviços com baixa tolerância a atraso, variação de atraso, perda e vazão. A classe principal de serviço que EF deve oferecer garantias é a de telefonia.

### 3.2.4 Exemplos de Aplicações com Restrições de QoS

Independente da tecnologia de acesso aos dados, as aplicações multimídia e de tempo real demandam diferentes recursos de rede e geram classes diferentes de tráfegos. Para alocar recursos de rede às aplicações dos usuários, os principais requisitos de QoS devem ser considerados e traduzidos para parâmetros de desempenho da rede.

A QoS solicitada ao NSP depende dos requisitos específicos das aplicações. Aplicações de áudio e vídeo exigem a garantia de uma vazão mínima, baixo retardo e baixa variação de retardo, mas são ligeiramente tolerantes a erros e perdas. A perda de alguns quadros em fluxo de vídeo, por exemplo, não é suficiente para comprometer a percepção do usuário com relação ao conteúdo da informação audiovisual. Por outro lado, aplicações de transferência de arquivos como FTP são intolerantes a erros, mas não necessitam de garantias de retardo ou vazão, embora uma transferência rápida proporcione melhor satisfação ao usuário. A Tabela 7 apresenta os requisitos de QoS de algumas aplicações

multimídia e de tempo real.

Tabela 7 - Requisitos de QoS em diferentes aplicações

<i>Requisitos de QoS</i>	<i>Voz</i>	<i>FTP</i>	<i>E-mail</i>	<i>Vídeo Broadcast</i>	<i>Vídeo Interativo</i>
Largura de Banda	Baixa a Média	Baixa	Baixa	Alta	Alta
Descarte de Pacotes	Média	Média	Média	Média	Média
Atraso	Alta	Baixa	Baixa	Baixa	Alta
<i>Jitter</i>	Alta	Baixa	Baixa	Média	Alta

### 3.2.5 Requisitos de QoS para Aplicações de e-Health

De acordo com a HIMSS (*Healthcare Information and Management Systems Society*), o conceito de *eHealth* é descrito como qualquer aplicação de Internet utilizada em conjunto com outras TICs e focada na melhoria de acesso, eficiência, efetividade e qualidade dos processos clínicos e assistenciais necessários a toda a cadeia de prestação de serviços da saúde. Tal conceito engloba várias dimensões, que vão desde a entrega de informações clínicas aos parceiros da cadeia de atendimento, passando pelas facilidades de interação entre todos os seus membros, chegando a disponibilização dessa mesma informação nos mais difíceis e remotos lugares.

Para Skorin-Kapov e Matijasevic (2010), as aplicações típicas de *e-Health* podem envolver conferências multimídia, transmissão de parâmetros fisiológicos do paciente, transferência de imagens médicas de alta resolução, transmissão de dados clínicos, transmissão de dados administrativos e acesso a EHR (*Electronic Health Record*). Essas aplicações geram tráfego com diferentes requisitos de QoS, diferindo em largura de banda, interatividade em tempo real versus tempo não real e tolerância à perda de pacotes. Muitas vezes, um serviço de *e-Health* envolverá a transmissão simultânea de múltiplos fluxos de mídia, como, por exemplo, um sistema de emergência móvel, incluindo áudio/vídeo, imagens médicas e sinais de ECG.

Em Chu e Ganz (2004) é proposto um sistema móvel, em tempo real, para cuidados de teletrauma, para serviços pré-hospitalar não emergencial. O sistema foi implementada em laboratório onde vídeo em tempo real, imagens e sinais de ECG foram transmitidos simultaneamente entre o paciente e o médico. Os autores descobriram que os diferentes tipos de conteúdos (vídeo, imagens, ECG) necessitam de diferentes prioridades e níveis de sensibilidade quanto aos parâmetros de QoS e sugeriram limiares qualitativos, sendo “sim – para permitido”

e “não – para não permitido”, conforme ilustrado na Tabela 8.

Tabela 8 - Requisitos de QoS em protocolos de transmissão em sistemas de teletrauma

<i>Tipo de Mídia</i>	<i>Atraso fim-a-fim</i>	<i>Taxa de Perda</i>	<i>Erro de Transmissão</i>	<i>Prioridade</i>	<i>Integridade dos dados</i>	<i>Método de Transmissão</i>
Vídeo em tempo real	Não	Sim		Baixa	Baixa	UDP
Imagem médica	Sim	Não		Média	Alta	TCP
ECG		Não	Não	Alta	Alta	TCP

Fonte: Ullah, Fiedler e Wac (2004)

Nos experimentos, o vídeo foi apresentado com resolução de 320x240/160x120, sendo transmitido com o uso do protocolo UDP, enquanto que as imagens médicas e o ECG foram transmitidos com o protocolo TCP. A integridade dos dados foi considerada como uma métrica de desempenho.

Os parâmetros de QoS em aplicações de emergência na área da saúde também foram avaliados em Vouyioukas, Maglogiannis e Komnakos (2007). As aplicações são categorizadas em tempo real e quase em tempo real. Os requisitos de QoS são sugeridas em termos de *throughput*, atraso e variação do atraso (*jitter*) para diferentes aplicações de emergência médica, incluindo teleconsulta, telemonitorização, teleeducação, telediagnóstico e o acesso remoto a EHR (*Electronic Health Record*), ilustrados na Tabela 9.

Tabela 9 - Requisitos de QoS em serviços de emergência em Telemedicina

<i>Tipo de serviço</i>	<i>Throughput</i>	<i>Pequeno Atraso</i>	<i>Pequeno Jitter</i>	<i>Sensível ao Contexto</i>
Teleconsulta	Alto	Sim	Sim	Sim
Telediagnóstico	Alto	Sim	Não	Sim
Telemonitoramento	Baixo	Sim	Não	Sim
Teleeducação -VoD	Alto	Não	Não	Não
Acesso ao EHR	Baixo/Alto	Não	Não	
Tele cirurgia assistida	Alto	Sim	Sim	Sim

Fonte: Cavaro-Menard, Zhang-Ge e Callet (2013); Vouyioukas, Maglogiannis e Komnakos (2007)

O estudo da sensibilidade ao contexto (CAVARO-MENARD, ZHANG-GE, CALLET, 2013) foi apresentado em após discussão com especialistas médicos. Neste caso está sendo considerado que os requisitos de QoS de uma determinada aplicação de *eHealth* podem variar de acordo com o contexto, conforme exemplos apresentados a seguir.



Em uma situação de emergência, um diagnóstico de especialistas remotos pode exigir uma transmissão quase direta de dados médicos em tempo real, enquanto que em uma situação diferente e não emergencial, os dados médicos do paciente são transferidos (com tolerância para atraso) para um local remoto para serem analisados por especialistas (SKORIN-KAPOV; MATIJASEVIC, 2010).

[...] Um serviço com requisitos rigorosos de QoS e envolvendo transmissão de dados críticos para pacientes é a cirurgia. Portanto, determinar o contexto do serviço em termos de emergência ou de atendimento ao paciente versus a não-emergência e a entrega de serviços não críticos é crucial para fornecer informações para o mecanismo de agenciamento de fluxo de dados (SKORIN-KAPOV; MATIJASEVIC, 2010).

Um serviço que envolve conferências de áudio/vídeo entre um paciente e um médico para um exame de rotina pode ser considerado "menos crítico" com relação às garantias de QoS (isto é, pode tolerar degradação de atrasos de 150-400ms) em comparação com um serviço de conferência de áudio/vídeo empregado em uma situação de acidente de emergência onde a comunicação visual com um médico especialista remoto é imperativa (ou seja, uma interatividade "dura" em tempo real com atraso unidirecional 0-150 ms), (SKORIN-KAPOV; MATIJASEVIC, 2010).

Considerando os diferentes requisitos de QoS das aplicações de *eHealth* é necessário estabelecer parâmetros de referência, contudo, conforme apresentado nas Tabelas 7, 8 e 9 os requisitos de QoS para aplicações de *eHealth* são valorados por literais, ao invés de valores numéricos, dificultando o uso dos mesmos como valores de referência em contextos similares.

Na pesquisa apresentada em Ullah, Fiedler e Wac (2012), os autores constataram que não há uma visão unificada sobre os requisitos qualitativos de QoS-QoE. Para eles, as propostas existentes não contribuem para a compreensão da avaliação da QoE sob a percepção do

usuário, nem facilitam derivações de limites quantitativos da QoS para assegurar a percepção do usuário.

Para Skorin-Kapov e Matijasevic (2010), como os serviços de *eHealth* são providos por aplicações multimídias, os requisitos de QoS podem ser valorados seguindo padronizações regulamentadas (ITU-T G.1010, 2001; ITUT-T Y.1541, 2002; 3GPP, 2008, 2009). Com base em trabalhos existentes na literatura, a pesquisa sobre os requisitos de QoS para os serviços de *eHealth* em redes de próxima geração é apresentada e resumizada na Tabela 10, Skorin-Kapov e Matijasevic (2010).

Tabela 10 - Requisitos de QoS em serviços de e-Health.

Tipo de Serviço de eHealth	Exemplo de Aplicação de eHealth	Tipo de Mídia mais usada	Requisitos de QoS	
			Delay	Perda
Teleconsulta baseada em voz conversacional em tempo real.	Áudio conferência entre paciente e médico e, médico e médico.	Áudio.	<150ms E2E sentido único.	<1% Preferido <3%, limite.
Teleconsulta baseada em vídeo conversacional em tempo real.	Vídeo conferência entre paciente e médico e, médico e médico.	Vídeo.	<250ms E2E, sentido único, limite 400ms.	1% limite.
Serviços de robótica em tempo real.	Telecirurgia Teleultrasonografia.	Controle de dados robótica, áudio, vídeo.	<300ms de RTT.	Zero (pode tolerar o mínimo de 0,5%).
Telemonitoramento em tempo real.	Transmissão de sinais vitais de pacientes e streaming <sup>1</sup> de vídeo em situações de emergência.	Dados biomédicos coletado dos sensores.	Depende das aplicações e sensores, <300ms E2E, sentido único para ECG em tempo real (certas aplicações podem tolerar <1s E2E para ECG).	Zero.
Telemonitoramento, em tempo não real.	Transmissão de sinais vitais de pacientes para atendimento domiciliar pós-hospitalar.	Dados biomédicos coletado dos sensores, dados do contexto (e.g., coletados por sensores de ambiente).	Não disponível.	Zero.
Telediagnóstico em tempo real.	Transferência de imagens médicas	Imagens, texto e dados.	Não disponível (depende do)	Zero.

<sup>3</sup> Streaming é a forma de transmissão contínua de áudio e vídeo através de uma rede de comunicação de dados, sem a necessidade de efetuar downloads do que está se vendo e/ou ouvindo, pois neste método a máquina recebe as informações ao mesmo tempo em que as repassa ao usuário.

Tipo de Serviço de eHealth	Exemplo de Aplicação de eHealth	Tipo de Mídia mais usada	Requisitos de QoS	
			Delay	Perda
	para localização remota em situações de emergência.		tamanho da imagem. Imagens menores devem ser transferidas em poucos segundos).	
Telediagnóstico em tempo não real.	Diagnóstico remoto não emergencial: Transferências de imagens médicas para um local remoto, onde especialistas analisam e retornam um relatório diagnóstico.	Imagens, texto e dados.	Não disponível.	Zero.
Acesso aos dados do EHR em tempo real.	Pessoal médico de emergência em caso de acidente/desastre acessando o site do EHR de um paciente.	Dados, texto, gráficos, imagens.	Não disponível.	Zero.
Acesso /armazenamento dos dados ao EHR em tempo não real.	Aplicação Web de usuários finais (paciente, médico, pessoal da saúde) para o acesso ao EHR durante o exame (check-up) do paciente.	Dados, texto, gráficos, imagens.	Não disponível.	Zero.
Mensagens em Tempo Real	Alarmes enviados a cuidadores indicando emergência de pacientes	Texto, imagens pequenas, dados.	Não disponível	Zero
Mensagem em tempo não real	Alertas automatizados para pacientes (e.g., lembrete do check-up, lembrete para tomar a medicação).	Texto, imagens pequenas, dados.	~ 10s.	Zero
Pesquisa e educação conversacional	Ferramentas colaborativas de pesquisa/educação envolvendo áudio e vídeo conversacional; Grupos de apoio ao paciente virtual envolvendo áudio e vídeo.	Áudio, vídeo.	<150ms E2E sentido único para áudio. <250ms E2E para vídeo (limites superiores reposrtados como 400ms).	<3% áudio. <1% vídeo.
Pesquisa e educação interativa.	Simulações cirúrgicas interativas; controle remoto de instrumentos.	Dados, imagens.	<300ms RTT.	1%.
Streaming de pesquisa e	Ferramentas de educação	Áudio, vídeo, dados.	<10s de inicialização para áudio e vídeo.	<1% vídeo, <2% áudio.

<i>Tipo de Serviço de eHealth</i>	<i>Exemplo de Aplicação de eHealth</i>	<i>Tipo de Mídia mais usada</i>	<i>Requisitos de QoS</i>	
			<i>Delay</i>	<i>Perda</i>
educação.	envolvendo streaming de mídia (áudio e vídeo).			
Intercâmbio interativo de dados de informações sobre saúde.	Portais de saúde: web sites que oferecem dados relacionados à saúde.	Todas.	~2s para navegação na página Web.	Zero.
Recuperação não interativa de dados e informações de saúde.	Distribuição de livros didáticos de diagnóstico por imagem.	Todas.	Não disponível.	Zero.
Transações administrativas e financeiras.	Referências <sup>4</sup> de pacientes; agendamento de consultas; aplicativos de cobrança e cobrança.	Texto.	Não disponível.	Zero.

Fonte: Skorin-Kapov e Matijasevic (2010).

A Tabela 10 apresenta o tipo de serviço de *eHealth*, com um exemplo de aplicação do mesmo, o tipo de mídia mais utilizada e os requisitos de QoS (delay e perda). Na Tabela 10 os serviços são agrupados em conjunto com os requisitos de entrega (tempo real ou não real) e tipo de transmissão (comunicação conversacional bidirecional, transmissão unidirecional, solicitação interativa - resposta e recuperação de dados de *background*). Para certos serviços, os requisitos de atraso são indicados como "não disponíveis", já que nenhum requisito específico foi encontrado. Por exemplo, no caso de transferência de imagem, o atraso dependerá do tamanho da imagem e da largura de banda disponível. Segundo os autores, a transferência dos dados dos serviços de emergência deve ser concluída dentro de alguns segundos.

Os trabalhos apresentados nesta seção permitiram evidenciar que as restrições de QoS dos serviços de *eHealth* podem ser estimadas seguindo recomendações padronizadas por entidades, tais como, ITU-T, 3GPP. Contudo, os valores dos parâmetros de QoS podem variar, haja visto que o contexto deve ser avaliado para aferir valores com maior precisão. Apesar dessa conclusão, foi observado que os parâmetros de QoS são considerados como único aspecto representativo da experiência do usuário, contrariando o conceito de QoE, descrito a seguir.

<sup>4</sup> Referência, um ato de encaminhamento para consulta médica, revisão ou ação adicional.

### 3.3 QUALIDADE DE EXPERIÊNCIA

A QoE (*Quality of Experience*) refere-se a avaliação das percepções e expectativas dos usuários no uso de serviço. Essa avaliação pode ser influenciada por vários aspectos (e.g. funcionais, técnicos, humanos) que podem interferir na UX (*User Experience*) no consumo do serviço. Portanto, esta seção tem por propósito evidenciar as várias dimensões de influência da QoE.

#### 3.3.1 Relações entre Qualidade do Serviço e Qualidade de Experiência

Para oferecer um serviço com qualidade, durante muitas décadas, a grande preocupação estava centrada nas tecnologias de acesso, redes de computadores e telecomunicações. Sem, no entanto haver preocupação com a qualidade percebida pelo usuário. Conforme já abordado na contextualização (seção 1.1), inicialmente, o conceito de QoS foi focado na rede de comunicação de dados, posteriormente evoluiu para uma noção mais ampla contemplando as múltiplas camadas da interação usuário sistema (Figura 1).

Apesar da preocupação com a qualidade sob o ponto de vista do usuário, os recursos computacionais disponíveis no interior das instalações dos prestadores de serviços, geralmente são gerenciados com parâmetros de QoS convencionais, com Serviços Diferenciados (DiffServ), Blake et al. (1998), Serviços Integrados (IntServ), Braden, Clark e Shenker (1994) ou *Multi-Label Switching Protocol* (MPLS), Rosen, Viswanathan e Callon (2001).

Em muitos casos, o mesmo serviço é provido a diferentes usuários usando a mesma qualidade ou CoS predefinidas. Ocorre que ao adotar parâmetros convencionais, diferentes usuários podem ter diferentes percepções da qualidade do mesmo serviço, portanto, usando a percepção do usuário pode ser traduzido em uma prestação de serviços com padrão diferente. Em outras palavras, poderia ser relevante investigar a prestação de QoS com base em uma abordagem mais complexa, onde a experiência do usuário também é levado em conta.

É fato que a provisão da QoS é crítica e necessária, pois se em um sistema de comunicação não houver a gestão da QoS, a imagem de um vídeo transmitido pode aparecer distorcida (Figura 12), com atrasos para montar os frames que a compõem, o som pode vir com atraso e/ou com eco afetando assim, a UX do usuário. No entanto, ao considerar a

QoS como único determinante da QoE negligenciam-se aspectos relacionados com os anseios subjetivos e percepções cognitivas do usuário na interação com o ambiente.

Figura 12 - Captura de imagem distorcida



Fonte: ITU-T G.1080 (2008)

Em muitos casos a qualidade percebida (QoP – *Quality of Percetion*) pelo usuário tem sido relacionada, principalmente, com a QoS (FIEDLER; HOSSFELD; TRAN-GIA, 2010; SHAIKH; FIEDLER; COLLANGE, 2010). Em suas pesquisas, os autores (KHIRMAN; HENRIKSEN, 2002; KIM; CHOI, 2010; LI-YUAN; ZHOU; SONG, 2006) evidenciaram que a relação entre QoE e QoS é não linear.

Fiedler, Hossfeld e Tran-Gia (2010) propuseram uma relação exponencial entre a QoE e a QoS através da hipótese que existe uma interdependência exponencial entre a QoE e a QoS. Já Reichl et al. (2010), Reichl, Tuffin e Schatz (2011) identificaram uma relação logarítmica entre QoS e QoE para estimar a QoP do usuário na distribuição de recursos em redes de comunicação. Indo um pouco além, os resultados obtidos são analisados no contexto mais vasto da lei Weber-Fechner (WEBER, 1834) um princípio fundamental na psicofísica que descreve a relação geral entre a magnitude de um estímulo físico e sua intensidade percebida dentro do sistema sensorial humano.

Conforme visto, em soluções convencionais, tradicionalmente, os

parâmetros de QoS são considerados como ponto de referência para mensurar a qualidade do serviço e a satisfação do cliente. A grande maioria das pesquisas centra-se na identificação das relações entre os parâmetros de QoS e QoE. No entanto, a QoS é mais orientado para redes e aplicações, portanto, os parâmetros de QoS são incapazes de satisfazer as necessidades estéticas e hedônica dos agentes humanos. Uma boa estratégia para mudar essa concepção, consiste num estudo abrangente do conhecimento de todas as dimensões e fatores de influência da QoE.

### **3.3.2 Dimensões de Influência na Qualidade de Experiência**

Os fatores de influência de QoE têm sido definidos como "qualquer característica de um usuário, sistema, serviço, aplicativo ou contexto cujo estado real ou configuração pode ter influência sobre a Qualidade da Experiência para o usuário", Callet, Moller e Perkis (2012). Uma classificação que surgiu a partir da comunidade Qualinet, sob a forma de um Livro Branco (CALLET; MOLLER; PERKIS, 2013) apresenta os fatores de influência na QoE em três grandes categorias, ligadas a fatores humanos, sistema e contexto. Levando em consideração todos esses aspectos, observa-se que existe um expressivo número de fatores que podem influenciar a percepção do usuário no uso de um serviço, dentro de cada categoria. Uma multiplicidade de diferentes fatores pode ser visualizada na Figura 13.

A Figura 13 apresenta exemplos de fatores de influência na UX relacionados a diferentes aspectos, incluindo aspectos relacionados com a rede, o servidor que provê o serviço, conteúdo, tecnologia, custo, dispositivo, humanos, contexto pessoal, entre outros.

Para a identificação dos fatores de forma sistemática é necessário agrupá-los em categorias. Segundo os autores (BARAKOVIĆ; SKORIN-KAPOV, 2013) várias abordagens existentes têm endereçado e proposta a classificação dos fatores de influência na QoE dentro de múltiplas dimensões. Assim, considera-se que a QoE deve ser modelada como um constructo multidimensional, amparada por uma abordagem interdisciplinar. Nesse viés, as dimensões da QoE serão mapeadas, de acordo com algumas definições e trabalhos publicados na literatura.

Figura 13 - Diversidade de fatores de impacto na QoE



Fonte: Baraković e Skorin-Kapov (2013)

### 3.3.2.1 Dimensões de influência na Qualidade de Experiência a partir das definições existentes

Na literatura existem várias definições da QoE, no entanto, foram selecionadas aquelas que abordam a QoE numa visão multidimensional e que consideram principalmente aspectos humanos em suas definições, mapeadas no Quadro 3.

Quadro 3 - Conceitos de QoE em definições existentes

Conceito de QoE	Referência
“O grau de satisfação do usuário de um serviço, influenciado pelo conteúdo, rede, dispositivo, aplicação, expectativas e metas/objetivos dos usuários e contexto de utilização”.	ITU-T(2008)
“As características das sensações, percepções e opiniões das pessoas em relação à interação com os ambientes. Essas características podem ser agradáveis e prazerosas ou desagradáveis e frustrantes”.	Fiedler, Kilkki e Reichl (2009, p. 2)
“Uma medida de desempenho do usuário com base em ambas as medidas psicológicas objetivos e subjetivos no uso de um serviço de TIC ou produto”.	ETSI (2010)



Conceito de QoE	Referência
“Um constructo multidimensional de percepções e comportamentos de um utilizador, o que representa respostas emocionais, cognitivas e comportamentais, tanto subjetivas e objetivas ao usar um sistema”.	Wu et al. (2009)

A partir das definições apresentadas acima foram extraídas quatro dimensões da percepção humana sobre a qualidade, sendo elas: usuário, conteúdo, contexto e tecnologia (rede, dispositivo, aplicação, sistema).

### 3.3.2.2 Dimensões de influência na Qualidade de Experiência a partir das pesquisas existentes

Visando identificar se as dimensões contempladas nas definições da QoE têm sido aplicadas pelos trabalhos existentes foi feita uma pesquisa para levantar as propostas sobre QoE. Os dados foram coletados em três bases de dados, Scopus, Web of Science, IEEE Explorer. Várias combinações de filtros foram realizadas e os resultados foram apresentados nas Tabela 1 (seção 1.4). A partir da coleta de dados foram selecionados treze artigos, principalmente, por considerarem a QoE não apenas no escopo das TICs, mas por agregarem outras áreas do conhecimento.

O artigo (LI-YUAN; ZHOU; SONG, 2006) destaca que a satisfação do usuário está se tornando um dos temas mais importantes para os fornecedores de serviços. Os autores descrevem que os métodos existentes utilizados para avaliar a QoE em sua maioria dependem da pesquisa e pontuação/avaliação dos usuários, que são subjetivos, necessitam de muito tempo de processamento e tem alto custo. O ambiente de estudo é no contexto da computação pervasiva. No artigo é estudado o método de avaliação da QoE em ambiente de computação ubíqua, o modelo de avaliação de parâmetros de QoE é reforçado e um algoritmo é usado para reduzir os atributos de contexto e determinar o peso de cada atributo. O algoritmo foi validado em serviço de vídeo streaming, e a arquitetura do sistema de avaliação de QoE é descrita e validada.

Em Latré et al. (2008) é proposta uma KB para otimizar a QoE de forma autônoma em redes de acesso multimídia. A KB contém informações da rede, serviços e ações de reparação da QoE. O artigo mostra todos os componentes necessários para a gestão da QoE usando a ideia original do MAPE (*Monitor-Analyze-Plan-Execute*) proposta pela

IBM. A arquitetura é composta por três camadas (plano de monitoração, plano de conhecimento e plano de ação). O artigo apresenta uma visão geral de todos os requisitos presentes na KB, usando OWL (*Web Ontology Language*) e SWRL (*Semantic Web Rule Language*), que podem ser usados em uma arquitetura autônoma de gestão de QoE para redes de acesso. Além de armazenar todas as informações, a KB autonomamente é capaz de deduzir novas informações. Essas novas informações podem ser usadas para disparar alarmes para as camadas superiores da arquitetura ou propor autonomamente ações de reparação da QoE para os serviços que estejam com baixo desempenho.

Em Moor et al. (2008) a QoE é definida como um conceito multidimensional, sendo dado enfoque na QoE não apenas na perspectiva tecnológica, mas também sob a visão das necessidades dos usuários. Os autores fornecem um modelo conceitual para a QoE e discutem o *gap* que ainda existe entre QoE e QoS. O objetivo do trabalho consistiu em apresentar uma nova metodologia para correlacionar a experiência do usuário com os parâmetros de QoS. Vários fatores foram analisados no estudo, incluindo qualidade da eficácia, qualidade da eficiência, usabilidade, expectativas, contexto do usuário. A metodologia foi testado no contexto de um estudo exploratório interdisciplinar para realizar medições da QoE. A abordagem incluiu os usuários no projeto de desenvolvimento de celulares. Os usuários testaram e avaliaram a tecnologia e softwares suportados pelo dispositivo, antes do mesmo ser lançado no mercado. Os usuários também deram opiniões dos aspectos considerados relevantes em um dispositivo celular.

O artigo (MÖLLER et al., 2009) descreve que a QoS e a QoE são importantes não só para a transmissão de dados multimídia, mas também para os serviços que envolvem interação homem máquina multimodal. A fim de orientar a análise e avaliação de tais serviços, os autores desenvolveram uma taxonomia de aspectos de QoS e QoE, com os aspectos mais relevantes que resultam de interações homem-máquina multimodais. A mesma taxonomia, mais tarde foi apresentada por Wechsung et al. (2012).

Em Wu et al. (2009) é evidenciado a necessidade de uma abordagem multidisciplinar para criar um framework com base nos resultados teóricos dos vários campos, incluindo a psicologia, a ciência da cognição, a sociologia e tecnologia da informação. Os autores apresentaram uma metodologia de mapeamento para quantificar as correlações entre QoS e QoE e descreveram os estudos controlados e não controlados para analisar o framework proposto. Os resultados, de

estudos aprofundados, modelam as dimensões da QoE, o mapeamento do relacionamento entre QoS e QoE e capturam a qualidade humana centrada no contexto de ambientes multimídia interativos distribuídos.

No trabalho dos autores, Baraković, Baraković e Bajrić (2010), é descrita a importância de avaliar a QoE centrada nas necessidades do usuário e não apenas centrada em tecnologia. Usando uma abordagem conceitual, os fatores de influência na QoE são categorizados em cinco dimensões: (1) desempenho da tecnologia em quatro níveis: aplicação/serviço, servidor, rede e dispositivo; (2) usabilidade, referindo-se ao comportamento dos usuários ao utilizar a tecnologia; (3) avaliação subjetiva; (4) as expectativas; e o (5) contexto. Os autores apresentam técnicas e abordagens disponíveis para medir a QoE em serviços providos em redes de próxima geração, procurando levantar questões relacionadas com todos estes aspectos, a fim de alcançar conclusões sobre as perspectivas de desenvolvimento de soluções para a QoE.

O artigo (LAGHARI et al., 2010) descreve que para garantir a QoE, o sistema de prestação de serviços deve mudar para a abordagem centrada na experiência do cliente. O artigo apresenta um modelo ontológico para a experiência do cliente, para uso em ambientes de tempo de execução, com base em sistemas de gestão de política. A investigação em curso apresenta uma abordagem de entrega/prestação de serviços orientada para a experiência do cliente. O modelo de QoE opera dados quantitativos e qualitativos, é baseada em modelos da ciência cognitiva e psicologia para atender as necessidades do usuário. O modelo adquirido é transformado para um modelo ontológico que permite a entrega de serviços orientados à experiência do cliente.

O artigo (LAGHARI et al., 2011) descreve que a entrega do serviço e satisfação dos clientes são itens fortemente relacionados com uma adequada plataforma de gestão comercial. Aspectos técnicos voltadas para esta questão dizem respeito a parâmetros de QoS que podem ser tratadas pela plataforma, pelo menos parcialmente. Questões psicológicas subjetivas e aspectos cognitivos humanos são tipicamente aspectos não reconhecidos e determinantes no resultado da QoE. Esses fatores, finalmente, têm que ser considerados como insumos fundamentais para a operação de negócios bem sucedidos entre um cliente e uma companhia. No artigo é usada uma abordagem multidisciplinar para propor um modelo de interação de QoE com base nos resultados teóricos de diversas áreas, incluindo psicologia, ciência da cognição, sociologia, o ecossistema de serviços e tecnologia da informação. No trabalho é descrito um avaliador da QoE para avaliar a

prestação de serviços prestado pelo sistema a um usuário, em um ambiente integrado e distribuído.

Stankiewicz e Jajszczyk (2011) classificaram os fatores tecnológicos que impactam a QoE em três grupos: QoS, Grade de Serviço (GoS), e Qualidade de Resiliência (QoR). Além disso, eles levam em consideração o número de fatores adicionais (principalmente os não relacionados com a tecnologia), tais como emoções, perfil do usuário, política de preços, características específicas de aplicação, terminais, codecs, tipo de conteúdo, e aspectos ambientais, psicológicos e sociológicos, contudo esses fatores não estão agrupados.

Para os autores Wechsung et al. (2012) a QoS e a QoE devem ser considerados no projeto, construção e manutenção de serviços que envolvem interação homem máquina multimodal. A fim de orientar a avaliação de tais serviços, primeiro os autores desenvolveram uma taxonomia de QoS e QoE, com os aspectos mais relevantes, que resultam em interações homem máquina multimodais. A taxonomia consiste de três camadas: (1) os fatores que influenciam a QoS e QoE relacionados com o usuário, sistema e contexto de uso, (2) aspectos de desempenho de interação de QoS descrevendo o comportamento e desempenho do usuário e do sistema, e (3) aspectos de QoE relacionados com a percepção de qualidade e processos de julgamento do usuário. Para cada uma destas camadas são fornecidas métricas capazes de captar os aspectos da QoS e QoE de forma quantitativa, seja através de questionários ou medidas de desempenho.

O artigo (SKORIN-KAPOV; VARELA, 2012) aponta a necessidade de uma modelagem com ampla gama de fatores que influenciam e afetam a QoE do usuário final. Os autores estão conscientes que a QoE vai além da QoS e tornou-se um importante problema para provedores de serviços e redes, em especial para os provedores de serviços emergentes, como é o caso dos serviços multimídias interativos e distribuídos. No artigo, os autores apresentam o ARCU (*Application-Resource-Context-User*), um modelo genérico que categoriza os fatores de influência da QoE em quatro espaços multidimensionais: aplicação, recurso, contexto e usuário. No modelo é assumido que a QoE é composta por  $n$  dimensões, com fatores de pesos correspondentes. Os autores descrevem que diferentes métricas podem ter diferentes importâncias relativas (dependendo da aplicação propósito/tipo de aplicação, tarefa do usuário, preferências do usuário...) e, portanto, terem diferentes fatores de peso atribuídos. Por exemplo, uma avaliação da métrica de QoE "confiabilidade do conteúdo" é muito importante para uma aplicação *eHealth* e contribui para a obtenção geral

da QoE, enquanto a avaliação da mesma métrica, numa aplicação de jogos, contribui em menor grau para a obtenção geral da QoE. O modelo proposto não foi validado, ao invés, são apresentados exemplos de três aplicações, (video Stream, VoIP, jogos *multiplayer*), as quais o modelo pode ser aplicado, com os respectivos fatores dentro de cada dimensão.

Usando uma abordagem ontológica, apoiado principalmente pela Engenharia do Conhecimento, os autores Fallon e O’Sullivan (2012) afirmam que os dados necessários para a otimização automática de serviços para o usuário normalmente existem em sistemas atuais, mas os mesmos não são modelados ou relacionados de uma forma que facilite a automação. Então, uma ontologia e uma KB são propostas para formalizar o conhecimento da experiência do usuário e do contexto, no uso de um serviço de IPTV. Os dados da avaliação do serviço são coletados diretamente dos terminais dos usuários e mapeados para a KB. A base de dados é populada e a partir dela, segundo os autores, é possível aplicar técnicas semânticas para analisar e sugerir otimizações na qualidade do serviço.

Em Huong-Truong (2013) é proposta uma nova arquitetura, projetada para a Internet do futuro, para a avaliação da QoE num subsistema multimídia IP. Foi implementado um motor para avaliar a experiência do usuário no uso do serviço de IPTV. Para avaliar a QoE, os autores utilizaram regressão linear com o mapeamento das variáveis independentes (parâmetros de QoS da rede - descarte de pacotes, atraso, *jitter*; parâmetros de QoS da aplicação - taxa de bits) para obter a variável dependente (parâmetro de QoE - MOS). Na provisão dos serviços são consideradas as dimensões, conteúdo e QoS. A Arquitetura proposta foi avaliada e de acordo com os autores, o motor se comportou conforme o esperado.

A Tabela 11 apresenta a sumarização das pesquisas, destacando a abordagem, a aplicação da proposta e as bases conceituais que foram citadas e/ou aplicadas nos artigos.

Tabela 11 - Comparativo dos resultados da busca sistemática

<i>Artigo</i>	<i>Abordagem</i>	<i>Aplicação</i>	<i>Bases Conceituais</i>	<i>Dimensões da QoE</i>
Li-yuan, Zhou e Song (2006)	Disciplinar	Vídeo streaming em computação pervasiva	Computação sensível ao contexto, Fuzzy	QoS, contexto, perfil usuário, fatores humanos subjetivos
Latré et al. (2008)	Interdisciplinar	Rede de acesso multimídia	Engenharia do Conhecimento, TIC	QoS da rede, QoS da aplicação

<i>Artigo</i>	<i>Abordagem</i>	<i>Aplicação</i>	<i>Bases Conceituais</i>	<i>Dimensões da QoE</i>
De Moor et al. (2008)	Interdisciplinar	Qualidade no desenvolvimento de celulares	Interface homem máquina, Interface Design, TIC, Engenharia, Sociologia	Qualidade da aplicação, contexto usuário, fatores humanos objetivos e subjetivos
Möller, et al. (2009)	Interdisciplinar	Interação homem máquina multimodal	Interface homem máquina, Design de Interface	QoS, usuário, contexto, sistema (fatores do agente e funcionais), fatores humanos objetivos
Wu et al. (2009)	Interdisciplinar	Ambiente multimídia interativo distribuído	Psicologia, Ciência da Cognição, Sociologia, TIC	Qualidade da aplicação, fatores humanos objetivos e subjetivos
Baraković, Baraković e Bajrić (2010)	Interdisciplinar	Serviços de rede de próxima geração	TIC, Interface homem máquina	Tecnologia (aplicação/serviço, servidor, rede e dispositivo), usabilidade, avaliação subjetiva; expectativas, contexto.
Laghari et al. (2010)	Interdisciplinar	Entrega de serviços em ambientes distribuídos	Psicologia, Ciência da Cognição, Sociologia	Usuário, serviço, negócio, fatores humanos subjetivos
Laghari et al. (2011)	Interdisciplinar	Provisão e entrega de serviços em ambientes distribuídos	Psicologia, Ciência da Cognição, Sociologia, Interface homem máquina, Interface Design, TIC	QoS da aplicação e da rede, serviço, negócio, contexto, fatores humanos objetivos e subjetivos
Stankiewicz e Jajszczyk (2011)	Disciplinar	Serviços em redes convergentes	TIC	Tecnologia (QoS, Grade de Serviço, Qualidade de Resiliência), fatores não categorizados (emoções, perfil do usuário, preço, aplicação, terminais, codecs, conteúdo, aspectos ambientais, psicológicos e sociológicos).
Wechsung et al. (2012)	Interdisciplinar	Interação homem máquina multimodal	Interface homem máquina, Design de Interface	QoS, usuário, contexto, sistema (fatores do agente e funcionais), fatores humanos objetivos
Skorin-Kapov e Varela (2012)	Interdisciplinar	Video Stream, VOIP, Jogos Multiplayer	TIC, Matemática	QoS da rede e contexto.
Fallon e O' Sullivan (2012)	Interdisciplinar	IPTV em HAN (Home Area Network)	Engenharia do Conhecimento, TIC	QoS da rede e contexto

<i>Artigo</i>	<i>Abordagem</i>	<i>Aplicação</i>	<i>Bases Conceituais</i>	<i>Dimensões da QoE</i>
Huong-Truong et al.(2013)	Disciplinar	IPTV num subsistema multimídia IP	TIC	Usuário, conteúdo e QoS da aplicação e QoS da rede.

Analisando as pesquisas sumarizadas na Tabela 11 e fazendo um contraste das dimensões de influência evidenciadas na seção anterior (usuário, conteúdo, contexto e tecnologia), observa-se uma consonância entre as definições de QoE e as propostas de modelagem, conforme segue:

- A dimensão “usuário” contempla {perfil do usuário, fatores subjetivos, fatores objetivos, avaliação subjetiva; expectativas, psicológicos, emoções, preço}.
- A dimensão “contexto” contempla {usuário, ambientais, sociais}.
- A dimensão “tecnologia/rede”, contempla {QoS da rede, resiliência, serviço}.
- A dimensão “tecnologia/aplicação” contempla {QoS da aplicação, codec}.
- A dimensão “tecnologia/sistema” contempla {fatores do agente, fatores funcionais, usabilidade}.
- A dimensão “tecnologia/dispositivo” contempla {terminais, dispositivos do usuário}.

Fazendo uma análise mais detalhada nas propostas, observou-se que a dimensão “contexto” tem sido pouco explorada e ainda, constatou-se que a dimensão “conteúdo” não foi considerada na modelagem da QoE. O único trabalho que esta dimensão aparece (STANKIEWICZ; JAJSZCZYK, 2011), a mesma é apenas citada, mas nenhum estudo foi realizado para explorar esta dimensão.

Outra constatação a partir da análise foi que, diferente das dimensões evidenciadas anteriormente, a dimensão “negócios” aparece nos trabalhos dos mesmos autores Laghari et al. (2010) e Laghari et al. (2011). Esse fato é justificado pela atribuição de diferentes papéis para as entidades, chamadas de entidades de negócios, entidades humanas, entidades tecnológicas, elencadas no modelo das interações da QoE. Para os autores, os fatores de negócios estão associados com custo, promoções e anúncios, etc. Contudo, considera-se que fatores associados a custos e promoções podem ser derivados a partir da dimensão “contexto”.

Com base nessa análise, conclui-se que os fatores de influência

na QoE são inúmeros e podem evoluir em função da demanda, da experiência do usuário, do contexto e das evoluções tecnológicas, portanto a modelagem deve ser flexível e prever futuras implementações, assim, qualquer taxonomia proposta irá representar uma versão simplificada da realidade. Consideramos ainda, que as dimensões e fatores de influencia na QoE irão depender do tipo de serviço, incluindo diferentes atribuições de acordo com os papéis assumidos pelas entidades.

### 3.3.3 Métodos para Avaliação da Qualidade de Experiência

A QoE é uma métrica de desempenho usada para avaliar o grau de julgamento do usuário no uso de um serviço, sistema, aplicação, ambiente. Portanto, a construção de um modelo de QoE deve prever métodos de avaliação. É importante notar que diferentes modelos de QoE e métodos de avaliação são aplicáveis a diferentes tipos de serviços (por exemplo, de conversação, serviços de voz, serviços de *streaming* de áudio-visual, serviços interativos, serviços de colaboração), contudo, esta pesquisa não foca na modelagem de um tipo particular de serviço, e sim numa abordagem geral da modelagem da QoE.

#### 3.3.3.1 Métodos subjetivos

Os métodos subjetivos consistem em avaliar a QoE sob a perspectiva do usuário, ou seja, são os próprios usuários que avaliam o serviço durante o uso ou após o uso, e fornecem dados para um sistema ou agente controlador. As avaliações de qualidade subjetivos são baseadas em experimentos psicoacústicos/visuais que representam o caminho fundamental e mais confiável para avaliar a QoE dos usuários, que são complexos e dispendiosos.

A fim de estabelecer algum padrão ou metodologia para aplicação dos testes subjetivos, organismos de normalização, como o ITU-T, em sua recomendação (ITU-T P.800, 1996) apresenta uma metodologia para a realização de testes subjetivos. Esta recomendação inclui um dos métodos mais usuais para a avaliação subjetiva, MOS (*Mean Opinion Score*), Tabela 12.

Tabela 12 - Diversidade de fatores de impacto na QoE

<i>MOS</i>	<i>Qualidade</i>	<i>Prejuízo</i>
5	Excelente	Imperceptível
4	Boa	Perceptível, mas não irritante
3	Razoável	Um pouco irritante



<i>MOS</i>	<i>Qualidade</i>	<i>Prejuízo</i>
2	Ruim	Irritante
1	Péssima	Muito Irritante

O MOS (ITU-T P.800.1, 2003) é amplamente utilizado em testes de avaliação da qualidade de áudio e vídeo, onde um conjunto de avaliadores avalia o material processado e atribui uma pontuação usando uma escala numérica. Na escala MOS (Tabela 12), a satisfação do usuário é mapeada em cinco classes: [5..4] muito satisfeito; [4..3] satisfeito; [3..2] insatisfeito; [2..1] muito insatisfeito; [1..0] nenhum usuário satisfeito. A média desses valores representa um índice de referência para a avaliação do material processado.

Os testes subjetivos possuem extrema relevância para os usuários, pois são eles que aplicam sua experiência para opinar sobre a qualidade do material processado. No entanto, tem sido criticado por uma série de razões (HOßFELD; SCHATZ; EGGER, 2011; MITRA; ZASLAVSKY; AHLUND, 2014) dentre elas:

- Necessidade de um considerável espaço amostral para obter resultados credíveis;
- Demanda de tempo para selecionar os usuários, recuperar seus pareceres e processá-los;
- Limitados para grandes provedores de telecomunicações;
- A língua nativa dos usuários pode não ser a mesma e os resultados obtidos podem ser tendenciosos ou mesmo incompletos;
- A média das avaliações dos usuários é computada, mas, operações matemáticas, tais como média e desvio padrão não podem ser aplicados em avaliações subjetivas quando estas classificações são categóricas em natureza, utilizando literais (“excelente”, “justa”).

O MOS foi estendido a outros procedimentos de avaliação subjetivos recomendados pela ITU (2010), classificados por tipo de aplicação e meios de comunicação. Além dos métodos padronizados para a avaliação subjetiva da QoE, outros métodos adicionais, por vezes complementares, são utilizados para a avaliação da UX.

Com relação à coleta de dados e de execução dos experimentos da QoE, as avaliações podem ser conduzidas num laboratório experimental (MENKOVSKI et al., 2011) em laboratórios usando um ambiente real (DE MOOR et al., 2010), ou num ambiente do mundo real (SHAIKH; FIEDLER; COLLANGE, 2010; WAC et al., 2011). Alguns critérios de desempenho são modificados em um determinado intervalo

de uma forma controlada e, posteriormente, opiniões dos usuários sobre o desempenho do serviço são quantificadas.

Como uma emergente e prospectiva solução para a obtenção de um grande número de avaliações num ambiente do mundo real, a metodologia *crowdsourcing* foi estudada e utilizada (CHEN et al., 2010; HOBFELD et al., 2011).

### 3.3.3.2 Métodos Objetivos

Considerando os problemas, apontados anteriormente, no uso de testes subjetivos, existe um grande interesse em utilizar testes objetivos que sejam usados para avaliar a qualidade das aplicações de acordo com a percepção humana, evitando custos e tempo em avaliações empíricas. Para este propósito, organismo de padronização tem recomendado alguns testes objetivos para a avaliação do material processado. Alguns métodos objetivos da ITU-T tais como, e-Model (ITU-T G.107, 2014), PESQ (ITU-T P.862, 2001), uma vez desenvolvidos, podem ser usados para a previsão da QoE sem exigir testes subjetivos.

Para determinar a QoE, a maioria dos métodos objetivos realiza um mapeamento das suas pontuações para a escala MOS, tal como o e-Model, que calcula um número chamado de fator R (escala de 0 a 100), derivado de atrasos e fatores de deterioração causados pelos equipamentos. O fator R obtido é mapeado para o MOS (Tabela 13) usando uma equação não linear para determinar a QoE.

Tabela 13 - Representação do mapeamento entre Fator R e MOS

<i>Fator R</i>	<i>Satisfação do Usuário</i>	<i>MOS</i>
$90 \leq R \leq 100$	Ótima	4,3 – 4,5
$80 \leq R \leq 90$	Boa	4,0 – 4,3
$70 \leq R \leq 80$	Mediana	3,6 – 4,0
$60 \leq R \leq 70$	Pobre	3,1 – 3,6
$50 \leq R \leq 60$	Ruim	2,6 – 3,10
$00 \leq R \leq 50$	Péssima	1,0 – 2,6

De acordo com os autores, Baraković e Skorin-Kapov, (2013) cada modelo tem seu próprio domínio de aplicação e uma gama de condições do sistema ou serviço para o qual ele foi projetado. Considerando que não existe alguma avaliação universal de medição da qualidade usando a abordagem objetiva, as propostas podem ser categorizadas por vários critérios, a fim de determinar a sua área de aplicação (ITU-T G.1011, 2010): (1) escopo da aplicação; (2) características da qualidade que está sendo predita; (3) componentes de rede considerados e configurações; (4) parâmetros de entrada do

modelo; (5) método de medição; (6) nível de interatividade do serviço que possa ser avaliado; e (7) nível do conhecimento psicofísico ou dados empíricos que foram incorporados.

De acordo com o nível em que a informação de entrada é obtida, existem cinco tipos de modelos objetivos da avaliação da QoE (ITU-T G.1011, 2010) : (1) camada de mídia; (2) camada de rede; (3) camada de fluxo de bits; (4) híbrido; (5) modelos de planejamento.

Os modelos da camada de mídia (ITU-T P.862, 2001; ITU-T P.863, 2011; ITU-R BS.1387, 2001; ITU-T P.563, 2004) estimam a QoE do áudio/vídeo usando os sinais de mídia atuais como entrada. Em dependência da utilização da fonte do sinal, podem ser usadas três diferentes abordagens (KUIPERS et al., 2010; TAKAHASHI; HANDS; BARRIAC, 2008): (1) sem referência (NR - *no-reference*); (2) referência reduzida (RR - *reduced-reference*); e (3) modelo de referência completa (FR - *full-reference*).

Os modelos da camada de rede (ITU-T P.564, 2007) utilizam apenas informações do cabeçalho do pacote para estimar QoE, descritos como serviço de monitoramento da qualidade não intrusiva. Um modelo da camada de fluxo de bits é a combinação dos dois modelos anteriormente mencionados, uma vez que utilizam informações dos fluxos dos bits, bem como informações do cabeçalho dos pacotes.

Similarmente, os modelos híbridos (ITU-T G.107, 2011; ITU-T G.1070, 2007; ITU-T G.1030, 2005) são concebidos como uma combinação dos três modelos descritos anteriormente.

Finalmente, os modelos de planejamento (ITU-T G.1011, 2010) não adquirem informações de entrada de um serviço existente, mas estimam com base na informação do serviço disponível durante a fase de planejamento.

Além de padrões preconizados pelo ITU-T, o ETSI (2011) fornece um guia abrangente com definições genéricas e métodos para a maioria dos principais serviços de telecomunicações. Existem outros organismos de normalização que tratam da avaliação da QoE, incluindo VQEG, MPEG e JPEG.

Enquanto a maioria das propostas considera métodos objetivos, em relação a coleções de tecnologia orientada a dados, é importante salientar que as medidas objetivas também podem se referir a estimativas objetivas do comportamento do usuário (por exemplo, duração da tarefa, o número de cliques do mouse), que são comumente consideradas apenas como medidas subjetivas (BROOKS; HESTNES, 2010).

Em Tsolkas et al. (2017) são apresentadas e avaliados os modelos

para estimar o MOS dos serviços mais populares, incluindo VoIP, VoD, IPTV, transmissão de vídeo, navegação Web, serviços de download, Skype, conforme apresentado no Tabela 14.

Tabela 14 - Modelos de estimativa de QoE por tipo de serviço

<i>Tipo de Serviço</i>	<i>Modelo para estimar a QoE</i>	<i>MCPs e KPIs</i>
Transferência de Arquivos	Equação baseada em taxa de dados (THAKOLSRI et al., 2009).	Taxa de transmissão, Taxa de transmissão máxima e mínima esperadas.
Navegação Web	Tempo de resposta, (AMEIGEIRAS; RAMOS-MUNOZ, 2010).	Tempo de resposta.
Skype	Equação específica do Skype (WAMSER et al., 2013).	Taxa do Frame, qualidade da imagem, resolução.
VoIP	E-Model, (COLE; ROSENBLUTH, 2001; ITU-T G.107, 2014; ITU-T G.108, 1999).	Taxa de perda de pacotes, atraso, codec, taxa de codificação.
Vídeo Streaming	IPTV model (KHAN; SUN; IFEACHOR, 2009; KHAN et al., 2010). YouTube com modelo convencional (HOBFELD et al., 2013). YouTube com modelo de streaming adaptativo (HOBFELD et al., 2014; SEUFERT et al., 2015).	Taxa de dados, Taxa de frames. Número de eventos paralisantes, duração dos eventos paralisantes, duração do vídeo. Tempo na camada mais alta, amplitude, frequência da qualidade dos computadores.
Vídeo Online	E-model (ITU-T G.1070, 2012)	Taxa de perda de pacotes para áudio e vídeo, delay relativo entre pacotes de áudio e vídeo, taxa de dados, taxa de frames, tamanho do monitor.

Fonte: Tsoolkas et al. (2017)

Na Tabela 14, para cada serviço é referenciada uma equação para estimar o MOS, com os principais KPI (*Key Performance Indicators*) e MCP (*Major Configuration Parameters*) usados para determinar o desempenho da QoE. Como resultados dos experimentos realizados, Tsoolkas et al. (2017) constataram que os KPIs e os MCPs são altamente variáveis por tipo de serviço, e que, mesmo para o mesmo serviço, diferentes fatores contribuem com um peso diferente da QoE percebida. Segundo os autores essa descoberta permite um provisionamento de recursos mais significativo em diferentes aplicações, em comparação com esquemas agnósticos da QoE.

Segundo os autores (MITRA; ZASLAVSKY; AHLUND, 2014), os métodos objetivos são difíceis de desenvolver, modelar e implementar devido a grande quantidade de parâmetros. Além disso, qualquer modificação feita aos métodos atuais, por adição ou supressão de parâmetros, novos testes devem ser realizados para refinar o modelo atual ou para derivar novos modelos estatísticos de previsão de QoE (BROOKS; HESTNES, 2010).

### 3.3.3.3 Métodos orientados ao contexto

Mitra, Zaslavsky e Ahlund (2014) afirma que a maioria dos métodos de medição e previsão de QoE foram desenvolvidos em ambientes laboratoriais controlados, com um número limitado de parâmetros subjetivos e objetivos, tais como atraso, *jitter*, perda de pacotes, e largura de banda. Contudo, os autores Moor et al. (2010); Jumisko-Pyykkö e Hannuksela (2008); Ickin et al. (2012); Mitra, Ahlund e Zaslavsky (2011a, 2011b) defendem que as medições e previsões da QoE devem acontecer em ambiente da vida real dos usuários, pois a QoE dos usuários difere do ambiente laboratorial para o ambiente diário do usuários (JUMISKO-PYYKKÖ; HANNUKSELA, 2008).

Para Mitra, Zaslavsky e Ahlund (2014), uma infinidade de parâmetros de contexto e de QoS podem afetar a QoE, pois no ambiente da vida real, o contexto pode alterar dinamicamente de acordo com as atividades e movimentações do usuário. Fatores como horário do dia, podem ajudar a explicar a ascensão ao congestionamento da rede, levando à diminuição da QoE dos usuários. Além disso, o contexto social dos usuários muda durante todo o dia e a QoE pode ser afetada se houver pessoas nas proximidades.

Os métodos objetivos (CHEN; HUANG; HUANG, 2006; CHEN; TU; XIAO, 2009; FIEDLER; HOSSFELD; TRAN-GIA, 2010; RUBINO; TIRILLY; VARELA, 2006) não consideram o agrupamento de vários parâmetros de contexto, como a localização do usuário, hora do dia e a resolução da tela para a previsão QoE. Assim, de acordo com Mitra, Zaslavsky e Ahlund (2014), os métodos objetivos que não consideram o contexto podem gerar imprecisões nas predições da QoE.

Brooks e Hestnes (2010); Moor et al. (2010); Mitra, Ahlund e Zaslavsky (2011a, 2011c), evidenciaram a importância de dois requisitos dos métodos de avaliação da QoE orientada ao contexto: (i) considerar tanto métodos subjetivos quanto objetivos e (ii) medir a QoE em uma única escala, combinando vários parâmetros de QoE.

Na direção do primeiro requisito, Wac et al. (2011) propuseram dois métodos para avaliações da QoE em aplicações móveis. Um método (ESM – *Experience Sampling Method*) (HEKTNER, SCHMIDT; CSIKSZENTMIHALYI, 2006) recolhe as classificações de QoE dos usuários em seus telefones móveis, várias vezes por dia, enquanto o outro (DRM – *Day Reconstruction Method*) (DINDA et al., 2007) tem sido utilizado semanalmente para entrevistar os usuários, com o propósito de obter padrões de uso e experiências com as aplicações

móveis. Estes métodos têm servido para analisar possíveis relações e causalidades entre classificações de QoE, QoS e contexto.

Orientados para o segundo requisito, Mitra, Zaslavsky e Ahlund (2015) propuseram, desenvolveram e validaram uma abordagem para a modelagem, medição e previsão da QoE orientada a contexto. Os autores relatam a experiência do desafio da medição e previsão da QoE, com diferentes escalas de medições, podendo envolver diferentes unidades de medidas. A abordagem, sensível ao contexto, usa redes bayesianas e teoria da utilidade para medir e prever QoE dos usuários sob incerteza. A mesma foi validada através de extensa experimentação, estudos de usuários e simulações. Os resultados demonstram que a proposta é promissora.

#### 3.3.3.4 Outros métodos de avaliação da Qualidade de Experiência

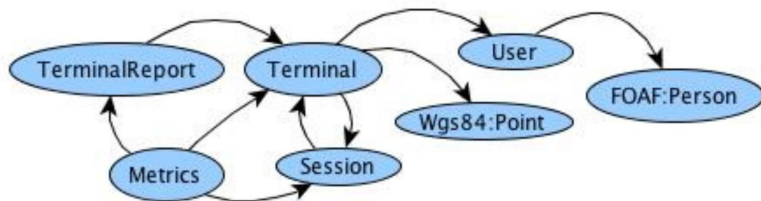
No trabalho descrito em Alreshoodi e Woods (2013) é apresentada uma revisão bibliográfica com diferentes métodos de avaliação da QoE, usados em modelos de QoE. Tais modelos incluem métodos de análises estatísticas, aprendizagem de máquina, *crowdsourcing* para testes de métodos subjetivos, recursos arbitrários do sistema, fatores ambientais e do equipamento, baseados em abordagem qualitativa e quantitativa.

#### 3.3.4 Representação do Conhecimento da Qualidade de Experiência

De todos os trabalhos referenciados na Tabela 11, apenas três demonstraram a representação do conhecimento da QoE (FALLON; O’SULLIVAN, 2012; LAGHARI et al., 2010; LATRÉ et al., 2008).

O propósito do trabalho de Fallon e O’Sullivan (2012) consiste em analisar se é possível usar ontologias para avaliar a QoE do serviço de IPTV diretamente do terminal do usuário em uma HAN (*Home Area Network*). A estrutura de classes da ontologia proposta (Figura 14) foi construída, considerando o reuso das ontologias já existentes, FOAF (BRICKLEY; MILLER, 2000) e WGS84 (BRICKLEY, 2003) foram usadas, respectivamente para representar o contexto do usuário e para obter a localização geográfica de um usuário.

Figura 14 - Estrutura da KB para armazenar a experiência do serviço

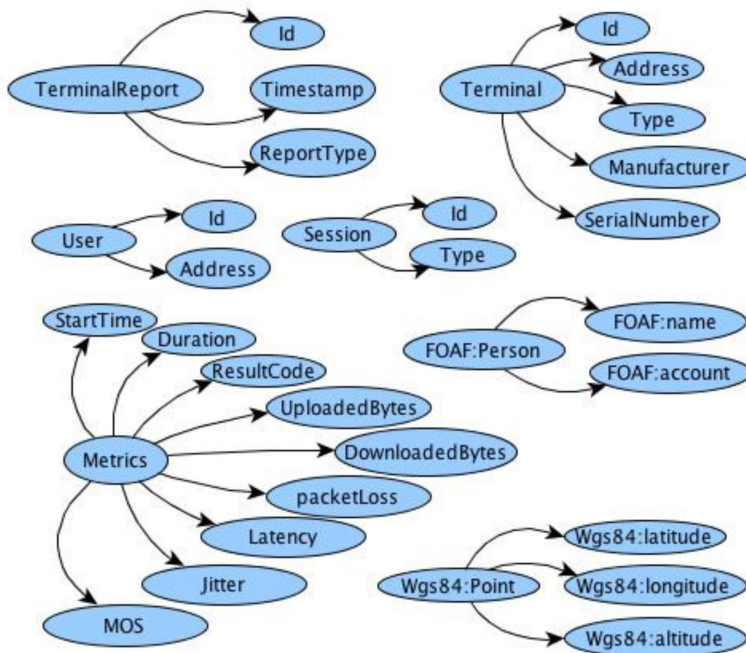


Fonte: Fallon e O'Sullivan (2012)

Através de um ambiente simulado, um módulo de avaliação da QoE foi instalado nos terminais dos usuários e, a cada intervalo de tempo pré-definido as avaliações são enviadas em formato de relatório para uma máquina controladora. Os relatórios no formato XML, semanticamente anotados (SAWSDL, 2007), transportam informações quantitativas (métricas de QoS, tais como vazão e atraso) e qualitativa (métrica de QoE, MOS do vídeo calculado no terminal do usuário) para a máquina controladora. Os relatórios são transportados por um protocolo próprio que noticiam a UX e o contexto no início da sessão, durante e na finalização da sessão. Na máquina controladora, o conhecimento dos relatórios dos terminais é semanticamente mapeado e armazenado numa KB. Esse conhecimento futuramente pode ser utilizado para analisar e otimizar a qualidade do serviço de IPTV na federação HAN. A estrutura da KB utilizada para armazenar a experiência do serviço é mostrada na Figura 14.

Os indivíduos RDF gerados a partir dos relatórios dos terminais contêm as propriedades de dados mostradas na Figura 15. Segundo os autores, os dados dos relatórios dos terminais que foram identificados como sendo de particular interesse para análise são: tempo, equipamento físico, localização, usuário, sessão e métricas.

Figura 15 - Propriedade de dados mapeados dos relatórios dos terminais para conceitos na KB



Fonte: Fallon e O'Sullivan (2012)

Os dados brutos dos relatórios dos terminais (FALLON; O'SULLIVAN, 2011) são usados para extrair informações para inferir ou para conhecer a experiência do usuário no uso do serviço. Um *framework* foi desenvolvido para extrair e codificar a informação semântica a partir dos relatórios. O *framework* recebe os dados no esquema XML com SAWSDL e converte as informações de interesse em RDF. Todo relatório recebido passa pelo *framework* onde são separadas as informações semânticas e inseridas na KB.

Um cenário foi proposto (FALLON; O'SULLIVAN, 2011) para avaliar se é útil em termos de desempenho usar ontologias para recolher informações dos relatórios dos terminais. O cenário foi simulado no escopo de uma HAN ou confederação de HAN, com variações de sessões ativas (6-300), simultaneamente, do serviço de IPTV. Nos terminais foi configurado que de tempos em tempos (intervalo de 1 minuto), os relatórios, com informações do MOS e do delay, devem ser



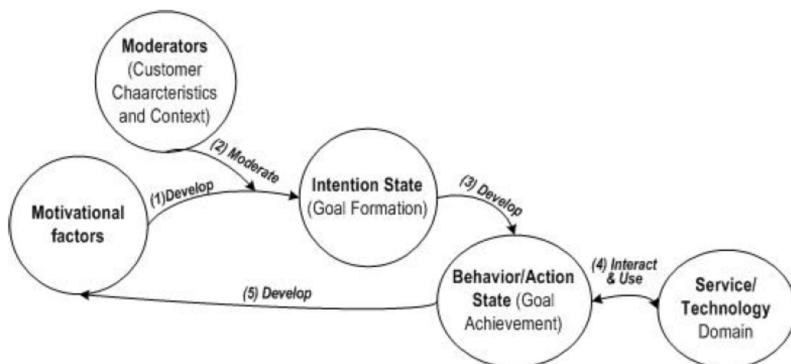
enviados periodicamente para a máquina controladora. Os resultados da avaliação demonstraram que popular uma KB com informações semânticas a partir de relatórios dos terminais é possível com uma carga de relatórios provenientes de seis terminais.

Em Laghari et al. (2010) é proposto um modelo para entrega de serviços baseado na experiência do cliente. A modelagem provê aspectos qualitativos e quantitativos da experiência humana. O modelo proposto é transformado para um modelo ontológico que permite a entrega de serviços orientada à experiência do cliente. O foco principal da pesquisa é determinado pelo seguinte cenário:

- Como conhecer os requisitos da experiência do cliente?
- Como associar a experiência do cliente com a ordem de serviço?
- Como entregar um serviço com base nos requisitos da experiência do cliente?

Inicialmente são definidos os conceitos para descrever a experiência do cliente no uso do serviço, incluindo fatores motivacionais, atributos e moderadores de contexto, intenções e domínio do serviço. A partir dessa modelagem é proposto o modelo geral da experiência do cliente, conforme ilustrado na Figura 16.

Figura 16 - Modelo geral da experiência do cliente



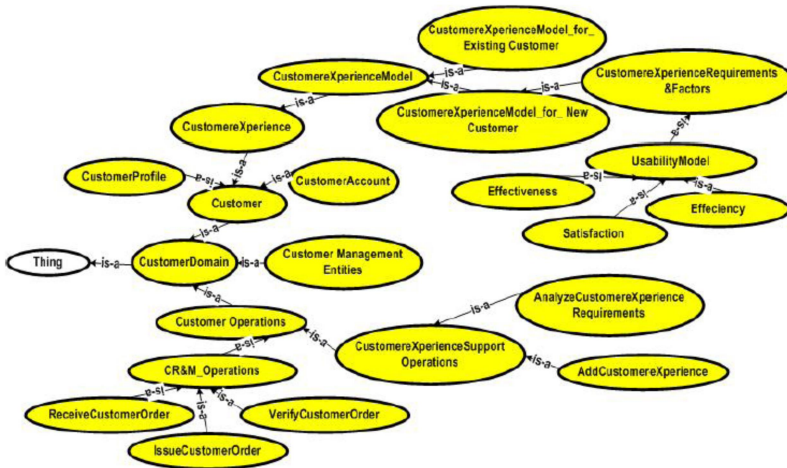
Fonte: Laghari et. al. (2010)

O modelo generalizado é baseado em Ciências Sociais e particularmente na Psicologia Humana, e é influenciado pela Teoria do Comportamento Planejado (AJZEN, 1991). Na proposta, para entender a experiência do cliente, os autores se concentram em apenas um fator da experiência do cliente, a usabilidade (satisfação, eficácia, eficiência),

além disso, assumiram que (i) existe algum modelo da experiência do cliente disponível, baseado em algum modelo da Psicologia Social, (ii) modelos de negócios são criados tendo em vista a informação da experiência do cliente. O modelo ontológico é definido a partir de três classes principais, modelo do cliente (Figura 17), domínio do serviço (Figura 18) e modelo de negócios (Figura 19).

A ontologia do domínio do cliente (Figura 17) foi desenvolvida com conceitos de Cliente, Operações do Cliente, Entidades de Gestão. Estes conceitos têm mais subclassificações conforme representado na Figura 17. Esta ontologia destina-se a fornecer fatores e requisitos da experiência do cliente em tempo de execução. Para simplificar, os autores modelaram apenas um fator da experiência do cliente, a “usabilidade” para um novo “pedido do cliente” e propuseram algumas operações de suporte da experiência do cliente que enriquecem as operações de CRM (*Customer Relationship Management*) definidas para preparar o pedido com os requisitos da experiência do cliente.

Figura 17 - Visão das classes da ontologia do domínio do cliente

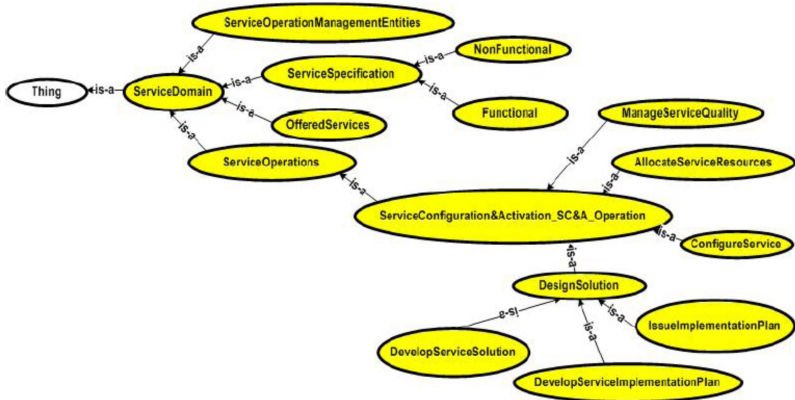


Fonte: Laghari et. al. (2010)

A ontologia do domínio do serviço (Figura 18) foi desenvolvida com os conceitos de especificações do serviço, operações dos serviços, serviços oferecidos e entidade de gerenciamento das operações dos serviços. Na Figura 18, com exceção do conceito serviços oferecidos, todos os demais possuem subclassificações. Os conceitos ontológicos do domínio de serviço permitem que o motor de política possa elaborar o

plano de serviço adequado, de acordo com os requisitos da experiência do cliente. A parte da ontologia de Solução de Design desenvolve características de serviço e parâmetros necessárias para suportar os requisitos da experiência do cliente.

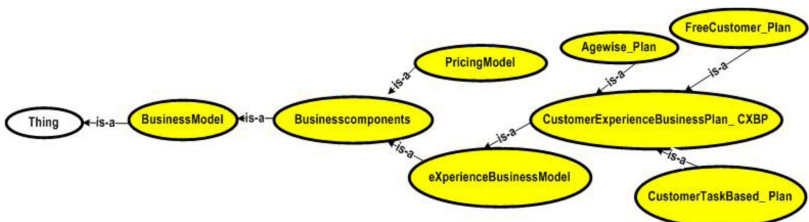
Figura 18 - Visão das classes da ontologia do domínio do serviço



Fonte: Laghari et. al. (2010)

A ontologia do modelo de negócio (Figura 19) é proposta para que a organização possa fazer melhor uso comercial das informações da experiência do cliente e com isso desencadear algum plano de negócios adequado e em conformidade que cumpram os objetivos dos negócios, combinando sentimentos do cliente. Os autores consideram que a classe mais importante neste modelo é o Plano de Negócio da Experiência do Cliente (CXBP) que inclui alguns possíveis planos de negócios com base nos atributos dos clientes ou informações de contexto, ou seja, idade, sexo, contrato, localização ou requisitos de usabilidade.

Figura 19 - Visão das classes da ontologia de negócios



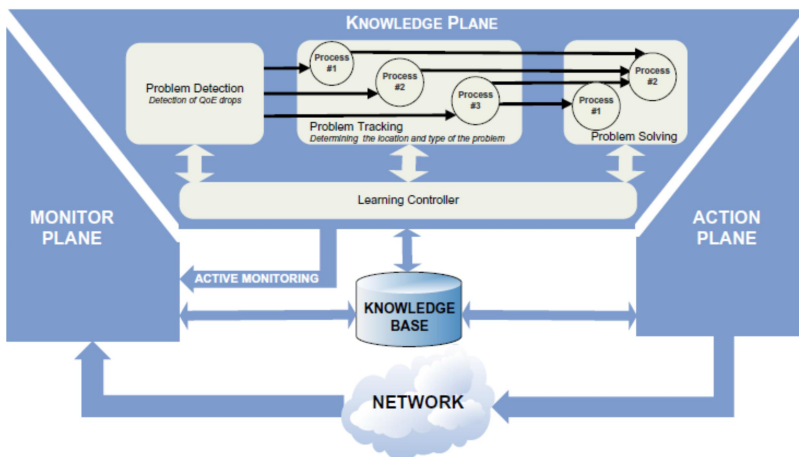
Fonte: Laghari et. al. (2010)

Para fazer a gestão dos serviços, os autores sugerem uma arquitetura funcional, composta pelo motor de políticas, KB e atuadores. O motor de política é baseado em regras, condição-ação. A base do conhecimento contém os conceitos, fatos, relações e os mecanismos para combiná-los para resolução dos problemas que podem ocorrer. O modelo ontológico é armazenado na KB para permitir que o motor de políticas possa fazer uso das informações da experiência do cliente com políticas adequadas. E os atuadores executam as ações de regras para alcançar os objetivos desejados.

A proposta não foi validada, mas, um cenário genérico foi apresentado para explicar o funcionamento dos componentes descritos acima. Os pedidos dos clientes são anotados semanticamente em SAWSDL, com conceitos ontológicos. As anotações podem incluir fatores da experiência do cliente e planos de negócios.

Em Latré et al. (2008) é apresentada uma abordagem semântica para otimização autônoma da QoE em redes de acesso multimídia. Os autores se concentraram na proposição de uma arquitetura autônoma e genérica para a otimização da QoE, composta pelos planos monitor, conhecimento e ação. Uma robusta KB foi projetada para dar suporte para cada um dos planos da Figura 20.

Figura 20 - Arquitetura para a maximização da QoE em serviços multimídia



Fonte: Latré et. al. (2008)

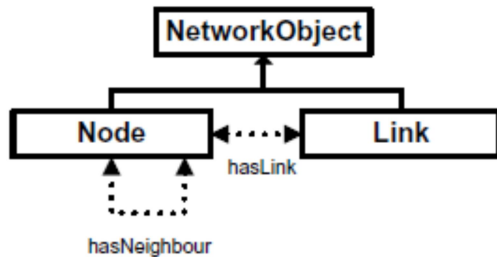
A ontologia, proposta por quatro submodelos ontológicos, é

capaz de classificar informações do Plano Monitor para disparar alarmes, caso um valor monitorado esteja fora dos limites declarados em uma política. O comportamento autônomo dos componentes da arquitetura proposta permite que ações de otimizações da QoE possam ser executados em resposta a degradação da QoE.

A KB é enriquecida com o conhecimento dos submodelos ontológicos propostos, incluindo os modelos da (i) topologia da rede, (ii) serviços disponíveis, (iii) otimizações da QoE e (iv) monitoramento dos serviços na rede de acesso. Esses submodelos serão apresentados a seguir.

A Figura 21 mostra como a topologia da rede pode ser modelada através de uma ontologia. A rede é definida como sendo um conjunto de objetos, sendo cada objeto representado por um *link* ou um nodo. A relação entre estes dois conceitos mostra como os nodos estão conectados ao *link* e vice e versa. O relacionamento *hasNeighbour* irá ser autonomamente deduzido usando regras em SWRL. No modelo dos autores, dois nodos são considerados vizinhos se eles compartilharem o mesmo *link*.

Figura 21 - Submodelo da ontologia da topologia da rede

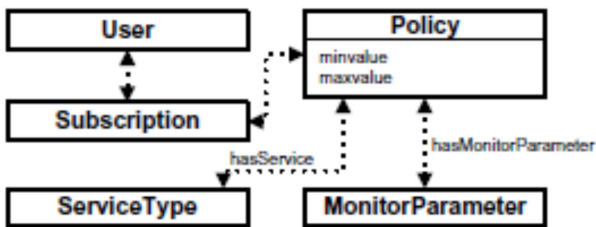


Fonte: Latré et. al. (2008)

Para a modelagem dos serviços e políticas por tipo de assinante, os autores evidenciaram um submodelo da ontologia, conforme ilustra a Figura 21. Essa parte da ontologia permite o operador modelar quais usuários tem acesso a rede, o tipo de assinatura e que serviços eles podem acessar. Tal como mostra a Figura 22, uma instância do conceito usuário pode ter uma assinatura (por exemplo, assinatura da Internet de alta velocidade) e cada assinatura pode ter acesso a um ou mais serviços (por exemplo, um serviço de TV por difusão), denotado pelo conceito *ServiceType*. Além disso, um operador pode aplicar uma política para cada tipo de serviço e assinatura.

A política foi definida contendo um valor máximo e mínimo para qualquer combinação de conceitos (*MonitorParameter*, *ServiceType* e *Subscription*). Esses valores descrevem o valor máximo e o valor mínimo que uma instância do conceito *MonitorParameter* (por exemplo, perda de pacotes) deve ter para ser classificada como um *MonitorParameter* do tipo normal. A política é definida para a combinação de *ServiceType*, *Subscription* e *MonitorParameter*. Se um valor do monitor não se encaixa no intervalo especificado na política, as regras especificadas em SWRL podem deduzir autonomamente conceitos problemáticos relacionados com o usuário ou com o serviço e com isso, através da ontologia é possível alertar o módulo de “detecção de problema” da arquitetura proposta (Figura 20).

Figura 22 - Submodelo da ontologia dos serviços disponíveis em uma rede de acesso



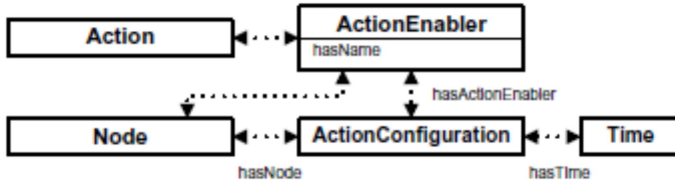
Fonte: Latré et. al. (2008)

Para a provisão de informações sobre a otimização da QoE e configurações históricas/atuais de cada nodo, foi criado o submodelo ontológico apresentado na Figura 23. Este submodelo irá descrever as capacidades presentes no módulo Plano de Ação (Figura 20) e a configuração de cada nodo num dado instante de tempo. A pretensão dos autores é descrever tanto informações estáticas (as diferentes ações de alto nível, como essas ações podem ser implementadas e as capacidades de otimizações de QoE de cada nó da rede) quanto informações dinâmicas (configuração atual e passada para cada nó em termos de otimizações de QoE).

No submodelo da Figura 23, instâncias do conceito *Action* tem diferentes instâncias *ActionEnabler* (e.g., gerar pacotes de paridade FEC - *Forward Error Correction*, decodificar paridade de pacotes e reconstruir o fluxo), que representam o conhecimento estático do Plano de Ação. O conhecimento dinâmico do Plano de Ação é representado através do conceito *ActionConfiguration*, que define quais instâncias do

conceito *ActionEnabler* existem por um determinado tempo e nodo. Isso possibilita incorporar informações passadas, o que pode ser necessário para o passo de raciocínio do Plano de Conhecimento.

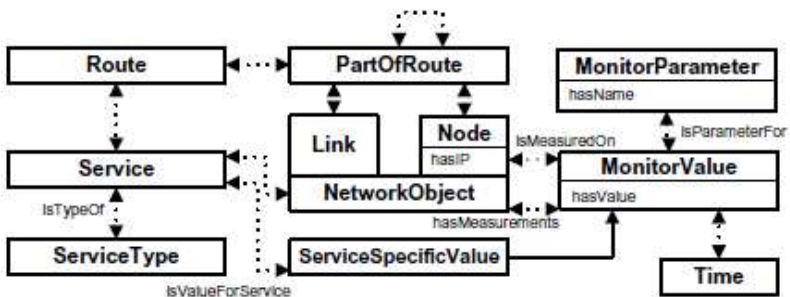
Figura 23 - Submodelo das capacidades de otimização da QoE e informações de configurações presentes e passadas de cada nodo



Fonte: Latré et. al. (2008)

Para representar o conhecimento das informações do Plano Monitor (Figura 20) foi proposto o submodelo ontológico da Figura 24. A Figura 24 mostra que este é o submodelo mais avançado dos quatro submodelos propostos, o mesmo foi proposto para descrever o monitoramento dos serviços na rede de acesso. Os conceitos *Route* e *PartOfRoute* podem ser deduzidos usando SWRL. A combinação de SWRL e OWL pode ser usada para deduzir anomalias na rede, tais como, um valor ruim proveniente de um parâmetro monitorado, de um nodo, entre outros.

Figura 24 - Submodelo da ontologia do monitoramento dos serviços na rede de acesso



Fonte: Latré et. al. (2008)

No submodelo representado pela Figura 24, encontram-se descritos diferentes aspectos, incluindo, o conjunto completo de valores medidos do monitor; os diferentes serviços que estão rodando na rede;

as diferentes rotas para cada serviço, etc. Portanto, vários novos conceitos são definidos a partir deste submodelo ontológico, a saber.

O conceito *MonitorValue*, derivado do conceito *ServiceSpecificValue* é introduzido. Estes conceitos descrevem os valores reais medidos e têm um relacionamento com um determinado valor de *MonitorParameter*, que descreve uma determinada métrica que pode ser monitorada (por exemplo, atraso, *jitter* e perda de pacotes).

As instâncias do conceito *ServiceSpecificValue* representam valores do monitor diretamente relacionado a um serviço específico (por exemplo, a taxa de perda de pacotes medida por um serviço de VoD rodando de um nodo A para um nodo B) e, portanto, tem um relacionamento com um novo conceito *Service*.

O conceito *Service* define um serviço real rodando na rede entre dois ou mais terminais. Como diferentes tipos de serviços (por exemplo, TV por difusão, VoD, jogos on-line) podem rodar na rede, um *Service* terá um *ServiceType*.

Além disso, o conceito tem *MonitorValue* tem um relacionamento com um conceito pré-definido *Time* para marcar o tempo em que uma medição foi realizada e as relações com o conceito *Nodo* e *NetworkObject* para marcar onde uma medição foi realizada e com que parte da rede ocorreu a mediação.

Finalmente, cada conceito *Service* pode ter uma ou mais rotas, definida como uma lista ligada do conceito *PartOfRoute*. Essa informação também pode ser deduzida usando OWL e SWRL.

Os autores apresentaram um cenário de transmissão de vídeo para demonstrar o comportamento autônomo dos componentes propostos. Foi assumida uma política para o serviço de TV por difusão, com taxa de perda de pacotes entre 0% e 0,5% e largura de banda utilizada de 0% e 0,90% da capacidade total do *link*. As regras SWRL são especificadas para determinar se um determinado valor monitorado pode conduzir a anomalias da rede, comparando-o com os valores de referência descritos na política. Com isso, a ontologia pode disparar alarmes quando os valores estiverem fora dos limites aceitáveis e o comportamento autônomo dos componentes da arquitetura proposta são programados para realizarem ações para corrigir a degradação da QoE. As ações para a otimização da QoE são realizadas no componente Plano de Ação, com base nas informações monitoradas, incluindo:

- Alterações na taxa de bits do vídeo para resolver problemas de congestionamento ou, incremento da qualidade do vídeo e
- Alteração do número de pacotes de FEC para combater a



perda de pacotes devido a erros de bits.

No cenário apresentado, o componente Plano de Ação decide que o cliente deve receber um vídeo de alta largura de banda e um número de pacotes FEC. Para os autores, isto pode corresponder a uma situação em que há perda de pacotes em um *link* na rede doméstica ou na linha xDSL.

### 3.4 QUALIDADE DO CONTEXTO

Quando o contexto é considerado na provisão de um serviço sensível ao contexto, é necessário analisar a qualidade da informação (QoC – *Quality of Context*) que será entregue. A QoC refere-se a toda e qualquer informação que ajuda a descrever a qualidade da informação usada como informação de contexto.

Informações de contexto são usadas para adaptar automaticamente o conteúdo ou serviços fornecidos. Portanto, para Nazário e Dantas (2015), a imperfeição de informações de contexto tem um impacto significativo sobre a experiência dos usuários com serviços sensíveis ao contexto (NAZÁRIO; DANTAS, 2015). De encontro com os autores, nesta pesquisa a QoC foi estudada como uma dimensão de influência na UX.

A QoC é medida através da análise de parâmetros que ajudam a mensurar a qualidade. Tomando como base um *web service* que disponibiliza informações de estoque de livros para uma página de comércio eletrônico de venda de livros, mas que no entanto, envia as informações com atraso, o parâmetro de QoC “atualidade” receberia um valor baixo da fonte provedora do contexto (*web service*) e esta informação não deveria ser usada pelo consumidor do contexto (página de comércio eletrônico), pois o fato das informações estarem desatualizadas reflete negativamente na UX, além de causar sérios problemas administrativos para a organização.

Contextualizando a QoC para a área de sensoriamento de sinais biomédicos, num AAL, onde os sinais vitais (e.g., temperatura, pressão arterial, pulso, batimentos cardíacos) são enviados para uma unidade de atendimento remoto, os parâmetros de QoC “atualidade” e “precisão”, por exemplo, devem ter um valor alto para que as informações do provedor de contexto (sensores de sinais vitais do paciente) possam ser usados pelo consumidor do contexto (aplicação de assistência a saúde).

### 3.4.1 Parâmetros e Mensuração da Qualidade do Contexto

Em Nazário e Dantas (2015) é apresentado uma modelagem abrangente de todos os parâmetros do QoC existentes na literatura. A escolha dos parâmetros da QoC dependerá do tipo de aplicação e da fina granularidade das informações de contexto. Todos os parâmetros assumem valores no Intervalo [0..1] e cada parâmetro tem uma forma de medição, conforme sintetização na Tabela 15.

A Tabela 15 apresenta os parâmetros da QoC, uma descrição, a forma com que cada parâmetro é mensurado (Equação 1, Equação 2 e Equação 3) e exemplos de aplicação da QoC em informações de contexto de sinais vitais em um AAL.

Tabela 15 - Parâmetros da QoC no sensoriamento de sinais vitais

Parâmetro de QoC	Descrição do parâmetro de QoC	Mensuração e Exemplo
Coverage (C)	Refere-se a validade da informação de contexto (sensor), sendo definido um conjunto de possíveis valores para a informação de contexto e um range (limite superior e limite inferior) para cada sensor.	<p><math>C = 1</math>, se o valor está na faixa válida  <math>C = 0</math>, nos demais casos</p> <p><b>Exemplo:</b> temperatura corporal é considerada válida entre 34°C e 40°C  <math>C = 1</math>, se temperatura = 38°C  <math>C = 0</math>, se temperatura = 0°C</p>
Up-to-dateness (U)	Indica a idade da informação de contexto usando um timestamp (lifetime, chamado de tempo de vida da informação). Para a variável lifetime é definida com um valor em que a informação se torna “velha”(obsoleta)	<p>Idade = tempo_vida – tempo_atual            Se idade &lt; tempo_vida,</p> $U = \frac{1 - idade}{tempo\_vida} \quad (1)$ <p><math>U = 0</math>, nos demais casos.</p> <p><b>Exemplo:</b>            Se tempo_vida = 10            Idade = 0 =&gt; <math>U = 1</math>            Idade = 5 =&gt; <math>U = 0,5</math>            Idade = 10 =&gt; <math>U = 0</math></p>
Precision (P)	Descreve como a informação de contexto fornecida reflete com a realidade, sendo calculada pela diferença entre o valor real e o valor medido pelo sensor.	$P = 1 - \left  \frac{valor_{real} - valor_{medido}}{valor_{real}} \right  \quad (2)$ <p><b>Exemplo:</b>            pulso real = 90; pulso medido = 93, logo, <math>P = 0,96</math></p>
Completeness (Cm)	Refere-se a quantidade de informação provida por um objeto de contexto (sensor). É calculada pela relação entre o n°. de atributos disponíveis e o total de atributos de um objeto de contexto, Considerando que os atributos de contexto	<p><b>Exemplo:</b>            Os sensores de temperatura corporal e pulso possuem apenas um atributo, logo,            - Se o valor da temperatura = disponível, <math>Cm = 1</math>, caso contrário <math>Cm = 0</math>;            - Se o valor do pulso = disponível, <math>Cm = 1</math>, caso contrário <math>Cm = 0</math>.</p> <p>O sensor de pressão tem dois atributos (valor da pressão sistólica e diastólica), considerando que</p>

Parâmetro de QoC	Descrição do parâmetro de QoC	Mensuração e Exemplo
	(informações de um sensor) podem ter pesos diferentes, o cálculo é feito pela razão do somatório dos pesos dos atributos disponíveis pelo somatório dos pesos de todos os atributos.	ambos possuem o mesmo peso, logo, - Se o valor da pressão sistólica <b>E</b> diastólica = disponíveis, <b>Cm = 1</b> ; - Se o valor da pressão sistólica <b>OU</b> diastólica = disponível, <b>Cm = 0.5</b> ; - Se o valor da pressão sistólica <b>OU</b> diastólica = indisponíveis, <b>Cm = 0</b> .
Significance (S)	Refere-se a importância de informação de contexto, sendo mensurada pela razão entre o VC (valor crítico) e VCMax (valor crítico máximo). O parâmetro <b>Significance</b> é usado para alertar situações que exigem atenção. O parâmetro <b>Coverage</b> indica se o valor está na faixa válida, mas, os valores podem ser críticos.	$S = \frac{CV}{CV_m} \quad (3)$ <p><b>Exemplo:</b> Se temperatura = 39°C, o paciente está com febre, porém, <b>Coverage</b> = 1, pois 39°C é um valor válido. Assim, <b>S = 1</b>, quando os valores são válidos (faixa do coverage), mas esperados; <b>S = 0</b>, para os demais valores dentro do considerado normal (faixa do coverage), porém, não esperados.</p> <p>-Se (temperatura &lt; 35) <b>OU</b> (temperatura &gt; 37,5) =&gt; <b>S = 1</b>; senão, <b>S = 0</b>; - Se (pulso &lt; 60) <b>OU</b> (pulso &gt; 100) =&gt; <b>S = 1</b>; senão, <b>S = 0</b>. - Se (diastólica &gt; 90) <b>OU</b> (sistólica &gt; 140) =&gt; <b>S = 1</b>; senão <b>S = 0</b>.</p>

Fonte: Adaptado de Nazário e Dantas (2015)

De acordo Yasar et al. (2012) um valor geral de qualidade pode ser calculado levando em consideração os parâmetros da QoC e um peso associado para cada parâmetro. Os pesos devem ser definidos e o valor é calculado para cada sensor ou provedor do contexto. O valor global da QoC está contido no intervalo [0..1], onde valor "1" significa adesão completa à qualidade da conexão, e "0" significa ausência total de qualidade de conexão. Para Yasar et al. (2012), a QoC global pode ser calculada pela Equação 4.

$$QoC = \frac{U * PU + C * PC + P * PP + Cm * PCm + S * PS}{PU + PC + PP + PCm + PS} \quad (4)$$

É importante ressaltar que em Nazário e Dantas (2015) foi proposta uma arquitetura de gerenciamento do contexto, avaliada num AAL ubíquo, onde o usuário, usando uma plataforma de *eHealth* (*Cooking-hacks*) realiza as medições de sinais vitais (pressão arterial, pulso, oxigênio no sangue, temperatura corporal) e as envia para serem processadas pelo gerenciador de contexto. O foco consistiu na avaliação dos parâmetros de QoC, para verificar até que ponto as informações de contexto eram atualizadas, válidas, precisas, completas e significativas.

A análise permitiu evidenciar que a avaliação da QoC num ALL ubíquo é importante e útil para detectar anomalias ou inconsistências em sensores, gerar alertas, ativar sensores de backup, descartar dados com QoC insuficiente, selecionar provedores apropriados, entre outras. Contudo, na arquitetura proposta por Nazário e Dantas (2015) não é prevista a distribuição do contexto em redes de transporte de dados.

### 3.5 CONSIDERAÇÕES DO CAPÍTULO

Neste capítulo foi realizada uma explanação abrangente dos fatores de influência na qualidade do serviço em geral, qualidade do serviço em sistemas de comunicação de dados e na qualidade da experiência. Também foi abordado o tema de qualidade de contexto, por considerar que em muitos ambientes e sistemas a UX pode ser afetada se a QoC não for tratada. Ao longo do capítulo foi possível evidenciar que muitas propostas de provisão e entrega de serviços ainda visualizam a QoE como uma extensão da QoS, porém, poucas analisam o contexto para a predição da QoE.

Considerando que a QoE é um constructo multidimensional da experiência humana, onde vários fatores de influência podem impactar a percepção do usuário no uso de um serviço, as propostas devem ser centradas nas necessidades dos usuários e pautadas numa abordagem interdisciplinar. Com este olhar, realizou-se uma revisão sistemática da literatura para evidenciar a aplicação da QoE, as bases conceituais que ampararam as propostas, formas de mensuração, dimensões que tem sido consideradas e o uso de ontologias como forma de representação do conhecimento. A partir das pesquisas, observou-se que apenas três propostas usam abordagens semânticas para representar o conhecimento da QoE, sendo que em apenas uma delas, a QoE é considerada para além da QoS.

Com base na análise das pesquisas observou-se, ainda, que a dimensão contexto tem sido pouco explorada. Nenhuma das propostas aborda a questão do conteúdo, levando em consideração o significado semântico do ponto de vista do usuário. Além disso, a maioria das propostas são projetadas para a arquitetura atual da Internet e com isso limitando-se em prover a mesma QoE para os diferentes usuários. E mais, na eventual degradação dos parâmetros de QoS, os recursos de rede não são otimizados, podendo refletir negativamente na experiência do usuário.

Após toda a contextualização do segundo e terceiro capítulos foi possível compreender o contexto da provisão de serviços, os tipos de

conhecimentos, a abordagem interdisciplinar como condição necessário para o estudo e os aspectos tecnológicos emergentes para a gestão do conhecimento da QoE. E com isso, conclui-se que o segundo objetivo específico foi parcialmente contemplado.

O próximo capítulo aborda a SDN como arquitetura promissora para a provisão de serviços centradas nas necessidades dos usuários.



## 4 REDE DEFINIDA POR SOFTWARE

O modelo de comunicação que irá permitir a comunicação entre os agentes e a disseminação do conhecimento são artefatos nativos da área da Mídia (SCHMID; STANOEVSKA, 1998). Segundo Perassi, Pavanati e Neves Júnior (2010), neste modelo o processo de comunicação é constituído por duas instâncias:

- A informação semântica, de conteúdo, lógica que caracteriza a ideia que é transmitida de um emissor para um receptor.
- A informação sintática, estética, de material que decorre diretamente da mídia ou da participação da mídia no processo de comunicação.

Esses dois tipos de processos de informação (midiático e conteúdo) formam o processo final de comunicação. Assim, para tentar garantir a total aceitabilidade do usuário no uso do serviço, é preciso considerar tanto o meio físico que transporta o conteúdo, quanto o conteúdo em si, pois ambos os processos (provenientes da mídia e do conteúdo) podem influenciar na UX.

Seguindo a lógica do processo de comunicação descrito por Perassi, Pavanati e Neves Júnior (2010), a primeira instância foi abordada na seção 3.3, onde foram evidenciadas a modelagem e a representação do conhecimento da QoE. Agora, é necessário dissertar sobre a segunda instância do modelo de comunicação, ou seja, a infraestrutura usada para o transporte dos dados.

Considerando que a QoE é mais focada para o usuário e não para a tecnologia, buscou-se uma arquitetura de rede capaz de operar com abordagem *top-down* (do usuário para a rede), diferentemente da abordagem *bottom-up* (da rede para o usuário). Essa abordagem é passível de implementação, usando SDN (*Software-defined networking*) (ONF, 2012).

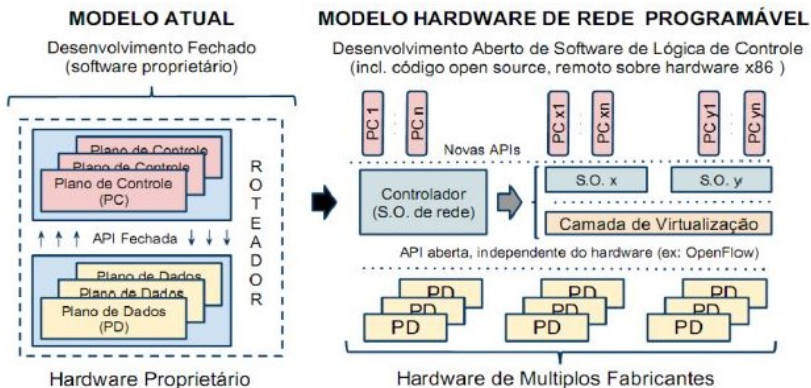
A SDN tem sido usada como infraestrutura de rede capaz de ser programada sob demanda para melhorar a QoE no uso do serviço, como por exemplo, na avaliação do serviço de IPTV centrada no usuário, num sub sistema multimídia IP (HUONG-TRUONG, 2013), na gestão de falhas, com o propósito de validar um modelo cognitivo para a gestão autônoma da rede (KIM, 2013), na gestão dos centros de dados da google (JAIN et al., 2013).

#### 4.1 REDE DEFINIDA POR SOFTWARE E ARQUITETURA TCP/IP

Rede definida por software (ONF, 2012) é uma abordagem inovadora na área de redes de computadores, uma vez que propõe um modelo programável e flexível para controlar o encaminhamento e roteamento dos pacotes. Ou seja, um software controlador, residente numa máquina da rede (local, remota, centralizada/distribuída), determina as ações dos fluxos na tabela de encaminhamento dos switches, de acordo com a demanda.

A arquitetura TCP/IP é inflexível no sentido de permitir que a rede seja programada sob demanda, uma vez que tanto o plano de controle quanto o plano de dados são executados no mesmo equipamento, impossibilitando que ações de encaminhamentos não previstas nesses protocolos, sejam realizadas. Na SDN (ONF, 2012) tanto o plano de controle quanto o plano de dados são separados, permitindo assim, total flexibilidade para alterar, em tempo de execução a tabela de encaminhamento dos *switches* para que os fluxos sejam enviados por outras rotas no caso de haver degradação de qualidade, congestionamento na rede etc. A Figura 25 apresenta um comparativo do modelo atual e modelo de rede programável.

Figura 25 - Comparativo do modelo de hardware de rede com o modelo Programável



Fonte: Rothenberg, et al. (2011)

A arquitetura atual é formada basicamente por hardware e software proprietário, dedicados ao encaminhamento de pacotes. O



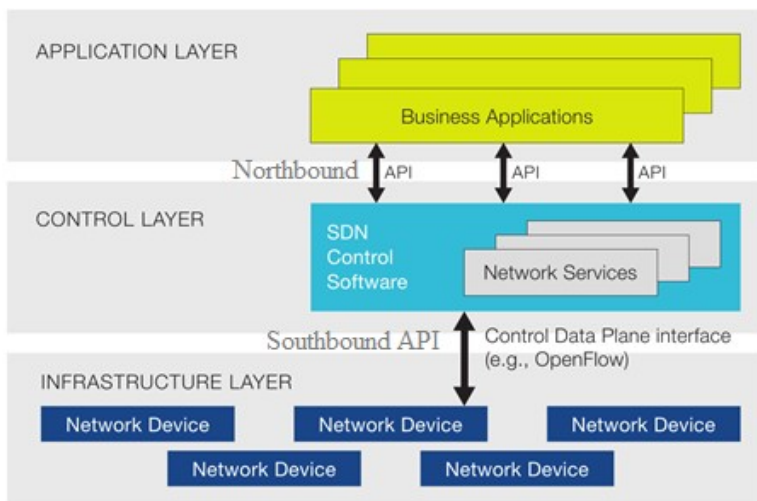
software dessas aplicações realiza a tomada de decisão de roteamento, não sendo possível qualquer tipo de alteração a não ser às disponibilizadas pelo proprietário. Os dados de roteamento e as rotas estabelecidas são realizados de acordo com uma API proprietária, que preconiza a forma de encaminhamento dos dados para o hardware.

O modelo programável apresenta uma estrutura flexível, com hardware independente do fabricante, uma camada de controle separada e APIs abertas e independentes do hardware.

## 4.2 ARQUITETURA SDN

A arquitetura SDN proposta pela ONF (2012), apresentada na Figura 26, é uma arquitetura estruturada em camadas (gerenciamento, controle e dados), que provê artefatos para que agentes humanos ou softwares possam gerenciar serviços de rede através da abstração de funcionalidade de nível superior. Isto é feito através do desacoplamento do sistema que toma decisões sobre onde o tráfego é enviado (o plano de controle) a partir dos sistemas subjacentes que encaminham o tráfego para o destino selecionado (o plano de dados).

Figura 26 - Arquitetura SDN



Fonte: ONF (2012)

Para que o plano de controle possa se comunicar com o plano de dados é necessário um protocolo. De acordo com a ONF (2012), o

*OpenFlow* (MCKEOWN et al., 2008) é a primeira interface padronizada de comunicação usada entre as camadas de controle e encaminhamento de dados. O *OpenFlow* provê uma programação simplificada para acessar e manipular do plano de encaminhamento dos dispositivos de rede (*switches* e roteadores, tanto físicos quanto virtuais) através de uma interface padronizada.

#### 4.2.1 Camada de Aplicação

Nesta camada estão localizadas todas as aplicações de negócios da organização, que oferecem serviços de rede, tais como virtualização, *firewalls*, balanceadores de carga, sistema de gestão da QoS/QoE, entre outros.

#### 4.2.2 Camada de Controle

O principal componente desta camada é o controlador padrão SDN. O controlador é responsável por tomar decisões, adicionar e remover entradas na tabela de fluxos dos swiches, de acordo com o objetivo desejado. Cada instância do controlador é instalada numa máquina da rede e se comunica com a parte do software dos swiches usando um canal seguro. O controle da rede pode ser realizado de forma centralizada ou distribuída, com adoção de softwares controladores. Atualmente existem alguns controladores, tais como NOX (GUDE et al., 2008), POX (AL-SHABIDI; MCCAULEY, 2015), *Floodlight* (PROJECT FLOODLIGHT, 2014), Ryu (RYU, 2014), entre outros. Todos eles possuem suporte ao protocolo *OpenFlow*.

O controlador é capaz de se comunicar com os swiches do plano de dados, via *API southbound*, usando um protocolo aberto, tal como o *OpenFlow*. Para permitir uma gestão dinâmica da rede, além do controlador padrão, é possível desenvolver módulos de aplicações de controle, tais como controle de admissão, reserva e alocação de recursos, roteamento, visão geral da rede, status dos dispositivos, coletores estatísticos entre outros. Todas essas aplicações permitem a gerência de políticas de rede por usuários; grupo de usuários (e.g., largura de banda, QoS), compartilhamento a mesma infraestrutura e políticas (e.g., segurança). Assim, o controlador atua como uma interface entre as aplicações de controle da rede e os elementos do plano de dados.

Além dos módulos de aplicações de rede, esta camada pode contemplar Base de Dados (BD), com informações da topologia da rede,

recursos disponíveis, recursos alocados e, base de regras. Em Bari et al. (2013), além de módulos de aplicação para o gerenciamento e o controle da SDN, foi proposta uma base de regras. Essa base pode ser usada pelo plano de aplicação para traduzir as políticas de rede de alto nível em regras de controle. O controlador e as aplicações de controle (e.g., roteamento) usam essas regras para calcular as entradas na tabela de fluxo para cada *switch*.

Tanto as aplicações de rede, quanto as bases são acessíveis via *Northbound* API. Ou seja, para cada módulo de aplicação de rede desenvolvido haverá uma interface de comunicação entre as camadas de controle e aplicação.

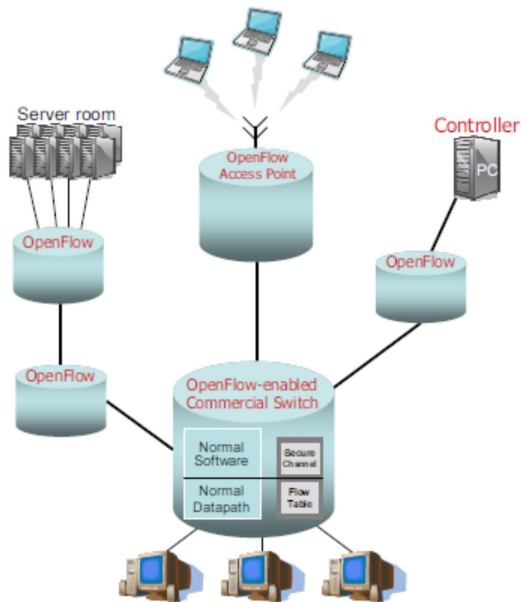
### 4.2.3 Camada de Dados

A camada de dados é composta por dispositivos de redes com algum protocolo padronizado. O *OpenFlow* (MCKEOWN et al., 2008) é o primeiro protocolo aberto que disponibiliza uma API para a interconexão dos planos de controle e dados, sendo, portanto, o mais indicado. No entanto, qualquer API que forneça uma camada de abstração da rede física para o controlador pode ser adotado. Outros protocolos de comunicação alternativos, NetConf (IETF, 2006a), BGP (IETF, 2006b), MPLS-TP (IETF, 2011a), XMPP (IETF, 2011b), OVSDb (IETF, 2013), com funções bem definidas, tem sido usados em SDN.

## 4.3 PROTOCOLO OPENFLOW

É o primeiro padrão SDN que usa protocolo aberto para prover a interconexão das camadas de controle e encaminhamento. Na Figura 27 é possível ver uma rede com *switches OpenFlow*. Os elementos de roteamento desse ambiente virtual são roteadores virtualizados que fazem a comutação de fluxos de forma controlável e programável via comandos do controlador.

Figura 27 - xemplo de rede com *switches* e roteadores *OpenFlow*



Fonte: McKeown et al. (2008)

A API *OpenFlow* define um fluxo de tráfego como uma sequência de pacotes, contendo informações que permitem a sua identificação em vários níveis (VLANs, MAC address, IP, porta), conforme ilustra a Figura 27 (MCKEOWN et al., 2008). Desta forma, como exemplos de fluxo, têm-se todos os pacotes em uma faixa de endereços IP (roteamento IP tradicional), uma conexão TCP em uma porta específica ou todos os pacotes pertencentes a uma mesma VLAN.

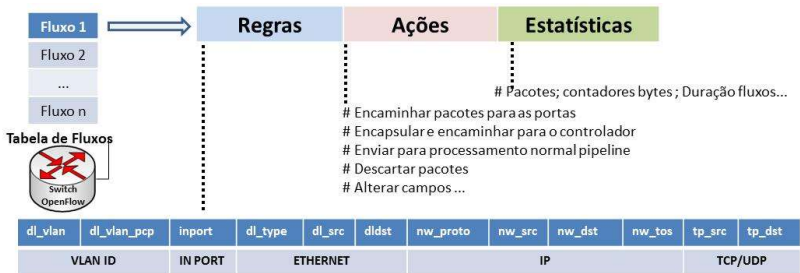
A especificação *OpenFlow* também oferece APIs padronizadas para o controlador gerenciar o plano de dados na estrutura de comutação. O controlador usa a API *OpenFlow* para instalar as regras de encaminhamento de tráfego nas tabelas de fluxos dos *switches*, descobrir a topologia da rede, monitorar o fluxo das estatísticas, ver o status *up/down* dos dispositivos e fluxos ativos.

A tabela de fluxo dos *switches* é composta por entradas. Cada entrada é composta por regras, ações e contadores estatísticos. As regras são formadas pela definição do valor de um ou mais campos do cabeçalho do pacote. Para cada regra é associado um conjunto de ações

que determinam a forma com que os pacotes serão processados e para onde devem ser encaminhados.

A Figura 28 apresenta uma visão geral de uma entrada da tabela *OpenFlow*. É possível visualizar exemplos de ações que podem ser associadas aos fluxos e exemplos de informações de contadores. Os contadores são usados para manter informações estatísticas de uso por tabela, fluxos, porta, bem como para a remoção de fluxos inativos.

Figura 28 - Entrada na tabela de fluxos de um *switch OpenFlow*



Fonte: adaptado de McKeown et al. (2008)

O Quadro 4 apresenta alguns exemplos de regras possíveis em uma *switch OpenFlow*, sendo elas aplicáveis, respectivamente, para realizar o encaminhamento de pacotes na camada de enlace, implementar um *firewall* e realizar encaminhamento de pacotes na camada de enlace utilizando VLANs.

Quadro 4 - Regras de um *switch OpenFlow*

Port	VLAN	ETH SRC	ETH DST	IP SRC	IP DST	IP Proto	L4 SRC	L4 DST	Ações
*	*	*	00:4F...	*	*	*	*	*	Porto 4
*	*	*	*	*	*	TCP	*	22	DROP
*	1	*	00:4F...	*	*	*	*	*	Porto 4, 6, 9

Fonte: Guedes et al. (2012)

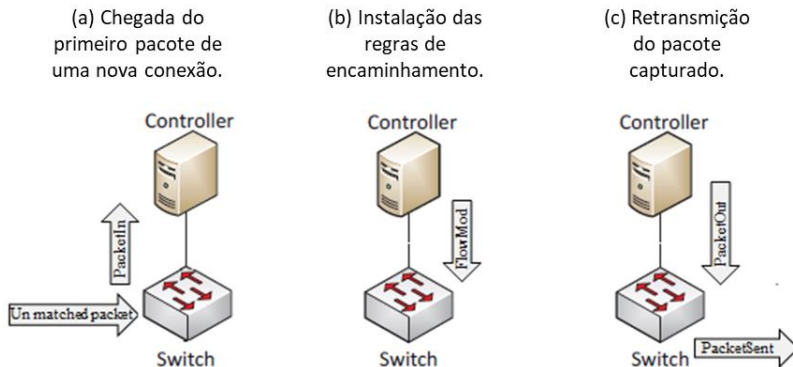
As entradas da tabela de fluxo podem ser interpretadas como decisões em cache, no hardware, ou no plano de controle, no software. Quando um pacote chega num *switch* com *OpenFlow* habilitado, os cabeçalhos dos pacotes são comparados com as regras das entradas das tabelas de fluxos, os contadores são atualizados e as ações

correspondentes são realizadas. Se não houver uma correspondência, o pacote completo, ou apenas o cabeçalho é encaminhado para o controlador, mantendo o pacote no *buffer* do hardware. Geralmente, os pacotes que chegam até o controlador correspondem ao primeiro pacote de um novo fluxo ou em função do tipo de pacote e aplicação. O controlador pode instalar uma regra no *switch* para que todos os pacotes de determinado fluxo (e.g., pacotes de controle, ICMP, DNS, DHCP ou de roteamento, OSPF, BGP) sejam enviados para o controlador para serem tratados individualmente.

#### 4.3.1 Troca de Mensagens entre Controlador e Switch OpenFlow

O controlador pode preconfigurar o *switch* com regras de reencaminhamento, bem como responder de forma reativa a solicitações dos *switches*, que são enviadas quando não há regra de encaminhamento de um pacote (Figura 29).

Figura 29 - Procedimento de instalação de um novo fluxo



Fonte: Adrichem, Doerr e Kuipers (2014)

Ao receber um pacote e não havendo regra de encaminhamento correspondente na tabela de fluxo, o *switch* OF envia uma mensagem *PacketIn* ao controlador (Figura 30(a)). O controlador responde com a instalação de um caminho usando uma ou mais mensagens de modificação *FlowMod* da tabela de fluxo (Figura 30(b)) e instrui o reenvio do pacote usando uma mensagem *PacketOut* (Figura 30(c)). As mensagens *FlowMod* são utilizadas na gerencia da tabela de fluxos, a qual o controlador determina a inserção, alteração ou remoção de uma

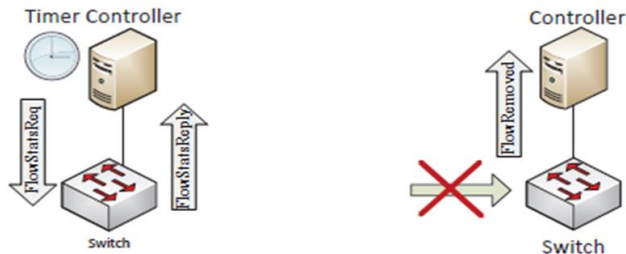
entrada na tabela de fluxo no *switch*.

Além de gerenciar o plano de encaminhamento, o protocolo *OpenFlow* também é capaz de solicitar estatísticas do contador de perdas e de injetar pacotes na rede para estimar o atraso no caminho. As mensagens *Flow-stats* são usadas na coleta de estatísticas do *switch*, sendo *request*, do controlador para o *switch*, ou *response*, do *switch* para o controlador. A Figura 30 ilustra a troca de mensagens sobre o estado do fluxo.

Figura 30 - Troca de mensagens do estado do fluxo entre o SDNC e *switch*

(a) Enquanto um fluxo está ativo, o controlador pode usar um temporizador ou outro evento para obter estatísticas específicas do fluxo.

(b) O fim de um fluxo é anunciado com o envio de um pacote *FlowRemoved* do *switch* para o controlador.



Fonte: Adrichem, Doerr e Kuipers (2014)

Usando as mensagens *PacketIn* e *FlowRemoved*, um controlador pode determinar quais os fluxos ativos existentes. Além disso, a mensagem *FlowRemoved* contém a duração, o pacote e a contagem de bytes do fluxo recentemente removido, permitindo que o controlador mantenha estatísticas sobre os fluxos passados. Essas informações em combinação com as mensagens de solicitação de estatísticas de fluxo (*StatsRequest* e *StatsReply*) consultadas periodicamente, permitem que o controlador obtenha informações do *throughput*, do descarte de pacotes, assim como também possibilita injetar pacotes na rede para monitorar o atraso no caminho fim a fim.

As estatísticas de tráfego no *OpenFlow* podem ser extraídas em diferentes níveis de abstração (e.g., porta, fila, fluxo), a partir de *switches* individuais, com base nos contadores nas tabelas de fluxo. O Quadro 5 apresenta as mensagens estatísticas que podem ser obtidas com a versão 1.0 do protocolo *OpenFlow*.

Quadro 5 - Estatísticas por fluxo, por porta e por fila

<i>Estatísticas por Porta</i>	<i>Estatísticas por Fluxo</i>	<i>Estatística por Fila</i>
Rx packet count Tx packet count Rx packet count Tx packet count Rx packet drop count Tx packet drop count Rx errors Tx erros Rx overruns	Rx packet count Rx byte count Duration in seconds Duration in nano seconds	Tx packet count Tx byte count Tx errors

Fonte: adaptada de Vejlø e Andersen (2014)

Ao receber uma solicitação de estatística por porta, o *switch* responde com valores Rx e Tx para a porta especificada (1ª coluna do Quadro 5). Se uma solicitação de estatística por fluxo é enviada ao *switch*, ela é respondida com valores mostrados na 2ª. coluna do Quadro 5. Também é possível solicitar estatísticas para uma fila (3ª. coluna do Quadro 5). A estatística da fila pode ser solicitada para uma fila específica, identificada por um número de porta e um ID da fila, ou as estatísticas podem ser solicitadas para todas as filas dentro do *switch*. Ao solicitar estatísticas para mais de um fluxo/porta/fila, a resposta será formatada como uma matriz, de modo que as estatísticas de cada fluxo/porta/fila individual possam ser extraídas de forma exclusiva (VEJLØ; ANDERSEN, 2014).

### 4.3.2 Suporte a Qualidade do Serviço com o Protocolo OpenFlow

O protocolo *OpenFlow* em constante evolução, desde a especificação 1.0 até a 1.4, passou por várias melhorias. Em cada versão são agregadas novas funcionalidades de redes e melhorias para suportar QoS. O suporte a QoS descrito a seguir é uma síntese da pesquisa realizada por Bari et al. (2013).

A partir de especificação *OpenFlow* 1.0 os pacotes pertencentes a um fluxo podem ser enfileirado numa fila específica de uma porta de saída. A fila pode ser configurada através de protocolos padrão, como SNMP, CLI e Netconf (IETF 2006a). Os controladores podem consultar parâmetros de configuração e estatísticas das filas existentes. Os *switches* podem reescrever o campo IP ToS (*Type of Service*) no cabeçalho IP, que pode ser utilizado para implementar mecanismos de



QoS como DiffServ (BLAKE et al., 1998). O suporte para reescrever os bits de notificação de congestionamento explícito (ECN - *Explicit Congestion Notification*) foi incorporado no *OpenFlow* desde a versão 1.1.0.

A ONF criou um protocolo auxiliar chamado OF-Config (ONF, 2014) para suportar a configuração de várias características de um *switch OpenFlow*. OF-Conf pode ser usada para configurar a taxa de transmissão mínima e máxima de uma fila em um *switch OpenFlow*. De acordo com a especificação *OpenFlow* 1.2, um controlador *OpenFlow* também pode ler essas taxas de um *switch*.

As mais recentes adições relacionadas com QoS no *OpenFlow* são as tabelas *meter* e *meter bands*, as quais podem ser usadas para limitar a taxa de transmissão de uma porta de saída.

Em termos de monitoramento, um controlador *OpenFlow* pode consultar um *switch* para obter estatísticas em diferentes níveis de agregação, por tabela, fluxo, porta e fila. Em síntese, o *OpenFlow* fornece um extensivo suporte para configurar e monitorar QoS e facilitar o controle de situações complexas, introduzidas pelo roteamento distribuído e mecanismos de engenharia de tráfego.

### 4.3.3 Mensuração das Métricas de QoS com o Protocolo OpenFlow

Esta seção apresenta como a mensuração das métricas de QoS (*throughput*, atraso, *jitter* e perda de pacotes) podem ser obtidas, usando mensagens do protocolo *OpenFlow* e pacotes de sonda (*probe*).

#### 4.3.3.1 Throughput

O protocolo *OpenFlow* dispõe de algumas mensagens de estatísticas para que o controlador consulte e obtenha informações sobre o estado de execução do *switch*. Exemplos incluem estatísticas de fluxo, estatísticas de contadores, estatísticas de fila, estatísticas agregadas, estatísticas de tabelas e estatísticas de portas. Contudo o controlador precisa dar sentido aos valores das estatísticas brutas (ou seja, contadores de bytes na maioria dos casos), levando em consideração o *timestamping* da coleção de estatísticas; a latência do plano de controle e o intervalo da coleta das estatísticas.

Em Santos, Ribeiro e Cesar (2015) o *throughput* por porta é estimado com o uso do par de mensagens, *Request* e *Reply*. Uma aplicação de monitoramento, com o suporte do módulo *Gerenciador de Estatísticas*, coleta as estatísticas de cada porta. O *Gerenciador de*

*Estatísticas* envia periodicamente um OFPT\_MULTIPART\_REQUEST com o tipo OFPMP\_PORT\_STATS para solicitar o volume de transmissão e o volume de pacotes descartados. O *switch* responde com um OFPT\_MULTIPART\_REPLY, contendo *lx\_bytes* e *lx\_dropped* (1ª. coluna do Quadro 5). Os dados transmitidos são comparados com a capacidade do *link* pelo módulo *Gerenciador da Topologia*, sendo possível assim, descrever a taxa de utilização do *link*. A Equação 5 calcula o volume de dados transmitidos por porta, em um intervalo de *pooling* [i, i-1].

$$TxRate(i) = \frac{tx\_bytes - tx\_bytes(i-1)}{Time(i) - Time(i-1)} \quad (5)$$

De forma similar, em Vejlø e Andersen (2014) o *throughput* atual de uma porta é estimado pela Equação 6, usando os contadores estatísticos da 1ª. Coluna do Quadro 5.

$$T(t)_{Port, Instantaneous} = \frac{T_x(t)[Byte] - T_x(t-1)[Byte]}{Time(t)[s] - Time(t-1)[s]} \quad (6)$$

Onde  $T_x(t)$  é o último número total de bytes transmitidos por uma porta no instante  $t$  e  $T_x(t-1)$  é a amostra anterior.  $Time(t)$  é o momento em que o valor de  $T_x(t)$  foi recebido de um *switch*, e igualmente  $Time(t-1)$  é o tempo em que  $T_x(t-1)$  foi recebido.

Utilizando as mensagens descritas na seção 3.4.4, em Adriche, Doerr, Kuipers (2014), para determinar o *throughput* para cada fluxo, um módulo de monitoramento consulta regularmente os *switches* obtendo as estatísticas por fluxo. Em cada consulta, o módulo recebe a quantidade de bytes enviados e a duração de cada fluxo (2ª. coluna do Quadro 5), permitindo que seja calculado o *throughput* efetivo por fluxo.

Em Vejlø e Andersen (2014) o *throughput* médio, durante todo o tempo de vida de um fluxo pode ser calculado usando o contador estatístico de “duração” em segundos e “RxBytes” (2ª. Coluna do Quadro 5). O *throughput* médio por fluxo pode ser calculado como mostrado na Equação 7.

$$T_{Flow, Life-timeaverage} = \frac{Rx [Byte]}{D[s]} \quad (7)$$

Onde  $Rx$  é o número total de bytes recebidos para um fluxo específico, e  $D$  é o momento da entrada do fluxo no *switch*, ou seja, o

quanto de tempo o fluxo existiu na rede.

#### 4.3.3.2 Taxa de Perda de Pacotes

Para estimar a taxa de perda de pacotes por porta, Santos, Ribeiro e Cesar (2015) propuseram a Equação 8, a qual faz uso dos contadores estatísticos da 1ª coluna do Quadro 5.

$$LossRate(i) = \frac{tx\_dropped(i) - tx\_dropped(i - 1)}{Time(i) - Time(i - 1)} \quad (8)$$

Conforme visto na Equação 8 é simples obter a taxa de perda de pacotes por porta, mas, no *OpenFlow* não existe uma maneira óbvia de mapear essas informações para cada fluxo individual. De acordo com Vejlø e Andersen (2014) uma maneira é usar as informações do estado de um fluxo, por exemplo, quando as estatísticas são pesquisadas para um fluxo, as estatísticas serão processadas de acordo com o estado real do fluxo. O estado do fluxo é, neste contexto, definido a partir da ação do fluxo. Se estiver configurada para “*drop*”, o fluxo está em estado de “*drop*” (fluxo descartado da rede), caso contrário, é determinado como em estado “ativo”.

Para Vejlø e Andersen (2014), uma resposta estatística para um fluxo contém valores absolutos, isto é, quantidade de bytes e pacotes que o *switch* recebeu para um fluxo específico. Esses valores absolutos podem ser convertidos em valores incrementais, levando o valor mais recente e subtraindo o valor anterior, como mostrado na Equação 9.

$$\hat{R}x(t)_{total,drop} = \hat{R}x(t - 1)_{total,drop} + \overbrace{Rx(t)_{total} - Rx(t - 1)_{total}}^{Incremental\ packet\ drops} \quad (9)$$

Sempre que um novo valor de descarte incremental é calculado, ele é adicionado aos valores anteriores, pois as atualizações são incrementais. Os valores incrementais combinados com a informação do estado de um fluxo (isto é, se o fluxo está em estado de descarte), podem ser usados para estimar a taxa de perda de pacotes como mostrado na Equação 10 (VEJLØ; ANDERSEN, 2014).

$$\hat{P}(t)_{total} = \frac{\hat{R}x(t)_{total,drop}}{Rx(t)_{total}} \quad (10)$$

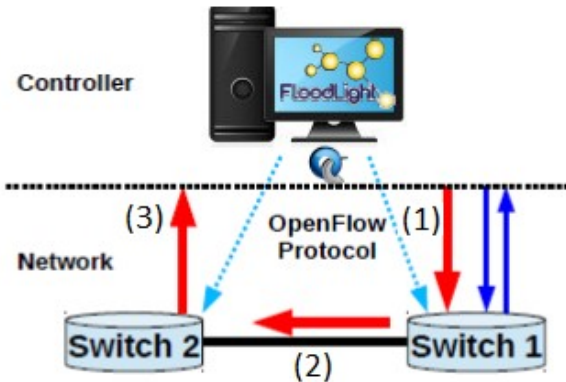
Onde  $\hat{P}(t)_{total}$  é a taxa de perda de pacote estimada,  $\hat{R}x(t)_{total,drop}$  é o número estimado de pacotes descartados e  $Rx(t)_{total}$

é o número total de pacotes recebidos. Para a Equação 9  $\hat{R}x(t - 1)_{total,drop}$  é a estimativa antiga dos pacotes descartados;  $Rx(t)_{total}$  é o número absoluto mais recente de pacotes recebidos e o  $Rx(t1)_{total}$  é o número absoluto de pacotes recebidos.

#### 4.3.3.3 Latência entre o Controlador e Switch

O protocolo *OpenFlow* não fornece medidas de atraso, contudo o atraso médio entre o controlador e os *switches* pode ser calculado facilmente. O controlador envia solicitações regularmente e recebe respostas, e esse intervalo de tempo é uma boa medida de atraso entre o controlador e os *switches*. O problema consiste em avaliar o atraso entre os *switches*. Em Phemius e Bouet (2013) é proposta a medição da latência (Figura 31), usando apenas as mensagens (*statistics request*, *statistics reply*, *packet out* e *packet in*) do protocolo *OpenFlow* versão 1.3 e a inserção de pacotes sondas (probe) na rede. Essa forma de estimar a latência (com pequenas alterações) é avaliada em Santos, Ribeiro e Cesar (2015) e Adrichem, Doerr e Kuipers (2014).

Figura 31 - Processo de monitoração da latência



Fonte: Adaptado de Phemius e Bouet (2013)

O processo de monitoração da latência apresentado na Figura 31 será descrito de acordo com Santos, Ribeiro e Cesar (2015). Inicialmente, no passo (1) o controlador cria um pacote especial com um *Ethertype* desconhecido e o envia ao *switch1* (S1). Este pacote de sonda (probe) é enviado através de uma mensagem *OFPT\_PACKET\_OUT* com um comando para que o S1 o encaminhe ao *switch2* (S2) no *link* a

ser monitorado. No passo (2), o S1 envia o pacote para S2. Uma vez que não há nenhuma regra compatível com este pacote, ao chegar em S2, o pacote desconhecido é encaminhado ao controlador, no passo (3) via mensagem OFPT\_PACKET\_IN. Quando o controlador recebe a mensagem com o pacote construído por ele, e considerando que o atraso entre o controlador e os *switches* é conhecido, é possível estimar a latência. Para isso considera-se o tempo total do percurso do pacote subtraído da diferença do atraso entre os *switches*, conforme Equação 11 (PHEMIUS; BOUET, 2013).

$$Latency_{(s1,s2)} = T_{total} - \frac{Ts_1}{2} - \frac{Ts_2}{2} - C \quad (11)$$

Onde, *S1* e *S2* representam os *switches* no caminho monitorado; *Total* é o tempo total para o cálculo da latência; *Ts1* é o tempo de ida e volta (RTT - *Round Trip Time*) entre o controlador e *s1*; *Ts2* é o RTT entre o controlador e *s2* e, *C* é a calibração do controlador.

Para que o controlador identifique a seção de rede de cada pacote retornado da etapa 3, as informações de identificação devem ser inseridas no pacote de sonda, incluindo o número de identificação do *switch* que enviou o pacote e o tempo que o pacote foi enviado (SANTOS; RIBEIRO, CESAR, 2015).

#### 4.3.3.4 Jitter

Em Santos, Ribeiro e Cesar (2015), o cálculo do *jitter* é obtido com base na RFC 3550 (SCHULZRINNE et al., 2003) que considera o acúmulo de medidas anteriores do atraso. Como já se tem a Equação 11 usada para calcular o atraso, é necessário armazenar o atraso anterior e atual para calcular o *jitter* em um intervalo de *pooling* [*i*, *i-l*]. A Equação 12 calcula o *jitter* atualizado, considerando o instante *i* e o instante anterior (*i-l*). O atraso nesse intervalo é *D(i-l, i)*.

$$J_{(i)} = J(i - 1) + (|D(i - 1, i)| - J(i - 1))/16 \quad (12)$$

#### 4.3.4 Funções de Custo dos Links entre os Switches OpenFlow

Em muitas circunstâncias é necessário alternar o caminho dos fluxos com base na engenharia de tráfego dos *links*. Desta forma serão apresentadas as funções de custos usadas numa aplicação de gerenciamento de rede SDN para a troca de caminhos de acordo com os

requisitos do fluxo (SANTOS; RIBEIRO, CESAR, 2015).

Em Santos, Ribeiro e Cesar (2015), cada função de custo tem três elementos: identificação do link; o valor do custo atual (o custo do caminho total calculado a partir da fonte até o link atual); estatísticas atuais, usadas para calcular o custo.

A seguir são apresentadas as funções de custos descritas na íntegra de acordo com o proposto por Santos, Ribeiro e Cesar (2015). Para definir as funções de custo os autores assumiram um *link* atual  $j$  em um instante atual  $t$ .

#### 4.3.4.1 Função de Custo da Utilização do Link

Após o cálculo da taxa de transmissão (Equação 6) é possível estimar a função de custo da utilização do *link*. Assim, considera-se que se um link  $j$  é amplamente utilizado, os bytes transmitidos na capacidade do *link* aumentam o custo, Equação 13.

$$LinkCost_{(j)} = \frac{1}{(LinkCapacity_{(j)} - TxRate_j)} \quad (13)$$

#### 4.3.4.2 Função de Custo da Taxa de Perda de Pacotes

Embora seja complexo, mas, é possível obter o cálculo do custo da perda ao longo do caminho, uma vez que outros cálculos sejam realizados anteriormente. A partir do cálculo da taxa de perda (Equação 8) e a taxa de utilização (Equação 6), calcula-se o percentual de perdas do *link*  $j$  pela Equação 14.

$$Loss_{(j)} = \frac{Transmitted_{source} - Received_{dest} + Dropped_{dest}}{Transmitted_{source}} \quad (14)$$

Está sendo considerado que os  $(1 - Lost(j))$  pacotes atravessam o *link*  $j$  e a perda total em um caminho com  $n$  *links* é definida pela Equação 15.

$$Loss_{(1,2,\dots,n)} = 1 - \prod_{j=1}^n (1 - Loss(j)) \quad (15)$$

A Equação 15 deve ser transformada de forma que  $LossCost(j)$

seja uma função de  $Loss(j)$  para ser usado pelo algoritmo de roteamento. É aplicado um logaritmo para alterar o produto para a soma do log (Equação 16).

$$\begin{aligned} \log(1 - Loss_{(1,2,\dots,n)}) &= \log\left(\prod_{i=1}^n (1 - Loss_{(j)})\right), \text{ then} \\ -\log(1 - Loss_{(1,2,\dots,n)}) &= \sum_{j=1}^n [-\log(1 - Loss_{(j)})] \quad (16) \end{aligned}$$

Quanto maior o  $Loss(1, 2, \dots, n)$ , maior se torna o lado direito; então, o custo de cada link é definido como a parte interna da soma, sendo calculado pela Equação 17.

Então, podemos definir o custo de cada link como parte interna da soma:

$$LossCost_{(j)} = -\log(1 - Loss_{(j)}) \quad (17)$$

Finalmente o custo acumulado da taxa de perda de pacotes pode ser calculado pela Equação 18

$$\sum_{j=1}^n LossCost_{(j)} \quad (18)$$

#### 4.3.4.3 Função de Custo do Atraso

O atraso de cada *link* é determinado pela Equação 11 e, como valor positivo, pode ser diretamente associado ao custo, sendo que quanto maior o atraso, maior é o custo. Portanto, na tentativa de obtenção do menor atraso, o custo do link  $j$  para o atraso é calculado pela Equação 19.

$$DelayCost_{(j)} = D_j(i - 1, i) \quad (19)$$

O atraso esperado de um caminho completo pode ser dado pela soma dos atrasos de cada *link*, que pode ser deduzido diretamente.

#### 4.3.4.4 Função de Custo do Jitter

O *Jitter* total de um caminho não pode ser definido exatamente

como a soma do *jitter* de cada *link*, porque assim pressupõe-se que cada *link* tem uma variação de atraso para o mesmo sinal, ou seja, quando um atraso aumenta, cada outro atraso aumenta também. Isto seria aplicável apenas para um pior caso. No entanto, a soma dos valores do *jitter* é uma boa abordagem para evitar caminhos com volumes potencialmente elevados de *jitter*. Devido à simplicidade do cálculo e bons resultados, Santos, Ribeiro e Cesar assumem que o custo do parâmetro *Jitter* é como calculado pela Equação 12, sendo o custo total do *jitter* descrito pela Equação 20.

$$JitterCost_{(j)} = J_j(i) \quad (20)$$

#### 4.4 CONSIDERAÇÕES DO CAPÍTULO

Conforme relatado no capítulo 3 infere-se que (i) o uso dos métodos de avaliação da QoE orientados ao contexto contribui para uma fina granularidade da estimativa da QoE; (ii) o uso de ontologias para a representação do conhecimento, combinado com regras de inferência e componentes autonômicos, permite a detecção da degradação da QoE. Contudo é preciso considerar o modelo de comunicação de dados para a restaurar e/ou evitar a degradação da QoE, bem como para disseminar o conhecimento.

Para tentar contemplar ao máximo a UX no uso de um serviço é necessário prover mecanismos na rede de comunicação para que os serviços sejam oferecidos cientes da UX, usando uma abordagem do usuário para a rede e não mais da rede para o usuário. Com base nisso, evidenciou-se a SDN como uma abordagem flexível para gestão dos recursos da rede. A flexibilidade da SDN permite que diversas aplicações de rede sejam criadas para atender as demandas das diferentes QoE.

Para contemplar o segundo, terceiro e quarto objetivos, no próximo capítulo é proposto o modelo de gerenciamento da QoE projetado para prover serviços para a Internet do Futuro, utilizando o paradigma de SDN.



## 5 MODELO DE GERENCIAMENTO DA QUALIDADE DE EXPERIÊNCIA DO USUÁRIO

As propostas que envolvem a experiência do usuário devem ir além da tecnologia e conhecimento do perfil, pois a UX deve ser tratada de forma mais ampla, incluindo aspectos cognitivos, estéticos e necessidades hedônicas. Essa abordagem só é possível se amparada por uma abordagem interdisciplinar (POMBO, 2005, 2008; SOMMERMAN, 2006). Nesta perspectiva o modelo de gerenciamento da QoE foi proposto, considerando os aspectos do conteúdo sintático, do conteúdo semântico e da qualidade da informação de contexto.

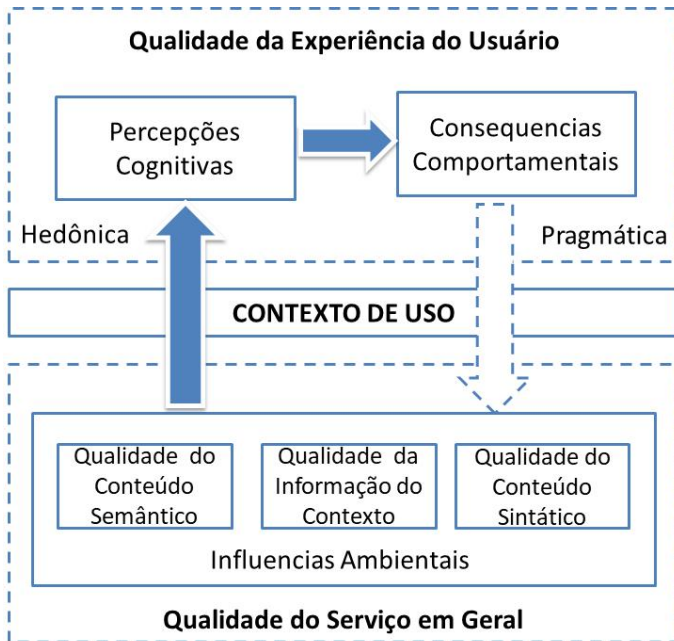
O modelo é composto por (i) uma taxonomia com as dimensões de influência na QoE; (ii) um modelo ontológico, como forma de representação do conhecimento e (iii) uma arquitetura para a provisão e entrega de serviços, utilizando o paradigma de SDN.

### 5.1 ABSTRAÇÃO DO MODELO PROPOSTO

De acordo com as teorias da Psicologia (MEHRABIAN e RUSSELL, 1974) os estímulos ambientais influenciam fortemente as percepções cognitivas, e por sua vez, moldam as intenções comportamentais de uma pessoa. E na interação com o ambiente, diante de um contexto, quando o usuário encontra “algo” que desperta interesse mental, os sensores (fisiológicos, auditivos, visuais) são acionados manifestando intenções comportamentais. Com base nos pressupostos de Mehrabian e Russell (1974) e no modelo de relação causa e efeito proposto por Wu et al. (2009) idealizou-se o modelo abstrato de gerenciamento da QoE (Figura 32).

Na Figura 32, diante de um contexto, tem-se a qualidade do conteúdo semântico, a qualidade da informação do contexto e a qualidade do conteúdo sintático. O conteúdo semântico está relacionado com o valor que o emissor e o receptor humano atribuem à semântica do conteúdo num contexto. O conteúdo sintático está relacionado com a qualidade do serviço (QoS), ou seja, refere-se à capacidade do sistema de comunicação prover uma infraestrutura física suficiente para transportar o conteúdo (texto, áudio, vídeo) com qualidade. E a qualidade da informação de contexto (QoC) refere-se a qualidade de qualquer informação usada como informação do contexto. Todos esses elementos devem ser criteriosamente controlados para que ao consumir um serviço, as necessidades hedônicas e pragmáticas dos usuários possam ser contempladas.

Figura 32 - Abstração do modelo de gerenciamento da QoE



Na Figura 32, a seta tracejada indica que existe um circuito de retroalimentação justificado pela sensibilidade ao contexto. A sensibilidade ao contexto permite que o serviço seja provido com informações atuais, precisas, válidas, completas e significantes. A sensibilidade ao contexto permite ainda, que os recursos de rede sejam ajustados para evitar degradação da QoE.

Os modelos de provisão de serviços cientes de contexto devem ser dinâmicos para garantir que UX será considerada na provisão de um serviço, pois a medida que o usuário vai interagindo com o ambiente e/ou consumindo o serviço, a sua experiência vai sendo aprimorada e os comportamentos, motivados pelas intenções, podem sofrer alterações. Desta forma, o modelo provê a avaliação do usuário para cada sessão. O *feedback* do usuário, juntamente com informações contextuais (perfil do usuário, políticas de QoS/QoC, estado dos recursos de provisão e entrega dos serviços; informações do serviço e do conteúdo) permitem a oferta do serviço de acordo com UX.

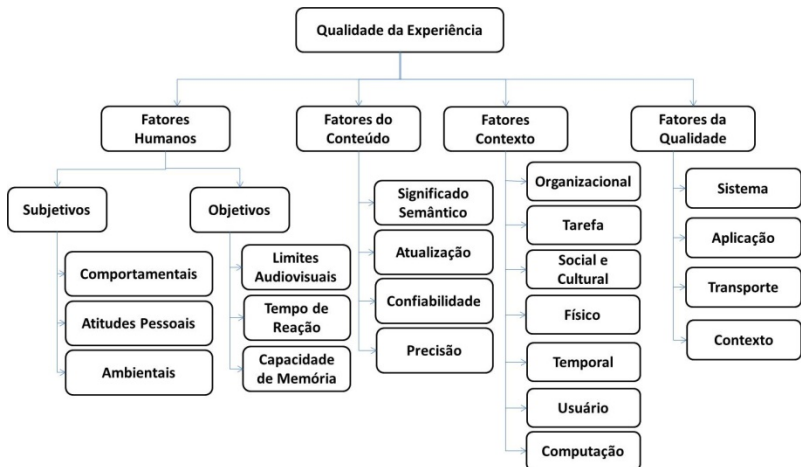
A concretização do modelo é composta por uma taxonomia com

todos os fatores de influência na QoE; um modelo ontológico para a representação do conhecimento da QoE e uma arquitetura para a provisão dos serviços orientada as necessidades dos usuários.

## 5.2 TAXONOMIA DAS DIMENSÕES DE INFLUÊNCIA NA QUALIDADE DE EXPERIÊNCIA

Ao analisar os trabalhos apresentados ao longo do terceiro capítulo observou-se que a proposição dos modelos existentes tem sido viabilizada, mesmo que de forma implícita, pela aplicação de pelo menos duas das seguintes bases teóricas: Psicologia, Sociologia, Ciência da Cognição, Design de Interface, Engenharia do Conhecimento, Computação Sensível ao Contexto e TIC. Nessa perspectiva e com base na análise dos conceitos e propostas de QoE apresentadas, foi possível gerar a taxonomia das dimensões de influência na QoE, conforme ilustrado na Figura 33.

Figura 33 - Taxonomia das dimensões de influência na QoE



### 5.2.1 Fatores Humanos

Os fatores humanos estão relacionados a diversos aspectos subjetivos e objetivos que influenciam o nível de satisfação do usuário ao consumir uma serviço ou usar uma aplicação.

### 5.2.1.1 Fatores Humanos Subjetivos

Os fatores humanos subjetivos são de natureza qualitativos. Estão relacionados com o comportamento, atitudes e motivações do usuário ao consumir um serviço ou utilizar uma aplicação.

Os fatores de atitude pessoal, incluem, facilidade de uso, alegria de uso, utilidade, conforto, complexidade, aborrecimento, tédio.

Os fatores ambientais referem-se a fatores externos que podem afetar o comportamento de um indivíduo. Existem ambientes sociais e físicos. Os sociais incluem a família, amigos, colegas, organização, pressões sociais, questões legais e fatores demográficos. O físico refere-se ao espaço físico e condições ambientais do local onde se encontra o usuário ao consumir um recurso (serviço, aplicativo), tais como, o tamanho da sala, a temperatura ou a disponibilidade de certas facilidades para o conforto e bem estar do usuário. Alguns atributos do perfil demográfico do usuário, como sexo e idade, por exemplo, podem ser influenciados em algum contexto específico (contexto de localização, contexto social etc). Assim como o atributo localização pode influenciar no contexto físico, por exemplo. Existem ainda os fatores relacionados ao controle comportamental percebido, que incluem, imagem, marca, *help desk*, facilidade de uso da tecnologia ou recurso.

Para a captura e análise de cada aspecto num ambiente real, pode-se fazer uso de teorias da Psicologia, IHC, Design de Interface. A escolha de qual teoria adotar depende da natureza do serviço, do ambiente, do contexto e que variáveis são importantes considerar no estudo. Essas teorias têm sido usadas pelos trabalhos descritos em Wu et al. (2009) e Laghari et al. (2011).

A aplicação de modelos psicológicos para compreender as intenções e comportamentos humanos fornece uma avaliação subjetiva global das necessidades dos utilizadores frente a um serviço correspondente. Vários modelos evoluíram e são usados para esses fins.

Um modelo amplamente reconhecido é o Modelo de Aceitação de Tecnologia (TAM), de Davis (1989), que é derivado da Teoria da Ação Racional (TRA), de Fishbein e Ajzen (1975). O modelo TAM é aplicado em muitos cenários para prever as intenções humanas e uso do sistema. Pelo fato desses modelos lidarem com comportamentos simples, onde o seu efetivo desempenho exige apenas a formação da intenção, surgiu a Teoria do Comportamento Planejado (TPB), de Ajzen (1991), acrescentando na TRA um aspecto chamado de Controle Comportamental Percebido (PBC). O PBC preocupa-se em saber se a pessoa se sente no controle da ação em questão. Há também uma versão

estendida do TPB, chamado decompositor TPB (DTPB), de Taylor e Todd (1995), que inclui subclassificações pormenorizadas dos parâmetros.

Para abstração dos fatores humanos subjetivos, sugere-se o uso do modelo DTPB, pelo fato do mesmo ter melhor poder de predição, além de fornecer uma maneira abrangente para entender como a atitude, normas subjetivas e controle comportamental percebido podem influenciar as intenções do indivíduo, tendo em conta fatores de atitudes pessoais e fatores de controle comportamental percebido.

### 5.2.1.2 Fatores Humanos Objetivos

Os fatores humanos objetivos são de natureza quantitativa. São usados para predizer o desempenho da tarefa humana, tais como capacidade de memória, limites humanos audiovisuais, tempo de reação da resposta do sistema. Eles podem ser obtidos por pesquisas controladas ou não controladas, usando métodos invasivos ou não invasivos. São apoiados pelas bases conceituais da Ciência Cognitiva, Psicologia Cognitiva e Modelos Mentais. Os autores Laghari et al. (2010) e Wu et al. (2009) têm essas disciplinas como bases teóricas para fundamentar suas propostas.

A Ciência Cognitiva e os Modelos Mentais têm sido usados para entender o desempenho da tarefa humana, fornecendo informações quantitativas sobre o desempenho humano (LI-YUAN; ZHOU; SONG, 2006). Na Interação Humano Computador (IHC) e Design de Interface, alguns modelos cognitivos de desempenho humano também são usados para capturar o desempenho da tarefa humana (BONNIE; KIERAS, 1996). Qualquer um desses modelos pode ser usado para obter o conhecimento dos fatores humanos objetivos.

### 5.2.2 Fatores de Conteúdo

De acordo com a mídia do conhecimento Perassi, Pavanati e Neves Júnior (2010) o processo de comunicação deve contemplar tanto o conteúdo semântico (ponto de vista do valor do conteúdo para o agente humano) quanto o conteúdo sintático (ponto de vista da tecnologia de transporte de dados). Esses dois objetos de estudo são estudados em diferentes dimensões da Figura 33. Ou seja, a comunicação sintática é estudada na dimensão “Qualidade do Serviço”, sendo considerados todos os parâmetros de QoS no transporte de dados. E a comunicação semântica é estudada na dimensão “Qualidade do

Conteúdo” sob a visão da semiótica. Os fatores desta dimensão são dependentes da representação, do valor, do significado da informação para o utilizador. Para Costa, Bertolato e Perassi (2011), a semiótica é uma das áreas que trata do significado das palavras, do valor que o receptor atribui à mensagem de acordo com a sua semântica. Esse valor pode ser influenciado pelo contexto e pela própria subjetividade do ser humano.

O significado que o usuário atribui a um conceito pode ser mapeado em ontologias. Com o uso das ontologias, é possível associar vários sinônimos a um mesmo conceito de acordo com a semântica do usuário, assim como, desambiguar palavras, associando o conceito a um contexto. O trabalho apresentado por D’Agostini e Fileto (2009) permite a desambiguação de palavras, como, por exemplo, na busca pelo termo “São Paulo”, o usuário pode fazer associação ao contexto de futebol, de cidade ou de santidade.

Além do significado semântico do conteúdo, a dimensão também considera os aspectos da qualidade da informação, de forma que seja possível gerenciar a atualidade, a confiabilidade e a precisão do conteúdo disponibilizado ao usuário.

### **5.2.3 Fatores de Contexto**

O contexto é qualquer informação que possa ser usada para caracterizar a situação de entidades como: pessoa, lugar ou objeto, que sejam consideradas relevantes para interação entre usuário e aplicação (DEY, 2000).

Os fatores relacionados com o contexto são usados para caracterizar a situação das entidades, prover dinamismo e adaptações no ambiente, melhorar a interação do usuário com o sistema. De acordo com Nazário e Dantas (2015), a partir das informações de contexto, os serviços podem ser adaptados adequando-se às características do ambiente atual e do perfil do usuário. Ou seja, é possível prover serviços mais otimizados e personalizados, para tentar contemplar a QoE dos usuários. Ainda segundo os autores, a utilização do contexto também pode minimizar o consumo de recursos como energia, processamento e comunicação, disponibilizando serviços mais precisos e dinâmicos.

Em Chen e KOTZ (2000) e Bellavista et al. (2012), os fatores relacionados com o contexto são classificados em computação, físico, tempo, usuário. No entanto, Moor et al. (2008) descreve outras informações contextuais, tais como contexto organizacional, tarefa, social e cultural do usuário.

De acordo com Chen e Kotz (2000), para capturar as informações do contexto temporal e localização, podem ser usados sensores do próprio equipamento, como GPS, 3G etc. Em Bellavista et al. (2012) é descrito como o contexto computacional pode ser captado e processado. Algumas informações do contexto social podem ser obtidas do perfil do usuário, outras informações, segundo Cardone et al. (2012), podem ser obtidas com o consento do usuário das suas redes de relacionamentos, do microfone para analisar a situação social (conversando, trânsito, escritório) ou do acelerômetro para discriminar atividade física (correndo, caminhando, dirigindo). O contexto da tarefa pode ser obtido de acordo com o tipo de serviço que o usuário está solicitando ou pela atividade (tarefa) que o usuário está realizando no sistema. O contexto organizacional e outras informações podem ser obtidos perguntando diretamente ao usuário.

#### **5.2.4 Fatores da Qualidade do Serviço**

Esta dimensão está relacionada com os parâmetros do sistema (e.g. tempo de resposta das operações internas realizadas pelo sistema); aspectos da qualidade da aplicação (e.g., codec, resolução, *bitrate*); parâmetros do desempenho da rede (e.g., atraso, variação no atraso, descarte de pacotes, largura de banda) e parâmetros da QoC (e.g., precisão, atualidade, acurácia) dos dados sensoriados em ambientes ubíquos e pervasivos.

Uma vez que todos os requisitos de qualidade tenham sido e definidos e os parâmetros de QoC tenham sido analisados, o sistema de transporte recebe e transporta os dados num meio preparado para garantir que a informação será entregue ao usuário com níveis de QoE aceitáveis.

Os parâmetros de QoS de rede podem ser captados através de software de monitoração e controle de rede. Mapeando os parâmetros de QoS da rede com o nível de satisfação do usuário, cujo serviço tenha sido avaliado na escala MOS (ITU-T P.800.1, 2003), por exemplo, é possível, em tempo de execução, ajustar dinamicamente o nível de QoS na rede para não degradar a QoE do usuário. Essa flexibilidade é facilitada com uso da SDN (ONF, 2012), permitindo ajustes e alocação de recursos de forma dinâmica.

### 5.3 COMPETÊNCIA DE ESTUDO DAS DIMENSÕES DA QUALIDADE DE EXPERIÊNCIA

De acordo com ETSI (2010) e através das dimensões de QoE levantadas observou-se que, embora os fatores das dimensões possam estar relacionados, nem todas as dimensões se aplicam ao mesmo propósito e irão depender das funcionalidades e do tipo de serviço que se pretende avaliar. Assim, a gestão da QoE é de competência de diferentes entidades que cooperam para oferecer o serviço com qualidade.

A seguir são apresentados alguns exemplos com situações e fatores de interesse de avaliação, de acordo com os papéis assumidos pelas entidades.

- **Fatores humanos objetivos:** se o objetivo é avaliar a usabilidade da interface de um sistema, testes com usuários e/ou uso de ferramentas automatizadas devem ser realizados para evidenciar a eficiência, eficácia na execução de uma tarefa. Sendo importante avaliar o desempenho da memória humana, limites audiovisuais, tempo de reação, perfil do usuário (idade, conhecimento, nível cultural e intelectual, traços de personalidade). Esses fatores são de interesse de estudo do Designer de Interface.
- **Fatores da QoS da aplicação e QoS da rede:** se o objetivo é avaliar a qualidade do vídeo, fatores relacionados com o tamanho da tela, resolução, GoP, codec devem ser considerados, incluindo fatores humanos objetivos de limitações audiovisuais. Esses fatores são de interesse de estudo do provedor de conteúdo. Considerando que o vídeo será transmitido por uma rede de comunicação, fatores relacionados com a qualidade do desempenho da rede (atraso, variação do atraso, descarte, erros) devem ser gerenciados pelo provedor de serviço/contéudo e pelo provedor de rede.
- **Fatores da QoS do sistema/rede e fatores humanos objetivos:** se o objetivo é analisar o nível de satisfação do usuário em relação ao tempo de processamento/retorno das operações pelo sistema, devem ser observados os recursos de hardware e rede tanto do cliente quanto do servidor, estimando o tempo de resposta das transações no banco de dados, operações de leitura/escrita no sistema, execução das operações funcionais no sistema. Esses fatores são de



interesse de estudo do provedor de serviço e/ou analista de dados. Importante ressaltar que fatores humanos objetivos também devem ser considerados, tal como o tempo de reação da memória humana em relação aos estímulos provocados pelo sistema. E por último, os aspectos de QoS da rede de transporte devem ser gerenciados pelo provedor de rede.

- **Fatores de QoC dos dados:** em ambientes ubíquos e pervasivos a maioria dos dados são provenientes de sensores, logo, a qualidade dos dados das fontes provedoras do contexto devem ser avaliadas. Tomando como base um AAL que utiliza o serviço de telemonitoramento dos sinais vitais de um paciente, por exemplo, os dados sensorizados devem ser atuais, precisos e íntegros, desta forma os parâmetros de QoC avaliados devem permitir que o sistema detecte anomalias ou inconsistências nos sensores; ative sensores backup; descarte dados com QoC insuficientes; escolha o provedor de contexto adequado, entre outras ações. Esses fatores são de interesse de estudo do analista de dados.
- **Fatores de conteúdo:** se o objetivo é avaliar o julgamento do usuário em relação ao conteúdo disponibilizado no ambiente, devem ser considerados o valor e o significado semântico do usuário quanto ao conteúdo, incluindo o designer. A forma com que a informação é disposta no ambiente, o quanto ela é atual, precisa e confiável, são fatores que qualificam a informação. Todos esses aspectos são objetos de estudo do provedor de conteúdo e designer de interface.

#### 5.4 MODELO DE REPRESENTAÇÃO DO CONHECIMENTO DA QUALIDADE DE EXPERIÊNCIA

O modelo de representação do conhecimento, proposto através de uma ontologia, consiste na construção de uma KB com formalismo semântico para aprender a UX no uso do serviço e autonomamente detectar degradações na QoE para alertar o sistema de gestão da QoE.

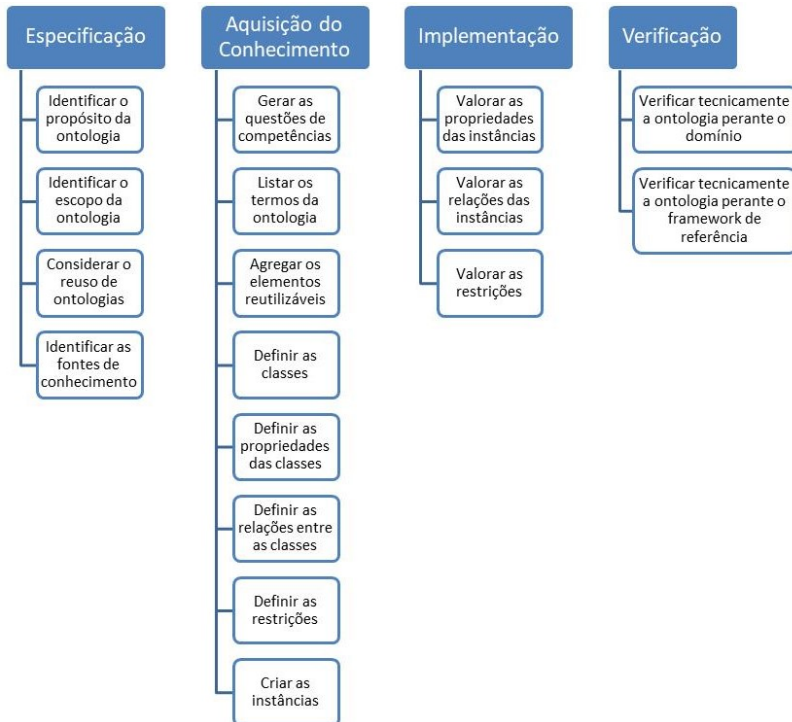
Considerando que os testes com usuários são difíceis de realizar, manter e tem um alto custo, inicialmente, uma KB pode ser construída com as fontes de conhecimentos já existentes, incluindo o conhecimento explícito dos profissionais das várias áreas de interesse; resultados de testes realizados e validados; pesquisa de campo; imersão na literatura. A partir desse conhecimento formalizado através de ontologias, novos

fatos podem ser derivados usando regras semânticas que incrementam o nível de expressividade do conhecimento. Além do reuso, as ontologias, juntamente com regras semânticas podem aprender a experiência do usuário no uso de um serviço e autonomamente detectar situações que podem degradar a qualidade percebida pelo usuário, dessa forma, auxiliando o sistema de gestão da QoE na restauração da qualidade.

#### 5.4.1 Metodologia para a Construção do Modelo de Conhecimento

A metodologia para o desenvolvimento da ontologia foi norteadada pelo processo metodológico proposto por Rautenberg e Todesco (2009), descrito na seção 2.4.5. O processo metodológico compreende as etapas de especificação, aquisição do conhecimento, implementação e verificação, conforme apresentado na Figura 34.

Figura 34 - Processo de desenvolvimento do modelo ontológico



Fonte: Rautenberg e Todesco (2009)

Aliado a esse processo metodológico foram usados os princípios dos métodos ágeis, tirando proveito do Scrum, que provê o desenvolvimento iterativo e incremental, cujas etapas de desenvolvimento são executadas em pequenas iterações com o objetivo de entregar partes do produto e não todo o produto. Assim, inicialmente um conjunto mínimo de questões de competências foi definido para obter uma versão mínima da ontologia, sendo que em cada iteração novas questões de competências foram inclusas.

A especificação da ontologia foi apoiada por duas ferramentas, OntoKEM (LECEGC-UFSC, 2009) e Protégé (BMIR, 2013). A OntoKEM, ferramenta desenvolvida no Laboratório de Engenharia do Conhecimento da UFSC, apoia o processo de construção e documentação de ontologias (TODESCO et al., 2009). E o Protégé (BMIR, 2013) é um *framework* de referência completo para a prototipação rápida de ontologias.

## 5.4.2 Modelo de Conhecimento para a Representação da QoE

### 5.4.2.1 Especificação da ontologia

#### 5.4.2.1.1 Propósito da Ontologia

O propósito da ontologia consiste em modelar os fatores de influência na Qualidade de Experiência, considerando características do usuário, do conteúdo, do contexto e da qualidade, com o objetivo de detectar degradações de parâmetros de qualidade, definidos em políticas.

Através da pesquisa realizada no capítulo 3 observou-se que existe um expressivo número de fatores, em diferentes níveis de abstração, que podem influenciar a percepção do usuário no uso de um serviço, porém, ainda não há um consenso de quais fatores devem ser avaliados, os termos utilizados não são formalizados, o conhecimento não está compartilhado e formalizado de maneira que possa auxiliar na gestão autônoma da QoE. Além disso, foi evidenciado que os fatores de influência podem escalar em função da demanda, da experiência do usuário, do contexto e das evoluções tecnológicas, portanto, a modelagem deve ser flexível e prever futuras implementações. Considerando ainda, que as dimensões e fatores de influencia na QoE irão depender do tipo de serviço, incluindo diferentes atribuições de acordo com os papéis assumidos pelas entidades, vale destacar que não há a pretensão de prover uma forma única de avaliação da QoE, mas

uma KB aberta que possa ser reaproveitada e incrementada pelas diferentes entidades de interesse no objeto de estudo.

#### 5.4.2.1.2 *Escopo da Ontologia*

A ontologia construída pode ser inserida em ambientes de prestação de serviços para que um mecanismo de inteligência possa aprender a experiência do usuário no uso de um serviço, alertar sobre degradações de valores de parâmetros que interferem na UX e prover uma KB para que outras aplicações possam otimizar a QoE, em caso de degradação da QoS. Além disso, a ontologia permite que uma KB, uma vez populada, forneça o conhecimento de todas as entidades modeladas (e.g. usuários, políticas, serviços). Portanto, o escopo da ontologia consiste em prover conhecimento da QoE no uso de um serviço e alertar sobre possíveis violações de parâmetros de QoS previstos na política de prestação de serviços.

#### 5.4.2.1.3 *Reuso de Ontologias*

Um ponto crucial ao desenvolver novas ontologias é considerar o reuso de ontologias correlacionadas para preservar os conceitos já estabelecidos e padronizados. Com essa intenção, além de alguns conceitos usados das ontologias apresentadas na sessão 3.3.4, outros conceitos foram reutilizados. Os vocabulários reutilizáveis são descritos a seguir.

FOAF (*Friend of a Friend*), Brickley e Miller (2000), provê um vocabulário descritivo expresso com RDF e OWL para descrever pessoas, suas atividades e suas relações com outras pessoas, redes e objetos. Isto é conseguido através da definição de relações entre as pessoas. Cada perfil tem um identificador único usado para definir essas relações.

*Basic Geo Vocabulary*, conhecido como WGS84 (BRICKLEY, 2003), provê um vocabulário RDF para a representação de informações espaciais sobre uma entidade, tais como latitude, longitude e outras informações relacionadas.

Ontologia de tempo, DAML *Time Ontology* (HOBBS e PAN, 2004) fornece um vocabulário para expressar aspectos temporais de uma entidade, apoiando a expressividade das relações topológicas entre instantes e intervalos de tempo, incluindo conceitos e propriedades para medidas de duração, representação de freqüência e unidades de tempo de calendário (data) e relógio (hora).

Ontologia de unidades de medidas, MUO (*Measurement Units Ontology*), criada por Polo e Berrueta (2008), é específica para a criação de unidades de medida e a representação de quantidades. A partir dos conceitos básicos presentes em MUO e das unidades de medidas listadas pelo Instituto *Regenstrief* no UCUM (*Unified Code for Units of Measure*), Schadow e McDonald (2009), uma grande quantidade de indivíduos foi criada, representando um conjunto completo com as unidades de medidas mais empregadas internacionalmente em ciência, engenharia e negócios.

#### 5.4.2.2 Aquisição do conhecimento

A primeira etapa da atividade de aquisição do conhecimento consiste na elaboração das questões de competências. As questões foram elaboradas com base em todo o referencial teórico e trabalhos relacionados descritos no capítulo 3 e também, com base nas dimensões da taxonomia proposta.

As questões de competências foram criadas para atender ao escopo da ontologia, ou seja, questões que uma entidade provedora de serviços precisa conhecer sobre o domínio do conhecimento e questões que a máquina deve obter para atuar com algum comportamento autônomo tanto para aprender a UX no uso do serviço quanto para alertar degradações na QoS que interferem na QoE.

Seguindo a abordagem ágil no desenvolvimento da ontologia, inicialmente um conjunto mínimo de QC (Questões de Competências) foi definido para obter uma versão mínima da ontologia. Posteriormente novas questões de competências foram definidas para a concretização das demais iterações e finalização da 1ª versão da ontologia. As QC foram elaboradas com base em todo o levantamento das propostas de QoE existentes e na taxonomia proposta.

- 1) Quais são os usuários (assinantes) de um determinado serviço?
- 2) Que tipo de assinatura possui um determinado usuário?
- 3) Quais políticas possui um determinado usuário?
- 4) Quais são as políticas existentes para um determinado serviço?
- 5) Quais as políticas de QoS para o parâmetro *latency*, considerando um tipo de serviço?
- 6) Quais as políticas de QoS para o parâmetro *jitter*, considerando um tipo de serviço?

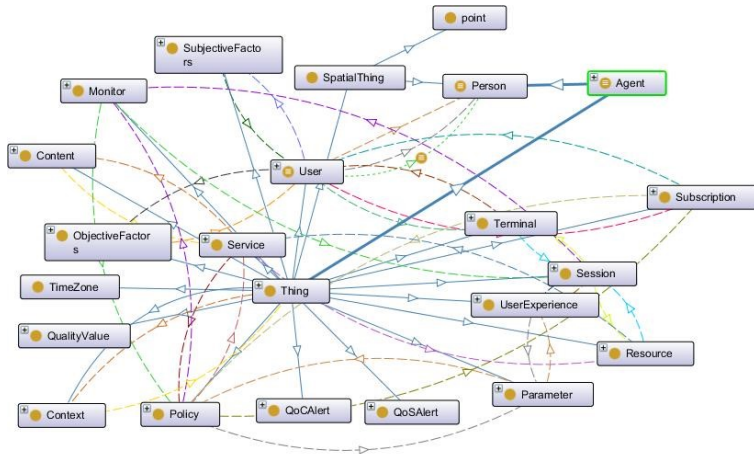
- 7) Quais as políticas de QoS para o parâmetro *packet loss*, considerando um tipo de serviço?
- 8) Quais as políticas de QoS para o parâmetro *throughput*, considerando um tipo de serviço?
- 9) Quais são os parâmetros de QoS de um determinado tipo de serviço?
- 10) Quais são os parâmetros de QoC de um determinado tipo de serviço?
- 11) Qual o conteúdo de um determinado serviço?
- 12) Qual o contexto do usuário na provisão de um determinado serviço?
- 13) Qual o contexto do usuário quando UX é avaliada na escala QoE-MOS [5-Excelente; 4 – Boa; 3 – Razoável; 2 – Ruim; 1 – Péssima] considerando um determinado serviço?
- 14) Qual o contexto da rede quando UX é avaliada na escala QoE-MOS [5-Excelente; 4 – Boa; 3 – Razoável; 2 – Ruim; 1 – Péssima] considerando um determinado serviço?
- 15) Qual o contexto do usuário e da rede quando UX é avaliada na escala QoE-MOS [5-Excelente; 4 – Boa; 3 – Razoável; 2 – Ruim; 1 – Péssima] considerando um determinado serviço?
- 16) Qual o contexto do usuário, do conteúdo e da rede quando UX é avaliada na escala QoE-MOS [5-Excelente; 4 – Boa; 3 – Razoável; 2 – Ruim; 1 – Péssima] considerando um determinado serviço?
- 17) Houve violação da política de QoS que interfere na UX no consumo de um tipo de serviço?
- 18) Houve violação da política de QoS do parâmetro *latency* para um determinado serviço?
- 19) Houve violação da política de QoS do parâmetro *jitter* para um determinado serviço?
- 20) Houve violação da política de QoS do parâmetro *throughput* para um determinado serviço?
- 21) Houve violação da política de QoS do parâmetro *packet loss*, para um determinado serviço?

A partir das questões de competências, dos trabalhos pesquisados e da taxonomia apresentada anteriormente, foram enumerados os termos da ontologia, preservando os termos comumente utilizados pelos especialistas de domínio, com base no referencial teórico pesquisado. Na sequência, foi verificado para cada termo, se o seu entendimento

remete a um conceito geral do domínio, gerando uma versão inicial das possíveis classes e subclasses. Posteriormente, seguindo a abordagem incremental e iterativa, novas classes da ontologia foram especificadas.

A Figura 35 apresenta as classes base da ontologia. A partir da taxonomia da Figura 33, outras classes foram especializadas a partir dos conceitos base, conforme sumarização do Quadro 6.

Figura 35 - Classes base da ontologia de avaliação da QoE



Quadro 6 - Conceitos da ontologia de avaliação da QoE

<b>CONCEITO</b>	<b>FUNÇÃO</b>
<i>Context</i>	Representa o contexto associado com as entidades, <i>UserContext</i> , <i>Organizational</i> , <i>Environment</i> , <i>Task</i> , <i>Location</i> , <i>Computation</i> , e <i>Temporal</i> . Essas entidades foram especializadas a partir do conceito contexto. Novas especializações podem ser criadas.
<i>User</i>	Representa o usuário do sistema.
<i>Subscription</i>	Representa as assinaturas dos serviços contratados pelo usuário.
<i>SubjectiveFactors</i>	Representam os fatores Objetivos do usuário. A partir do conceito <i>SubjectiveFactors</i> foram especializados os conceitos <i>Behavioural</i> , <i>Attitude</i> , <i>Motivational</i> .
<i>ObjectiveFactors</i>	Representam os fatores Objetivos do usuário. A partir do conceito <i>ObjectiveFactors</i> foram especializados os conceitos <i>AudioLimit</i> , <i>VisualLimit</i> , <i>ReactionTime</i> , <i>MemoryCapacity</i> .
<i>UserExperience</i>	Representa a experiência do usuário no uso de um

<b>CONCEITO</b>	<b>FUNÇÃO</b>
	serviço, ou seja o quanto o usuário está satisfeito com o serviço, levando em consideração o contexto. O usuário avalia o serviço ao término da seção usando a escala MOS [1..5].
<i>FOAF: Person</i>	Representa as informações sobre uma Pessoa conforme descrito na ontologia FOAF.
<i>WSG84:Point</i>	Representa informações de uma entidade do sistema, através dos conceitos <i>Latitude</i> , <i>Longitude</i> e <i>Altitude</i> , previstos no vocabulário WSG84.
<i>Time</i>	Representa informações temporais sobre uma entidade do sistema. Foi definida como uma subclasse de <i>TIME:TemporalEntity</i> .
<i>Parameter</i>	<p>Representam os parâmetros da qualidade do serviço, com as especializações <i>QoSSystem</i>, <i>QoSApplication</i>, <i>QoSNetwork</i>, <i>QoEMOS</i>, <i>QoCData</i>.</p> <p><i>QoSSystem</i> - representa os parâmetros relacionados ao tempo de resposta das operações internas realizadas pelo sistema.</p> <p><i>QoSApplication</i> - representa os parâmetros da aplicação combinados na sessão, contém outras informações <i>ScreenSize</i>, <i>Resolution</i>, <i>Codec</i>, <i>GoP</i> definidas através das propriedades de dados, no caso de aplicações multimídia.</p> <p><i>QoSNetwork</i> - representa as métricas objetivas dos parâmetros de desempenho, <i>throughput</i>, <i>latency</i>, <i>jitter</i>, <i>PacketLoss</i>, para qualquer serviço provido.</p> <p><i>QoEMOS</i> - representa a classe da Qualidade de Experiência, que mantém informações do MOS, calculado com base nos parâmetros de QoS monitorados, levando em consideração o contexto.</p> <p><i>QoCData</i> - representa as métricas associadas a qualidade dos dados. Tendo as seguintes especializações <i>Completeness</i>, <i>Coverage</i>, <i>Precision</i>, <i>Significance</i>, <i>UpToDateness</i>.</p>
<i>Measure</i>	Representa as entidades mensuráveis do sistema, sendo que toda entidade mensurável possui um valor e unidade de medida. Exemplo, o parâmetro de QoS <i>latency</i> possui um valor de 100, medido em milissegundos). <i>Measure</i> foi declarada como uma subclasse de <i>muo:QualityValue</i> .
<i>Session</i>	Representa a <i>Session</i> do serviço provido.
<i>Service</i>	Representa os serviços disponíveis. A <i>eHealth</i> é uma especialização de <i>Service</i> , que especifica os tipos de serviços disponíveis a partir do serviço <i>eHealth</i> , tais



<b>CONCEITO</b>	<b>FUNÇÃO</b>
	como telemonitoramento, teleconsulta etc.
<i>Resource</i>	Representa o recurso de rede que oferece o serviço ao usuário.
<i>Terminal</i>	Representa o terminal do usuário.
<i>Content</i>	Representa o tipo de <i>Content</i> usado durante a sessão do serviço.
<i>Policy</i>	Representa as políticas de QoS e QoC associadas por tipo de serviço e por usuário. A política de QoS, definida para cada parâmetro, é descrita com um valor máximo e mínimo aceitável. A partir de <i>Policy</i> foram definidas duas especializações para políticas <i>QoSPolicy</i> e <i>QoCPolicy</i> , representando respectivamente, as políticas de QoS e QoC de um serviço. Outras especializações foram criadas para acomodar as políticas em suas respectivas classes.
<i>Monitor</i>	Representa a entidade que monitora os parâmetros de qualidade mensuráveis, tais como os parâmetros de QoS associados com as políticas; os parâmetros de QoC e a avaliação da QoE/MOS na provisão do serviço. A partir de <i>Monitor</i> foram definidas duas especializações <i>QoSMonitor</i> e <i>QoCMonitor</i> representando respectivamente, as políticas de QoS e QoC de um serviço. Outras especializações foram criadas para associar os parâmetros monitorados em suas respectivas classes.
<i>CostPath</i>	Representa o custo de cada caminho existente da origem até o destino para a entrega dos dados.
<i>QoSAlert</i>	Representa os alertas de QoS gerados pelo motor de inferência ao detectar violações de políticas que irão degradar a QoE. Foram criadas especializações de alertas para cada parâmetro de QoS monitorado e que possui restrições de política de qualidade ( <i>LatencyViolation</i> , <i>JitterViolation</i> , <i>PacketLossViolation</i> , <i>ThroughputViolation</i> )
<i>QoCAlert</i>	Representa os alertas de QoC gerados pelo motor de inferência ao detectar violações de políticas que irão degradar a QoE. Foram criadas especializações de alertas para cada parâmetro de QoC monitorado e que possui restrições de política de qualidade ( <i>CompletenessViolation</i> ; <i>CoverageViolation</i> ; <i>PrecisionViolation</i> ; <i>SignificanceViolation</i> ; <i>Up-to-datenessViolation</i> )
<i>Inference</i>	Classe criada exclusivamente para armazenar os

<b>CONCEITO</b>	<b>FUNÇÃO</b>
	resultados das inferências de cada uma das QC. Para cada QC foi criada uma especialização de <i>Inference</i> .

Para cada classe foram definidas as propriedades (propriedade de dados) para identificar as características relevantes dos indivíduos (objetos) das classes. Considerando o reuso das ontologias (seção 5.4.2.3), poucas propriedades de objetos foram criadas, sendo definidas as seguintes:

- *name*: propriedade de dados com *range xsd:string* usada para identificar o nome de qualquer entidade do modelo.
- *description*: propriedade de dados com *range xsd:string* usada para descrever informações importantes sobre uma entidade de contexto.
- *idEntity*: propriedade de dados com *range xsd:int* usada para identificar uma entidade de contexto.
- *maxValue*: sub propriedade de *muo:qualityLiteralvalue* definida como uma propriedade funcional, usada para receber o valor máximo de um parâmetro associado a uma política.
- *minValue*: sub propriedade de *muo:qualityLiteralvalue* definida como uma propriedade funcional, usada para receber o valor mínimo de um parâmetro associado a uma política.
- *mos*: propriedade de dados com *range xsd:double* usada para representar o valor do MOS.
- *priorityQueue*: propriedade de dados com *range xsd:int* usada para representar a prioridade de um fluxo (serviço) associado a uma fila.
- *priorityInstant*: propriedade de dados com *range xsd:int* usada para representar a prioridade do fluxo no instante da medição.
- *routeStart*: propriedade de dados com *range xsd:string* usada para manter a informação da rota inicial do fluxo.
- *routeEnd*: propriedade de dados com *range xsd:string* usada para manter a informação da rota final do fluxo.
- *link*: propriedade de dados com *range xsd:short* usada para representar o *link*.
- *cost*: propriedade de dados com *range xsd:double* usada para representar o custo do caminho.
- *addressMAC*: propriedade de dados com *range xsd:string* usada para representar o endereço físico do recurso de rede.
- *addressIPv4*: propriedade de dados com *range xsd:string*

usada para representar o endereço lógico do recurso de rede.

- *transportPort*: propriedade de dados com *range xsd:int* usada para representar a porta lógica associada ao serviço.
- *GoP*: propriedade de dados com *range xsd:int* usada para representar o tamanho do grupo de imagem na compressão de um vídeo.
- *codec*: propriedade de dados com *range xsd:string* usada para representar o codec de um áudio ou vídeo.
- *resolution*: propriedade de dados com *range xsd:string* usada para representar a resolução de um vídeo.
- *screenSize*: propriedade de dados com *range xsd:string* usada para representar o tamanho de tela usado para exibir um vídeo.

Seguindo os passos metodológicos, a próxima etapa consistiu na definição das relações entre as classes, elencadas a seguir, com os respectivos relacionamentos de associação:

- *hasPersonID*: usada para relacionar indivíduos da classe *User* do sistema com indivíduos da classe *Person* da ontologia FOAF.
- *hasContext*: usada para relacionar outras classes ou indivíduos de outras classes com a classe *Context*.
- *hasContent*: usada para relacionar indivíduos da classe *Service* com indivíduos da classe *Content*.
- *hasSubscription*: usada para relacionar indivíduos da classe *User* com indivíduos da Classe *Subscription*.
- *hasPolicy*: usada para relacionar indivíduos da classe *Subscription* com indivíduos da Classe *Policy*.
- *hasService*: usada para relacionar indivíduos da classe *Policy* com indivíduos da Classe *Service*.
- *hasParameter*: usada para relacionar indivíduos das classes *Policy*, *UserExperience* com indivíduos das classe *Parameter*. Outras classes que possuem parâmetros mensuráveis podem se relacionar com a classe *Parameter* através da propriedade de objeto *hasParameter*.
- *hasUserExperience*: usada para relacionar indivíduos da classe *UserExperience* com indivíduos das classe *Session*.
- *hasObjectiveFactor*: usada para relacionar indivíduos da classe *User* com indivíduos das classes *ObjectiveFactors*.
- *hasSubjectiveFactor*: usada para relacionar indivíduos da classe *User* com indivíduos das classes *SubjectiveFactors*.

- *hasMeasure*: usada para relacionar indivíduos da classe *Parameter* com indivíduos da classe *Measure*. A *hasMeasure* foi definida como subpropriedade de *muo:qualityValue*.
- *hasMonitor*: usada para relacionar indivíduos das classes *Policy* e *Session* com indivíduos da classe *Monitor*.
- *hasSession*: usada para relacionar indivíduos da classe *Terminal* e *Resource* com indivíduos da classe *Session*.
- *hasTerminal*: usada para relacionar indivíduos da classe *User* com indivíduos da classe *Terminal*.
- *hasResource*: usada para relacionar indivíduos da classe *Service* com indivíduos da classe *Resource*.
- *isParameterOf*: relação inversa de *hasParameter*.
- *isMeasureOf*: relação inversa de *hasMeasure*.
- *isPolicyOf*: relação inversa de *hasPolyce*.
- *isServiceOf*: relação inversa de *hasService*.
- *isSubscriptionOf*: relação inversa da propriedade *hasSubscription*.
- *isContentOf*: relação inversa da propriedade *hasContent*.
- *isContextOf*: relação inversa da propriedade *hasContext*.
- *isMonitorOf*: relação inversa da propriedade *hasMonitor*.
- *isSessionOf*: relação inversa da propriedade *hasSession*.
- *isTerminalOf*: relação inversa da propriedade *hasTerminal*.
- *isResourceOf*: usada para relacionar indivíduos da classe *Service* com indivíduos da classe *Resource*.
- *isObjectiveFactor*: relação inversa da propriedade *ObjectiveFactor*.
- *isSubjectiveFactor*: relação inversa da propriedade *SubjectiveFactor*.

Após a definição das propriedades de dados e propriedades de objetos foram levantadas algumas restrições. Foi restringido que nenhum parâmetro quantitativo monitorado pode ter valor negativo. Todos os parâmetros relacionados com a política de QoS possuem restrições de valor mínimo e máximo. Foi restringido ainda, a unidade de medida de cada parâmetro, hora e data associadas com a sessão e medições dos parâmetros.

Na sequência foram definidas as instâncias para todas as classes da ontologia base, exceto para as classes e subclasses de *QoAlert* e *QoSAlert*, pois estas foram usadas exclusivamente pelo motor de inferência para raciocinar a partir das instâncias existentes na KB.

### 5.4.2.3 Implementação

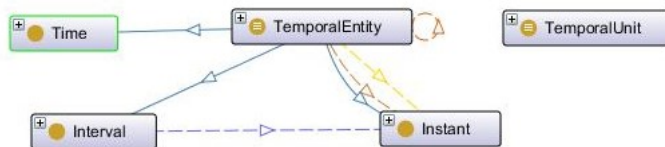
Na etapa de implementação foram valoradas as instâncias, criadas a partir das propriedades de dados definidas na seção anterior. Da mesma forma, foram valoradas as relações das instâncias, através das propriedade de objetos. E por último foram valoradas as restrições das instâncias presentes no domínio, considerando os valores possíveis para as suas propriedades de dados e para suas relações admitidas com as classes da ontologia.

Por uma questão de facilidade optou-se em agregar os recursos das ontologias existentes na etapa de implementação, por considerar que a aquisição do conhecimento sobre as mesmas foi realizada anteriormente e também porque a agregação, de fato, ocorreu nesta etapa.

Na versão atual da ontologia proposta, foram agregados recursos das quatro ontologias evidenciadas anteriormente, sendo a *OWL-Time* (HOBBS; PAN, 2004), *WGS84* (BRICKLEY, 2003), *FOAF* (BRICKLEY; MILLER, 2000) e *MUO* (POLO; BERRUETA, 2008).

Para relacionar o contexto temporal de qualquer entidade da ontologia proposta (e.g. parâmetro monitorado, tempo de início, finalização e duração de uma sessão) e poder usar as propriedades de objetos da ontologia *OWL-time*, foi necessário, antes, tornar a classe *Time* da ontologia proposta como subclasse da classe *TemporalEntity*, conforme mostra a Figura 36.

Figura 36 - Ligação do conceito Time da ontologia proposta com a ontologia DAML Time



A Figura 36 apresenta a relação hierárquica entre as classes (setas contínuas) e as relações entre as classes (tracejadas). A relação entre a superclasse *TemporalEntity* e a subclasse *Instant* expressa que toda entidade temporal possui instantes de início e de término, determinado pelas propriedades *hasBeginning* e *hasEnd*, respectivamente. Da mesma forma, a relação *inside* expressa que um dado instante *Instant*

pertence a um intervalo de tempo *Interval*.

Para relacionar os usuários da ontologia proposta com uma pessoa da ontologia FOAF, foi criada a propriedade de objeto *hasPersonID* e uma restrição com o quantificador existencial, relacionando a classe *User* com a classe *Person* da FOAF, conforme ilustra a Figura 37.

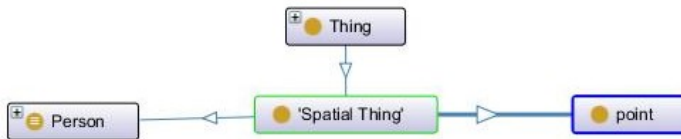
Figura 37 - Ligação da classe *User* com a classe *Person* da Ontologia FOAF



A Figura 37 apresenta as classes do vocabulário WGS84, *SpacialThing* e *Point*. A classe *Point*, possui propriedades para representar as coordenadas de latitude (*geo:lat*), longitude (*geo:long*) e altitude (*geo:alt*) de um objeto em relação ao ambiente.

Na Figura 38 é demonstrada a ligação da classe *Person* da FOAF, como uma subclasse de *SpacialThing*. A classe *SpacialThing* possui a propriedade de objeto *location* usada para relacionar instâncias de objetos *Point* com indivíduos de outras classes, tais como *User*, *Terminal*, *Resource*, da ontologia proposta. Desta forma é possível representar a localização espacial de uma pessoa ou dispositivo.

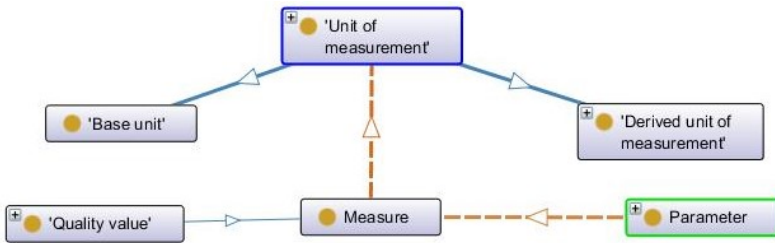
Figura 38 - Principais classes do vocabulário WGS84



E por último foi realizado o relacionamento da classe *Measure* com a classe *UnitOfMeasurement* da ontologia de modelagem de medidas, MUO, conforme descrito abaixo e apresentado na Figura 39.

Na MUO todo objeto possui uma quantidade física. Essa quantidade é expressa de maneira geral e específica. De maneira geral, através de tipos de quantidades físicas, como massa, peso, velocidade; e de maneira específica, através da medição dessa quantidade no objeto — a massa de uma pessoa, o peso de uma cadeira, a velocidade máxima de um *link*.

Figura 39 - Ligação da classe Measure com a classe UnitOfMeasurement da Ontologia MUO



A expressão geral de uma quantidade física é medida usando uma unidade, representada pela classe *UnitOfMeasurement* e as subclasses *BaseUnit* e *DerivedUnitOfMeasurement*, representam, respectivamente, os dois tipos de unidades de medida existentes, ou seja, unidades base e unidades derivadas.

A expressão específica de uma quantidade física é representada pela subclasse *QualityValue*, esta por sua vez, possui um valor específico de quantidade, relacionado através da propriedade de tipo de dados *qualityLiteralValue*, e uma unidade de medida, relacionado ao conceito *UnitOfMeasurement* através da propriedade de objeto *measuredIn*. A outra propriedade, *qualityValue*, declarada como inversamente funcional, é usada para que *QualityValue* se relacione com uma outra classe.

Na ontologia proposta foi criada a classe *Measure*, sendo definida como subclasse de *muo:QualityValue* e declarada como disjunta a todas as outras classes. Três propriedades estão sendo reutilizadas, *muo:measuredIn*; *muo:qualityLiteralValue* e *muo:qualityValue*.

A propriedade de objeto *measuredIn* é usada para indicar a unidade de medida do objeto gerenciado; a propriedade de dados *qualityLiteralValue* é usada para definir o valor do objeto e, por último, foi definida uma sub propriedade de dados de *QualityValue*, chamada de “*hasMeasure*” para que a classe *Parameter* possa se relacionar com a classe *Measure*.

Além disso, foram criadas duas restrições existências, para definir que *Measure* é medida utilizando uma unidade de medida de MUO (*muo:UnitOfMeasurement*), e que possui um valor numérico (*xsd:double*).

#### 5.4.2.4 Verificação

Essa etapa é composta por três atividades de verificação: - verificação da ontologia perante o domínio; verificação técnica da ontologia perante o *framework* e verificação da ontologia quanto ao uso.

Nesta etapa, inicialmente, foi feita a verificação da ontologia perante o domínio. Nesse momento foi averiguado se a ontologia modelada apresentava alguma inconsistência em relação ao entendimento aceito sobre o domínio nas fontes de conhecimento. Para esta atividade foi utilizado o raciocinador Pellet (SIRIN et al., 2007) e sempre que necessário, as fontes de dados (trabalhos correlatos, ontologias já existentes, conceitos sobre QoE) foram consultadas.

Na sequência foi realizada a verificação técnica da ontologia perante o *framework* de referência, ou seja, o propósito, escopo e as questões de competências foram revisitados para avaliar a consistência da ontologia frente os requisitos levantados. O propósito e escopo da ontologia foram mantidos, mas, em função das questões de competências foram realizados alguns ajustes na organização e relacionamento das classes.

Nessa etapa as questões de competências foram respondidas com o uso de consultas SPARQL (realizadas pelo usuário) e/ou regras SWRL (raciocinadas pela máquina). Essas duas abordagens são apresentadas nas próximas seções.

### 5.4.3 Consultas na Base de Conhecimento

Para a verificação do modelo ontológico as QC foram respondidas através de consultas SPARQL. Em cada consulta SPARQL é necessário o uso do cabeçalho dos vocabulários utilizados (conforme já apresentado no Quadro 2). Os Quadros 7 a 13 apresentam as estruturas das consultas SPARQL de algumas QC.

A consulta do Quadro 7 verifica se existem indivíduos na classe usuário que possuem assinatura (através da relação *hasSubscription*) e se a assinatura está associada com alguma política (através da relação *hasPolicy*), que por sua vez está associada ao serviço *telebiomedical* (através da relação *hasPolicy*). A consulta pelo nome do serviço é feita através da propriedade de dados *nameEntity*. E o resultado é mostrado em ordem alfabética pelas instâncias de usuários.

Vale destacar que todo serviço tem uma política de QoS associada. Nessa política cada parâmetro de QoS (*latency, jitter, packet loss, throughput*) é definido por um valor máximo e mínimo, logo, se



existir algum usuário que tem assinatura e política associada ao serviço de *Telebiomedical*, serão listados os usuário com as políticas associadas ao serviço.

Quadro 7 - Consulta SPARQL da QC 1

QC 1: Quais são os usuários de um determinado serviço?
<pre> SELECT ?User ?Subscription ?Policy ?Service   WHERE {     ?User ux:hasSubscription ?Subscription.     ?Subscription ux:hasPolicy ?Policy.     ?Policy ux:hasService ?Service.     ?Service ux:name "telemonitoring"^^xsd:string   } </pre>

A consulta do Quadro 8 procura por indivíduos na classe usuário com o nome “Alice da Silva Bob” e através da relação *hasSubscription* entre a classe usuário e a classe assinatura é verificado se Alice possui assinatura.

Quadro 8 - Consulta SPARQL da QC 2

QC 2: Que tipo de assinatura possui um determinado usuário?
<pre> SELECT ?User ?Subscription   WHERE {     ?User ux: nameEntity "Alice da Silva Bob"^^xsd:string.     ?User ux:hasSubscription ?Subscription   } </pre>

A estrutura do Quadro 9 busca por indivíduos na classe usuário, com o nome “Joao Costa e Silva”; verifica se João possui assinatura (através da relação *hasSubscription*) e se a assinatura está associada com alguma política (através da relação *hasPolicy*).

Quadro 9 - Consulta SPARQL da QC 3

QC 3: Quais políticas possui um determinado usuário?
<pre> SELECT ?Subscription ?Policy   WHERE {     ?User ux:nameEntity "Joao Costa e Silva"^^xsd:string.     ?User ux:hasSubscription ?Subscription.     ?Subscription ux:hasPolicy ?Policy   } </pre>

A consulta apresentada no Quadro 10 verifica na classe Política

se existem políticas relacionadas com o serviço, usando a relação *hasService*. Se na classe *eHealth* (sub classe de *Service*) existir um indivíduo com nome de *medical-images* serão retornadas todas as políticas associadas ao mesmo.

Quadro 10 - Consulta SPARQL da QC 4

QC 4: Quais são as políticas existentes para um determinado serviço?
<pre> SELECT ?Policy   WHERE {     ?Policy ux:hasService ?Service.     ?eHealth ux:nameEntity "medical-images"^^xsd:string.   } ORDER BY ?Policy </pre>

O Quadro 11 apresenta as consultas criadas para verificar quais são as políticas para um parâmetro de QoS (*Latency*, *Jitter*, *Packet Loss* e *Throughput*) considerando o serviço de *MedicalImages*. As consultas foram estruturadas representando, respectivamente, respostas para as QC 5, 6, 7 e 8. Inicialmente, através da relação *isParameterOf* é verificado se na classe *Parameter* existe algum indivíduo que está associado com alguma política de QoS. As consultas irão retornar os indivíduos da classe Parâmetro que possuem relações com os indivíduos da classe Política de QoS.

Quadro 11 - Consultas SPARQL das QC 5, 6, 7 e 8

QC 5 a QC 8: Quais as políticas de QoS para um parâmetro ( <i>Latency</i> , <i>Jitter</i> , <i>Packet Loss</i> e <i>Throughput</i> ) considerando um tipo de serviço?
<pre> SELECT ?Parameter ?QoSPolicy   WHERE {     ?Parameter ux:isParameterOf ?QoSPolicy.     ?Parameter rdf:type ux:Latency.     ?QoSPolicy ux:hasService ?MedicalImages.     ?MedicalImages ux:nameEntity "medical- images"^^xsd:string   } </pre>
<pre> SELECT ?Parameter ?QoSPolicy   WHERE {     ?Parameter ux:isParameterOf ?QoSPolicy.     ?Parameter rdf:type ux:Jitter.     ?QoSPolicy ux:hasService ?MedicalImages.     ?MedicalImages ux:nameEntity "medical- images"^^xsd:string   } </pre>

<pre> SELECT ?Parameter ?QoSPolicy   WHERE {     ?Parameter ux:isParameterOf ?QoSPolicy.     ?Parameter rdf:type ux:PacketLoss.     ?QoSPolicy ux:hasService ?MedicalImages.     ?MedicalImages ux:nameEntity "medical- images"^^xsd:string   } </pre>
<pre> SELECT ?Parameter ?QoSPolicy   WHERE {     ?Parameter ux:isParameterOf ?QoSPolicy.     ?Parameter rdf:type ux:Throughput.     ?QoSPolicy ux:hasService ?MedicalImages.     ?MedicalImages ux:nameEntity "medical- images"^^xsd:string   } </pre>

A consulta apresentada no Quadro 12 retorna todos os parâmetros de QoS para o serviço de *Teleconsultation*. A consulta será retornada se o serviço estiver relacionado com alguma política de QoS.

Quadro 12 - Consulta SPARQL da QC 9

<p>QC 9: Quais são os parâmetros de QoS de um determinado serviço?</p>
<pre> SELECT ?QoSPolicy   WHERE {     ?QoSPolicy ux:hasService ?Teleconsultation.     ?Teleconsultation ux:nameEntity "teleconsultation"^^xsd:string.   } </pre>

A consulta apresentada no Quadro 13 retorna todos os parâmetros de QoC para o serviço de *Teleenvironmental*. A consulta será retornada se o serviço estiver relacionado com alguma política de QoC.

Quadro 13 - Consulta SPARQL da QC 10

<p>QC 10: Quais são os parâmetros de QoC de um determinado serviço?</p>
<pre> SELECT ?QoCPolicy   WHERE {     ?QoCPolicy ux:hasService ?teleenvironmental.     ?Teleenvironmental ux:nameEntity "tele- environmental"^^xsd:string   } </pre>

#### 5.4.4 Regras de Raciocínio na Base de Conhecimento

Uma base de regras especificada em SWRL (HORROCKS et al., 2004) foi criada para responder cada uma das QC. As inferências foram realizadas com o motor semântico Pellet (SIRIN et al., 2007).

Muito embora, conforme apresentado anteriormente, as QC podem ser respondidas com consultas SPARQL, as mesmas, isoladamente, não são adequadas para prover características autonômicas ao mecanismo de inteligência. A diferença das regras semânticas, quando comparadas com as consultas SPARQL é justamente o fato delas contribuírem para que o motor semântico atue com características autonômicas.

Nos Quadros 14 a 27 são apresentas as regras SWRL. Para as regras que não exigem raciocínio autonômico do motor semântico, o resultado do processo de inferência é visualizado em classes específicas, criadas exclusivamente para este propósito. No entanto, os resultados dos alertas de QoC e QoS, são inferidos nas classes base da ontologia proposta (classes *QoCAAlert* e *QoSAlert* e respectivas subclasses).

Usando as propriedades de objetos, *hasSubscription* (relação entre as classes usuário e assinatura), *hasPolicy* (relação entre as classes assinatura e política) e *hasService* (relação entre as classes política e usuário) na regra do Quadro 14 é verificado se na classe usuário existe alguma instância que possui assinatura associada a um serviço. Como resultado são inferidas todas as instâncias existentes da classe usuário que satisfazem as condições específicas da regra. A saída da inferência é exibida na classe *QC-1-ServiceUser*.

Quadro 14 - Regra SWRL da QC 1

QC 1: Quais são os usuários de um determinado serviço?
$User(?usr) \wedge hasSubscription(?usr, ?sub) \wedge hasPolicy(?sub, ?pol) \wedge hasService(?pol, ?serv) \rightarrow ux:QC-1-ServiceUser(?usr)$

Através da propriedade inversa *isSubscriptionOf* e da propriedade de dados *nameEntity* a regra do Quadro 15 verifica se existe alguma assinatura associada ao usuário, cujo nome é “Joao Costa e Silva”. Como resultado são inferidas todas as instâncias de assinaturas contratadas pelo usuário informado. A saída da inferência é exibida na classe *QC-2-UserSubscription*.

## Quadro 15 - Regra SWRL da QC 2

QC 2: Que tipo de assinatura possui um determinado usuário?
Subscription(?sub) ^ isSubscriptionOf(?sub, ?usr) ^ nameEntity(?usr, "Joao Costa e Silva") -> ux:QC-2-UserSubscription(?sub)

A regra do Quadro 16 infere as políticas do usuário “Maria José Lima” através das propriedades de objetos *hasSubscription* e *hasPolicy* e, da propriedade de dados *nameEntity*. Como resultado são inferidas na classe *QC-3-UserPolicy* todas as instâncias de políticas contratadas pelo usuário em questão.

## Quadro 16 - Regra SWRL da QC 3

QC 3: Quais políticas possui um determinado usuário?
nameEntity(?usr, "Maria José Lima") ^ User(?usr) ^ hasSubscription(?usr, ?sub) ^ hasPolicy(?sub, ?pol) -> ux:QC-3-UserPolicy(?pol)

A regra do Quadro 17 infere as políticas do serviço através da propriedade de objeto *hasService* e da propriedade de dados *nameEntity*. São inferidas todas as instâncias da classe política associadas ao serviço de *Teleconsultation*, ignorando a sensibilidade da grafia do texto. O resultado da inferência é exibido na classe *QC-4-QoSPolicyService*.

## Quadro 17 - Regra SWRL da QC 4

QC 4: Quais são as políticas existentes para um determinado serviço?
nameEntity(?serv, ?str) ^ Service(?serv) ^ swrlb:stringEqualIgnoreCase(?str, "Teleconsultation") ^ Policy(?pol) ^ hasService(?pol, ?serv) -> ux:QC-4-QoSPolicyService(?pol)

O Quadro 18 apresenta as regras semânticas criadas para verificar as políticas para um parâmetro de QoS (*Latency*, *Jitter*, *Throughput* e *Packet Loss*), dado um tipo de serviço. Para cada regra, inicialmente, é verificado se existem instâncias de um parâmetro de QoS; Política e Serviço. Posteriormente, através da relação inversa *isParameterOf* e da propriedade de objeto *hasService* é verificado se o parâmetro de QoS é um parâmetro de política e se a política está associada ao serviço de *Teleconsultation*. O resultado do processo de inferência é feito nas classes respectivas (*QC-5-LatencyPolicy*; *QC-6-JitterPolicy*; *QC-7-PacketLossPolicy*; *QC-8-Throughput*).

Quadro 18 - Regras SWRL das QC 5, 6, 7 e 8

QC 5: Quais as políticas de QoS para a <i>Latency</i> , dado um tipo de serviço? isParameterOf(?lat, ?pol) ^ swrlb:stringEqualIgnoreCase(?str, "Teleconsultation") ^ nameEntity(?serv, ?str) ^ Latency(?lat) ^ Policy(?pol) ^ hasService(?pol, ?serv) -> ux:QC-5-LatencyPolicy(?pol)
QC 6: Quais as políticas de QoS para o <i>Jitter</i> , dado um tipo de serviço? isParameterOf(?jit, ?pol) ^ nameEntity(?serv, ?str) ^ Jitter(?jit) ^ Policy(?pol) ^ hasService(?pol, ?serv) ^ swrlb:stringEqualIgnoreCase(?str, "TELEconsultation") -> ux:QC-6-JitterPolicy(?pol)
QC 7: Quais as políticas de QoS para o <i>Packet Loss</i> , dado um tipo de serviço? swrlb:stringEqualIgnoreCase(?str, "Teleconsultation") ^ nameEntity(?serv, ?str) ^ isParameterOf(?loss, ?pol) ^ PacketLoss(?loss) ^ Policy(?pol) ^ hasService(?pol, ?serv) -> ux:QC-7-PacketLossPolicy(?pol)
QC 8: Quais as políticas de QoS para o <i>Throughput</i> , dado um tipo de serviço? isParameterOf(?thr, ?pol) ^ swrlb:stringEqualIgnoreCase(?str, "Teleconsultation") ^ nameEntity(?serv, ?str) ^ Throughput(?thr) ^ Policy(?pol) ^ hasService(?pol, ?serv) -> ux:QC-8-ThroughputPolicy(?pol)

Na regra do Quadro 19 após verificação das instâncias (Serviço, Política de QoS e Parâmetro de rede) é checado se a instância de QoS da rede é um parâmetro de política e se está associado ao serviço de Teleconsulta. As checagens são feitas através das propriedades de objetos, *isParameterOf* e *hasService*. O resultado do processo de inferência é disponibilizado na classe *QC-9-ServiceQoSParameters*.

Quadro 19 - Regra SWRL da QC 9

QC 9: Quais são os parâmetros de QoS de um determinado serviço? Teleconsultation(?serv) ^ ux:QoSPolicy(?pol) ^ QoSNetwork(?QoS) ^ isParameterOf(?QoS, ?pol) ^ hasService(?pol, ?serv) -> ux:QC-9-ServiceQoSParameters(?QoS)
--

Na regra do Quadro 20, inicialmente, é verificado se existem instâncias de Serviço e Política de QoS, na sequência, através do relacionamento *hasService* é analisado se a instância de política de QoS está associado ao serviço de Teleconsulta. A saída do processo de inferência é feita na classe *QC-10-ServiceQoSParameters*.

Quadro 20 - Regra SWRL da QC 10

QC 10: Quais são os parâmetros de QoS de um determinado serviço? QoSPolicy(?QoS) ^ Teleenvironmental(?serv) ^ hasService(?QoS, ?serv) -> QC-10-ServiceQoSParameters(?QoS)
--

A regra do Quadro 21 verifica se existem instâncias de Serviço e

do Conteúdo e, através do relacionamento *isContentOf* é conferido se a instância de conteúdo está associada ao serviço de Teleconsulta. O resultado da inferência é feita na classe *QC-12-ServiceContent*.

Quadro 21 - Regra SWRL da QC 11

QC 11: Qual o conteúdo de um determinado serviço?
$ux:Content(?cont) \wedge ux:Teleconsultation(?ser) \wedge ux:isContentOf(?cont, ?ser) \rightarrow ux:QC-11-ServiceContent(?cont)$

A regra do Quadro 22 verifica as instâncias de sessão, usuário, terminal, recurso e serviço e, através dos relacionamentos entre as classes é inferido o contexto do usuário “Joao Costa e Silva” ao consumir o serviço de Teleconsulta. O contexto do usuário, capturado durante a sessão, é inferido na classe *QC-12-UserContext*.

Quadro 22 - Regra SWRL da QC 12

QC12 - Qual o contexto do usuário na provisão de um determinado serviço?
$isSessionOf(?ses, ?res) \wedge isSessionOf(?ses, ?ter) \wedge User(?usr) \wedge isContextOf(?uc, ?ses) \wedge isTerminalOf(?ter, ?usr) \wedge UserContext(?uc) \wedge nameEntity(?usr, "Joao Costa e Silva") \wedge Teleconsultation(?serv) \wedge Terminal(?ter) \wedge Session(?ses) \wedge Resource(?res) \wedge isResourceOf(?res, ?serv) \rightarrow ux:QC-12-UserContext(?uc)$

Na regra do Quadro 23 é feita a verificação do contexto do usuário considerando a UX, ou seja, a nota atribuída pelo usuário ao consumir o serviço (5-Excelente; 4-Boa; 3-Razoável; 2-Ruim; 1-Péssima). No caso, a regra analisa o contexto do usuário, cujo MOS é maior ou igual a 3. Se as conjunções do antecedente são verdadeiras, então são inferidos, a partir do consequente da regra, o contexto e a UX na classe *UserExperienceContext*.

Quadro 23 - Regra SWRL da QC 13

QC13: Qual o contexto do usuário quando UX é avaliada na escala QoE-MOS, considerando um determinado serviço?
$isSessionOf(?ses, ?res) \wedge isSessionOf(?ses, ?ter) \wedge swrlb:greaterThanOrEqual(?valor, 3) \wedge User(?usr) \wedge mos(?ux, ?valor) \wedge isTerminalOf(?ter, ?usr) \wedge isContextOf(?uc, ?ses) \wedge UserContext(?uc) \wedge UserExperience(?ux) \wedge nameEntity(?usr, "Joao Costa e Silva") \wedge Teleconsultation(?serv) \wedge Terminal(?ter) \wedge Session(?ses) \wedge isUserExperienceOf(?ux, ?ses) \wedge Resource(?res) \wedge isResourceOf(?res, ?serv) \rightarrow ux:QC-13-UserExperienceContext(?uc) \wedge ux:QC-13-UserExperienceContext(?ux)$

A regra do Quadro 24 é similar a regra do Quadro 23, porém, neste caso é analisado o contexto da rede quando o usuário avalia o serviço na escala MOS, com valor maior que 3. Considerando que as conjunções do antecedente são verdadeiras, o contexto da rede e a experiência do usuário são inferidos na classe *QoSNetworkContext*.

Quadro 24 - Regra SWRL da QC 14

QC 14: Qual o contexto da rede quando UX é avaliada na escala QoE-MOS considerando um determinado serviço?
$\text{Telebiomedical}(\text{?serv}) \wedge \text{Route}(\text{?rou}) \wedge \text{isContextOf}(\text{?rou}, \text{?ses}) \wedge \text{Session}(\text{?ses}) \wedge \text{isSessionOf}(\text{?ses}, \text{?res}) \wedge \text{Resource}(\text{?res}) \wedge \text{isResourceOf}(\text{?res}, \text{?serv}) \wedge \text{UserExperience}(\text{?ux}) \wedge \text{isUserExperienceOf}(\text{?ux}, \text{?ses}) \wedge \text{mos}(\text{?ux}, \text{?valor}) \wedge \text{swrlb:greaterThan}(\text{?valor}, 3) \rightarrow \text{ux:QC1-14-QoSNetworkContext}(\text{?rou}) \wedge \text{ux:QC1-14-QoSNetworkContext}(\text{?ux})$

A regra do Quadro 25 completa a regra do Quadro 24, a diferença é que estão sendo analisados ambos os contextos, rede e usuário, quando o serviço é avaliado na escala MOS, com valor maior ou igual a 3. Assumindo que as conjunções do antecedente são verdadeiras, o contexto da rede e do usuário são inferidos na classe indicada.

Quadro 25 - Regra SWRL da QC 15

QC 15 - Qual o contexto do usuário e da rede quando UX é avaliada na escala QoE-MOS, considerando um determinado serviço?
$\text{isSessionOf}(\text{?ses}, \text{?res}) \wedge \text{Route}(\text{?rou}) \wedge \text{isSessionOf}(\text{?ses}, \text{?ter}) \wedge \text{swrlb:greaterThanOrEqual}(\text{?valor}, 3) \wedge \text{User}(\text{?usr}) \wedge \text{mos}(\text{?ux}, \text{?valor}) \wedge \text{isTerminalOf}(\text{?ter}, \text{?usr}) \wedge \text{isContextOf}(\text{?uc}, \text{?ses}) \wedge \text{UserContext}(\text{?uc}) \wedge \text{UserExperience}(\text{?ux}) \wedge \text{Teleconsultation}(\text{?serv}) \wedge \text{isContextOf}(\text{?rou}, \text{?ses}) \wedge \text{Terminal}(\text{?ter}) \wedge \text{Session}(\text{?ses}) \wedge \text{isUserExperienceOf}(\text{?ux}, \text{?ses}) \wedge \text{nameEntity}(\text{?usr}, \text{"Joao Costa e Silva"}) \wedge \text{Resource}(\text{?res}) \wedge \text{isResourceOf}(\text{?res}, \text{?serv}) \rightarrow \text{ux:QC-15-UserExperienceQoSNetworkContext}(\text{?uc}) \wedge \text{ux:QC-15-UserExperienceQoSNetworkContext}(\text{?rou})$

A regra do Quadro 26 complementa a regra do Quadro 25, contudo a regra analisa o contexto do usuário; da rede e do conteúdo. Neste caso, assumindo que as conjunções do antecedente são verdadeiras, os contextos analisados (usuário, rede e conteúdo) são inferidos na classe indicada somente quando o serviço é avaliado na escala MOS, com valor maior ou igual a 3.



Quadro 26 - Regra SWRL da QC16

QC16 - Qual o contexto do usuário, do conteúdo e da rede quando UX é avaliada na escala QoE-MOS considerando um determinado serviço?
$\begin{aligned} & ux:isSessionOf(?ses, ?res) \wedge ux:Route(?rou) \wedge ux:isSessionOf(?ses, ?ter) \wedge \\ & swrlb:greaterThanOrEqualTo(?valor, 3) \wedge autogen0:User(?usr) \wedge ux:mos(?ux, \\ & ?valor) \wedge ux:isTerminalOf(?ter, ?usr) \wedge ux:isContextOf(?uc, ?ses) \wedge \\ & ux:UserContext(?uc) \wedge ux:UserExperience(?ux) \wedge ux:Teleconsultation(?serv) \wedge \\ & ux:isContextOf(?rou, ?ses) \wedge ux:Terminal(?ter) \wedge ux:Session(?ses) \wedge \\ & ux:Content(?cont) \wedge ux:isContentOf(?cont, ?serv) \wedge autogen0:nameEntity(?usr, \\ & "Joao Costa e Silva") \wedge ux:isUserExperienceOf(?ux, ?ses) \wedge ux:Resource(?res) \\ & \wedge ux:isResourceOf(?res, ?serv) \rightarrow ux:QC-16- \\ & \quad UserExperienceContentQoSNetworkContext(?rou) \wedge ux:QC-16- \\ & \quad UserExperienceContentQoSNetworkContext(?cont) \wedge ux:QC-16- \\ & \quad UserExperienceContentQoSNetworkContext(?uc) \end{aligned}$

Na regra do Quadro 27, as conjunções do antecedente procuram por instâncias da sessão; da UX; dos parâmetros de QoS monitorados e, se os parâmetros monitorados estão associados com a sessão. Em adição é verificado a UX quando o serviço é avaliado com valor menor do que 2. Considerando que as conjunções do antecedente são verdadeiras, o resultado do processo da inferência é feito na classe *QoSViolation*.

Quadro 27 - Regra SWRL da QC 17

QC 17 - Houve violação da política de QoS que interfere na UX no consumo de um tipo de serviço?
$\begin{aligned} & Session(?ses) \wedge UserExperience(?ux) \wedge isUserExperienceOf(?ux, ?ses) \wedge \\ & QoSmonitor(?mon) \wedge isMonitorOf(?mon, ?ses) \wedge mos(?ux, ?valor) \wedge \\ & swrlb:lessThan(?valor, 2) \rightarrow ux:QC-17-QoSViolation(?mon) \end{aligned}$

O Quadro 28 apresenta as regras semânticas criadas para verificar se o valor (mínimo e máximo) de um parâmetro de QoS monitorado está fora dos valores especificados na política de QoS. São apresentadas as quatro regras usadas pelo motor de inferência para classificar os alertas.

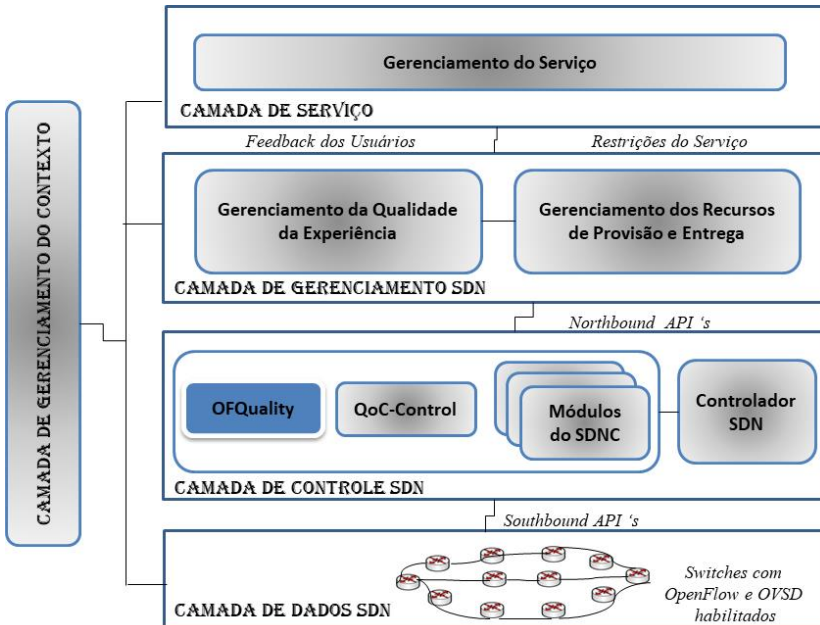
Quadro 28 - Regras SWRL das QC 18, 19, 20 e 21

QC 18: Houve violação da política de QoS do parâmetro <i>latency</i> para um determinado serviço?
$\begin{aligned} & MonitorLatency(?lat) \wedge qualityLiteralValue(?lat, ?valor) \wedge \\ & swrlb:greaterThan(?valor, ?max) \wedge MaxValue(?pol, ?max) \wedge isMonitorOf(?lat, \\ & ?pol) \wedge eHealth(?serv) \wedge hasService(?pol, ?serv) \wedge PolicyLatency(?pol) \rightarrow \\ & \quad ux:QoC-18-LatencyViolation(?lat) \end{aligned}$
QC 19: Houve violação da política de QoS do parâmetro <i>Jitter</i> para um determinado serviço?

$\text{MonitorJitter}(\text{?jit}) \wedge \text{qualityLiteralValue}(\text{?jit}, \text{?valor}) \wedge \text{swrlb:greaterThan}(\text{?valor}, \text{?max}) \wedge \text{MaxValue}(\text{?pol}, \text{?max}) \wedge \text{isMonitorOf}(\text{?jit}, \text{?pol}) \wedge \text{eHealth}(\text{?serv}) \wedge \text{hasService}(\text{?pol}, \text{?serv}) \wedge \text{PolicyJitter}(\text{?pol}) \rightarrow \text{ux:QoC-19-JitterViolation}(\text{?jit})$
QC 20: Houve violação da política de QoS do parâmetro <i>Throughput</i> para um determinado serviço?
$\text{MonitorThroughput}(\text{?thr}) \wedge \text{isMonitorOf}(\text{?thr}, \text{?pol}) \wedge \text{PolicyThroughput}(\text{?pol}) \wedge \text{qualityLiteralValue}(\text{?thr}, \text{?valor}) \wedge \text{swrlb:lessThan}(\text{?valor}, \text{?min}) \wedge \text{MinValue}(\text{?pol}, \text{?min}) \wedge \text{eHealth}(\text{?serv}) \wedge \text{hasService}(\text{?pol}, \text{?serv}) \rightarrow \text{ux:QoC-20-ThroughputViolation}(\text{?thr})$
QC 21: Houve violação da política de QoS do parâmetro <i>Packet Loss</i> para um determinado serviço?
$\text{MonitorPacketLoss}(\text{?pl}) \wedge \text{qualityLiteralValue}(\text{?pl}, \text{?valor}) \wedge \text{swrlb:greaterThan}(\text{?valor}, \text{?max}) \wedge \text{MaxValue}(\text{?pol}, \text{?max}) \wedge \text{isMonitorOf}(\text{?pl}, \text{?pol}) \wedge \text{eHealth}(\text{?serv}) \wedge \text{hasService}(\text{?pol}, \text{?serv}) \wedge \text{PolicyPacketLoss}(\text{?pol}) \rightarrow \text{ux:QoC-21-PacketLossViolation}(\text{?pl})$

## 5.5 ARQUITETURA DE GERENCIAMENTO DA QUALIDADE DE EXPERIÊNCIA

Figura 40 - Arquitetura conceitual para a gerência da QoE



Para que o modelo de representação do conhecimento da QoE seja avaliado, o mesmo precisa ser incorporado numa plataforma de prestação de serviços orientada a UX e que adote uma abordagem centrada na experiência do usuário. Para isso é proposta uma arquitetura flexível com o uso de SDN, onde o modelo é incorporado, cooperando e sendo cooperado com os componentes que constituem essa arquitetura.

A arquitetura proposta, estruturada por cinco camadas, é ilustrada através da Figura 40. A principal contribuição desta tese está relacionada com os componentes da camada de gerenciamento da Qualidade de Experiência.

### **5.5.1 Camada de Gerenciamento do Contexto**

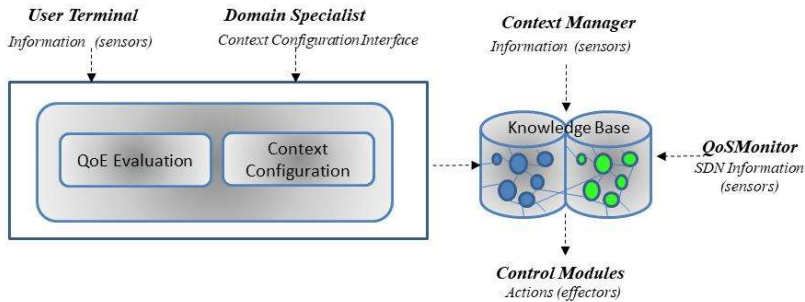
Responsável por capturar, processar e distribuir o contexto para as entidades das demais camadas da arquitetura proposta. Para o gerenciamento do contexto de um determinado serviço é necessário conhecer o provedor e o consumidor do contexto. Neste caso, o provedor do contexto são todas as entidades (e.g., usuário, sensores, recursos de rede) que provisionam dados e/ou informações para os consumidores do contexto.

Para o escopo deste trabalho, consideramos que o contexto é toda informação usada para caracterizar a situação das entidades que tenham influência na UX. Assim, habilitadores de contexto devem ser capazes de capturar o contexto do usuário; informações que permitam descobrir padrões de comportamento (e.g., horário de utilização, dia da semana, localização); informações do dispositivo de acesso ao serviço (e.g., tamanho da tela, bateria, processador, memória, rede de acesso); disponibilidade e desempenho dos recursos que provisionam e entregam o serviço.

### **5.5.2 Camada de Gerenciamento do Serviço**

Refere-se ao gerenciamento dos serviços providos por um provedor, tais como telemedicina, telemonitoramento, monitoramento residencial e de segurança, entre tantos outros. A Figura 41 apresenta os componentes propostos pelo módulo de gerenciamento do serviço, “*QoE Evaluation*” e “*Context Configuration*”, usados respectivamente, para a avaliação do serviço e para que o especialista do domínio possa definir as configurações necessárias para o funcionamento adequado do serviço.

Figura 41 - Componentes da camada de gerenciamento de serviços



### 5.5.2.1 Context Constraints

Fornece uma interface para que os especialistas do domínio informem a parametrização e restrições da QoS de acordo com o contexto, aqui tratadas como políticas. Dependendo do tipo de serviço (e.g., monitoramento de sinais ambientais; monitoramento de sinais vitais) restrições de QoC também podem ser definidas.

Uma arquitetura para o gerenciamento da QoC já foi proposta por Nazário e Dantas (2015), a novidade é que estamos propondo a interface de configuração do contexto e o mapeamento para uma base formal (SILVA et al., 2016c). Além disso, na arquitetura de Nazário e Dantas (2015) não é prevista a distribuição do contexto em redes de transporte de dados, sendo que os dados monitorados são persistidos apenas na base local do AAL, portanto, a QoS não é abordada.

Considerando a abordagem semântica proposta, as restrições de contexto, aqui definidas através de políticas, precisam ser mapeadas para a base ontológica do provedor de serviço. Assim, a partir de uma interface (aplicação de software), os especialistas de domínio informam as restrições da qualidade associada com as entidade de contexto, conforme simplificação da Tabela 16.

Tabela 16 - Exemplo de política de QoS e QoC

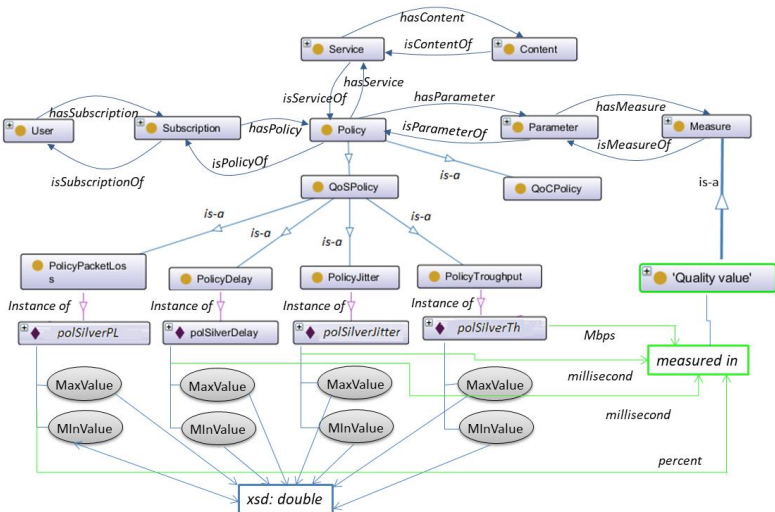
<b>Tipo de Serviço:</b> Monitoramento de sinais ambientais	
<b>Informações de Contexto:</b> temperatura, umidade, usuário	
<b>Tipo de conteúdo midiático:</b> dados	
<b>Políticas de QoS:</b>	<b>Políticas de QoC:</b>
<i>Throughput:</i> 0.5 a 1 Kbps;	Up-to-dateness: 25s;
<i>Latency:</i> 50 a 100ms;	Coverage: umidade: 20 – 90%, temperature: 0 – 50°C;
<i>Jitter:</i> 0 e 50ms;	Precision: Humidity 5%-P=0.95, temperature +-2°C -> P= 0.92;
<i>Packet Loss:</i> 0 a 0.5%;	Completeness: Both of the details (humidity, temperature);
	Significance: Humidity 30 – 85%, temperature 15 – 35°C.

As políticas de QoS e QoC definidas na KB são usadas no processo de inferência pelo mecanismo de inteligência (*QoE-ManApp*) para a tomada de decisões de engenharia de tráfego e restauração da QoE.

Para que as informações sejam mapeadas para a ontologia foi adotado o paradigma orientado a objetos. Assim, um objeto no código representa uma instância na ontologia, seus atributos são mapeados com as propriedades das instâncias e as classes em owl se tornam classes na linguagem de programação. Logo, para adicionar um recurso na ontologia é preciso adicionar o objeto, facilitando assim, o desenvolvimento dessas aplicações.

A Figura 42 é uma simplificação da estrutura da KB usada para definir as políticas contratadas por um assinante de um serviço. Na simplificação ontológica estão sendo definidas as políticas de QoS para os parâmetros de desempenho da rede (*atraso*, *jitter*, *descarte* e *throughput*), levando em consideração o contexto (usuário, assinatura, serviço, conteúdo). Para cada parâmetro de QoS foi definida uma política com um valor máximo e mínimo aceitável e a unidade de medida de cada parâmetro. Observe que na Figura 42 não está sendo mostrada a política de QoC, nem as entidades associadas ao contexto definidas na Tabela 16.

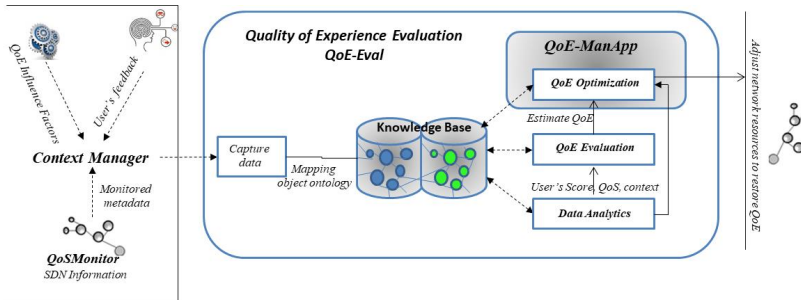
Figura 42 - Estrutura simplificada da KB usada na definição da política de QoS



### 5.5.2.2 QoE Evaluation (QoEval)

O modelo ideal de avaliação da QoE é aquele construído com a participação dos usuários, assim, um modelo híbrido, composto por métodos subjetivos e objetivos é adotado para estimar a qualidade percebida. Para a avaliação da QoE ser bem sucedida é necessário calcular os parâmetros de QoS monitorados durante a sessão; capturar o *feedback* do usuário após o consumo do serviço; capturar e processar demais informações de contexto que podem influenciar a UX para o serviço em questão. Todas as informações capturadas devem ser mapeadas para a KB. Esses procedimentos são sintetizados na Figura 43.

Figura 43 - Processo de avaliação da QoE



Numa rede SDN cada serviço provido a um usuário é representado por um fluxo, ou conexão, e antes de ser aplicado qualquer ação, o fluxo precisa ser identificado. Assim, quando um novo fluxo é detectado pelo módulo de controle, *QoE-Control* (seção 4.4.4.3), inicialmente ele observa na KB as políticas de QoS estabelecidas para o fluxo, aloca recursos e estabelece um caminho (rota) para a entrega do mesmo. Na sequência, através do módulo *QoS-Monitor* inicia-se o processo de monitoração dos parâmetros de QoS, cada parâmetro monitorado é armazenado na KB e o processo de monitoração fica ativo até o término da conexão.

Ao término da conexão o usuário pode fazer a avaliação do serviço e enviá-la ao *QoE-Eval* (módulo de avaliação da QoE). Quando o usuário avalia o serviço, o valor da pontuação (um valor numa escala de 1-5, sendo 5 para “excelente UX” e 1 “para péssima UX”) é persistido na KB juntamente com os valores das medições dos parâmetros de rede, conforme ilustra a Tabela 17.

Em paralelo durante a provisão do serviço são capturadas as informações contextuais<sup>5</sup> que interferem na QoE. Essas informações (*feedback* dos usuários, QoS, fatores de influência na QoE) são usadas como entradas para treinar um modelo de previsão da QoE/MOS e para descobrir novos valores máximos e mínimos da QoS.

A Tabela 17 apresenta exemplos de instâncias persistidas na KB, contendo o valor do MOS em função dos parâmetros de desempenho da rede, considerando um contexto conhecido. As instâncias são exemplos de avaliações do MOS pelos usuários Alice e Bob para o serviço de *eHealth*<sup>6</sup> “Pesquisa e educação conversacional”, utilizado com o uso de ferramentas colaborativas de pesquisa/educação envolvendo áudio e vídeo conversacional.

Tabela 17 - Exemplos de instâncias da KB usadas para a predição do MOS

<i>User</i>	<i>Contexto do Usuário</i>			<i>QoS da rede</i>				<i>MOS</i>
<i>Nome</i>	<i>Computação</i>	<i>Atividade Física</i>	<i>Ambiente Físico</i>	<i>Delay (ms)</i>	<i>Jitter (ms)</i>	<i>Perda (%)</i>	<i>Through put</i>	<i>Valor</i>
Alice	Celular	Caminhan do	Quarto	10	0	1	330006	3,913493
Alice	Notebook	Parado	Sala	0	2	1	732177	4,977004
Bob	Celular	Caminhan do	Sala	5	0	2	353601	4,036015
Alice	Notebook	Parado	Sala	10	50	1	643202	3,79699
Bob	Celular	Caminhan do	Quarto	0	0	0	401224	4,197624
Bob	Notebook	Parado	Sala	0	30	0	732177	4,476978

A Figura 44 representa uma simplificação da estrutura da KB usada para monitorar a UX. Por questão de simplificação e clareza não estão sendo exibidas as instâncias que representam os parâmetros/métricas monitorados, nem as propriedades de dados de cada instância.

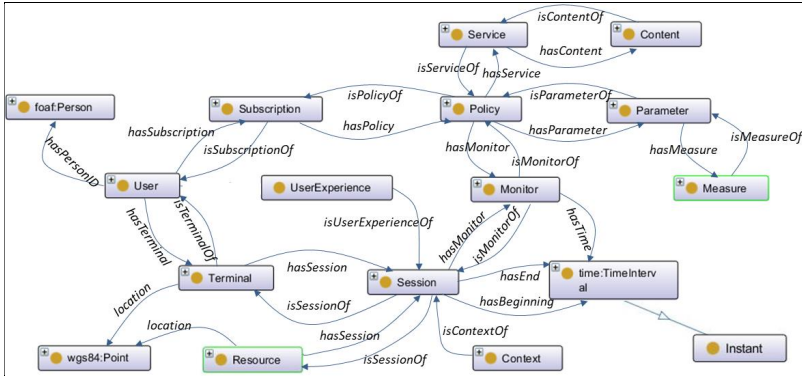
As informações persistidas na KB são usadas para que o mecanismo de inteligência possa verificar se o serviço está sendo provido de acordo com as políticas contratadas, bem como derivar novos fatos para otimizar a QoE. Com base na análise dos dados, havendo necessidade de restaurar ou otimizar a QoE, os mecanismos de controle são acionados para que a rede seja ajustada as necessidades do usuário. A etapa de restauração e/ou otimização da QoE apresentada na Figura 43 é realizada na camada de controle, portanto, é de

<sup>5</sup> Levantadas na Taxomia de dimensões de influência da QoE (Figura 33).

<sup>6</sup> Os tipos de serviços de *eHealth* e a aplicações de uso foram apresentados na seção 3.2.5 (Tabela 10)

responsabilidade da aplicação de gerenciamento da QoE (*QoE-ManApp*), analisar e planejar ações para otimização da QoE e posterior sinalizações para os mecanismos que controlam a rede.

Figura 44 - Simplificação da estrutura da KB usada para a avaliação da UX



### 5.5.2.3 Modelo de Mapeamento para Predição do MOS

Conforme dito anteriormente, o conhecimento das avaliações dos usuários persistido na KB é usado como entrada para treinar um modelo de predição do MOS. Para isso uma função de mapeamento dos parâmetros de QoS em QoE é adotada e, utilizando os valores dos parâmetros monitorados é capaz de estimar o MOS, considerando um determinado contexto. Esta etapa acontece *off-line* junto ao processo de “*Data analysis*” da Figura 43, sendo realizada por um algoritmo de aprendizagem de máquina.

Note que nem todos os serviços precisam de uma função de mapeamento de QoS em QoE e sim, somente aqueles em que há interação ou consumo midiático (e.g., áudio, vídeo) pelo usuário. Como por exemplo, no sistema de monitoramento de sinais biomédicos, o contexto é apenas transmitido (e.g., pressão arterial, temperatura corporal, pulso e oxigenação) para ser consumido remotamente por uma aplicação (e.g., prontuário eletrônico do paciente, painel de monitoração biomédica), neste caso, o usuário (máquina/sistema pertencentes ao usuário) que provê o contexto, portanto, não avalia, apenas provê dados. Neste situação, a função de mapeamento não se aplica. Contudo quando o serviço consumido é o de teleconsulta, por exemplo, há interação do usuário com pelo menos um tipo de conteúdo midiático (áudio ou áudio/vídeo), logo, o usuário pode avaliar o serviço e assim, uma função



de mapeamento é usada para estimar o MOS. Naturalmente que a KB deve estar populada para que haja a aprendizagem pelo algoritmo.

Para cada tipo de serviço em que há interação e/ou consumo midiático pelo usuário, existe uma função de mapeamento associada, assim, se o conteúdo midiático é composto apenas por áudio, existe uma equação específica para estimar o MOS do áudio e assim por diante. Em Tsolkas et al. (2017) são apresentados e avaliados os modelos para estimar o MOS dos serviços mais populares, incluindo VoIP, VoD, IPTV, transmissão de vídeo, navegação Web, transferência de arquivos, Skype.

A Equação 21 usa regressão linear para treinar um modelo de previsão do MOS do serviço de *eHealth* do tipo “Pesquisa e educação conversacional”. O modelo deve ser treinado cada vez que o contexto é alterado.

$$\text{MOS} = \alpha \text{ Thr} + \beta \text{ Jt} + \gamma \text{ Plr} + \varepsilon \quad (21)$$

*Thr* é o *throughput*; *Jt* é o *jitter* e *Plr* é taxa de perda de pacotes.  $\alpha$ ,  $\beta$ ,  $\gamma$ , e  $\varepsilon$  são os coeficientes calculados para cada caso particular.

Em síntese, não existe uma função específica para prever a UX, uma vez que os parâmetros e métricas obtidos/calculados são dependentes do tipo de conteúdo midiático do serviço (texto, áudio, vídeo, áudio e vídeo). Além disso, na aprendizagem de máquina, o algoritmo deve levar em consideração o contexto do usuário. E toda vez que o contexto do usuário é alterado o modelo deve ser treinado. Esse processamento é custoso computacionalmente, portanto, deve ser realizado *off-line*, mas, a KB é atualizada imediatamente após o treinamento do modelo.

Cada modelo deve ser criteriosamente analisado para conhecer a sua efetividade. No caso do modelo de regressão linear da Equação 21, as seguintes análises devem ser realizadas:

- 1) Existem dados suficientes na KB para a avaliação do modelo?
- 2) Qual o valor F de significância global do modelo?
- 3) Há evidências de que pelo menos uma variável no modelo está relacionada com o MOS?
- 4) Quais variáveis estão relacionadas com o MOS?
- 5) O quanto as variáveis independentes (métricas de QoS) explicam a variabilidade do MOS?

A respostas das perguntas acima descritas vão sinalizar a utilidade do modelo.

### 5.5.3 Camada de Gerenciamento SDN

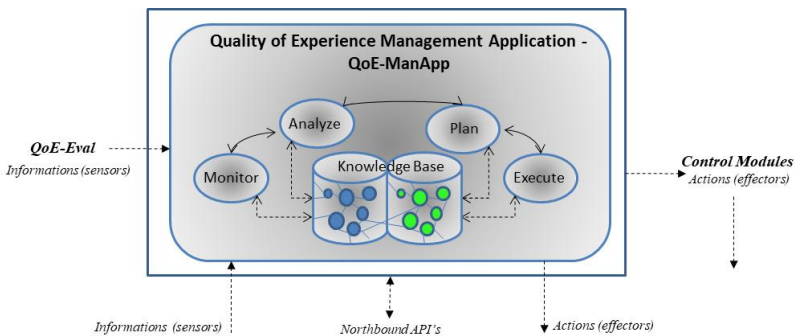
Provê as funcionalidades necessárias para a gestão da UX no uso de um serviço. Além de prover mecanismos para desencadear ações de adaptações nos recursos que provisionam e entregam o serviço. Essas ações são necessárias para manter o nível da QoE de acordo com as necessidades do usuário, levando em consideração o contexto.

A camada de gerenciamento SDN é composta por aplicações de “Gerenciamento da Qualidade de Experiência” e “Gerenciamento dos Recursos”, descritos a seguir.

#### 5.5.3.1 Quality Of Experience Management Application – QoE-ManApp

A Aplicação de Gerenciamento de QoE (*Quality Of Experience Management Application – QoE-ManApp*) foi implementada com duas principais características: (i) semântica e (ii) autonômica. Com estas características implementadas a *QoE-ManApp* é capaz de aprender a experiência do usuário, com base nas informações providas da *QoE-Eval*, disponíveis na KB; gerar alertas sobre degradações e solicitar a restauração da QoE, baseado em políticas.

Figura 45 - Arquitetura da aplicação de gerenciamento de QoE



A Figura 45 apresenta a arquitetura da *QoE-ManApp*. A primeira característica da *QoE-ManApp* foi contemplada com a inclusão da Ontologia de QoE, base de conhecimento, base de regras e pelos

recursos do motor semântico Pellet (SIRIN et al., 2007). O motor semântico é usado para verificar inconsistências no modelo e para raciocinar sobre as regras. As funcionalidades da *QoE-ManApp* foram implementadas com o *loop* de controle MAPE-K (*Monitor, Analyze, Plan, Execute, Knowledge*), agregando assim, a segunda característica em sua estrutura. O MAPE-K foi inspirado na arquitetura *blueprint* para computação autônoma da IBM (2005).

**a) Ontologia, Base de Regras e Base de Conhecimento:** a ontologia de QoE foi descrita na seção 4.4, ela é usada para unificar conceitos, facilitando o conhecimento sobre a UX no uso de um serviço e gerar alertas de violações de parâmetros estabelecidos na política. Os alertas classificados pelo raciocinador são realizados por regras semânticas.

A base de regras, apresentada na seção 4.4.4, foi criada para a inferência por tipo de alerta, considerando a política e o contexto associado. As regras são verificadas pelo motor de inferência, que em tempo de execução analisa os parâmetros passados na regra, observa as restrições, verifica na KB as instâncias que satisfazem as restrições das regras e, procede com a classificação dos alertas.

A KB mantém o conhecimento sobre as entidades gerenciadas (usuário, rede, contexto, conteúdo, serviço), políticas de QoS, políticas de QoC e as regras semânticas. Todas as ações realizadas pelo MAPE-K são feitas com base nas informações na KB e nas informações providas pelas aplicações de gerenciamento e controle SDN.

**b) Loop de Controle MAPE-K:** as funcionalidades da *QoE-ManApp* foram construídas com MAPE-K “*Monitor, Analyze, Plan, Execute e Knowledge*”, criando uma seqüência de ações necessárias para a gestão autônoma da QoE.

(i) *Funcionalidades do Módulo “Monitor”:* coleta os dados dos recursos de rede para estimar o desempenho dos parâmetros de QoS através da aplicação de gerenciamento *QoS-Monitor*. O *QoS-Monitor* coleta as estatísticas dos módulos *Statistics Collector* e *Topology* da camada de controle, usando a API REST do controlador *Floodlight*. Para a monitoração dos parâmetros de QoS desenvolvemos um algoritmo que obtém estatísticas dos recursos de rede, explicado nas seções 4.5.3.2 e 4.5.3.3.

(ii) *Funcionalidades do Módulo “Analyze”*: sempre que uma nova entrada de dados monitorados é persistida na KB, o motor de inferência automaticamente interpreta as regras semânticas e verifica se alguma restrição de QoS descrita na base de política foi violada. Conforme já explicado na seção 4.4.4, cada valor do parâmetro monitorado é comparado com as definições de políticas na KB. Havendo degradação, como resultado da inferência são classificados os alertas por tipo de parâmetro. Neste caso, o alerta é analisado e aciona-se o módulo “Plan” para que uma ação de adaptação de política seja planejada e aplicada.

(iii) *Funcionalidades do Módulo “Plan”*: realiza o planejamento e determina qual ação de adaptação de política deve ser adotada. A Tabela 18 apresenta alguns exemplos de ações de adaptações de políticas que podem ser aplicadas para otimizar a QoE. Para o planejamento da ação a ser aplicada, o módulo “Plan” necessita obter uma visão geral da rede, conhecer os recursos alocados e o caminho da disponibilidade dos recursos. O caminho da disponibilidade é calculado pelo *QoS-Router*, as informações da topologia da rede e dos recursos alocados são obtidos dos módulos de controle via API REST.

Tabela 18 - Ações de adaptações de políticas

Métrica de QoS	Ações de Adaptações de Políticas
Descarte de Pacotes	Alterar a configuração da fila. Encaminhar o fluxo através de uma rota alternativa, com melhor caminho.
Throughput	Alterar os limitadores de taxas dos fluxos que estão saturando a banda Encaminhar o fluxo através de uma rota alternativa.
Delay	Alterar a configuração da fila no <i>switch</i> . Planejar a transmissão do fluxo através de um caminho menos congestionado.
Jitter	Planejar a transmissão do fluxo através de um caminho menos congestionado.

Fonte adaptado de Bari et al. (2013)

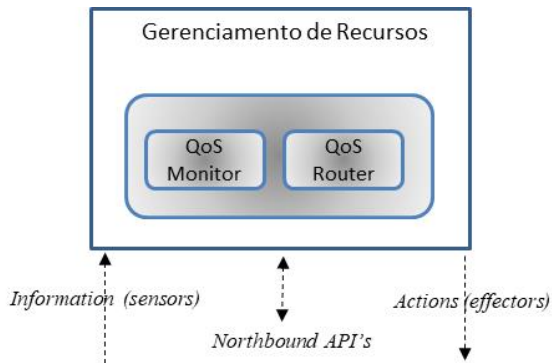
(iv) *Funcionalidades do Módulo “Execute”*: a fim de aplicar a ação planejada este módulo, usando a API REST, se comunica com os módulos de controle *OFQuality* e *QoE-Control* para solicitar a ação planejada sobre os dispositivos do plano de dados. Se a adaptação de política estiver relacionada com alterações na largura de banda ou na prioridade de um determinado fluxo, a comunicação é feita com o módulo *OFQuality*. No entanto, se a adaptação de política for referente ao roteamento, a comunicação será feita com *QoE-Control*. Em ambos os casos a comunicação é feita via API REST.

### 5.5.3.2 Gerenciamento de Recursos

Uma arquitetura de provisão e entrega de serviços deve ser dotada de aplicações que permita o monitoramento da QoS; conhecer os recursos disponíveis e o caminho da disponibilidade dos recursos, assim, decisões de engenharia de tráfego podem ser tomadas para evitar degradação de um serviço que está sendo provisionado. Com este propósito foram idealizadas as aplicações *QoS-Monitor* e *QoS-Route*, apresentadas da Figura 46.

A maioria dos serviços implementados pelo controlador *Floodlight* podem ser acessados pela camada de gerenciamento, via API REST, portanto as aplicações *QoS-Monitor* e *QoS-Router* foram prototipados usando os serviços dos módulos ‘*Topology & Routing*’ e ‘*Statistics Collector*’, acessíveis via API REST do *Floodlight*.

Figura 46 - Aplicações de gerenciamento de recursos



**a) QoS-Monitor:** monitora os parâmetros de QoS (*throughput*, latência, *jitter*, perda de pacotes, taxa de utilização) dos *switches* e dos *links* usando estatísticas do módulo *Statistics Collector* e informações do módulo *Topology*, ambos da camada de controle.

Para obter a métrica de cada parâmetro monitorado e estimar a latência entre os *switches* no caminho, foram utilizadas as mensagens *OpenFlow*, trocadas entre o controlador e os *switches*. Além das mensagens *OpenFlow* foi utilizado um pacote de sonda (*probe*) para auxiliar na estimativa da latência, conforme já apresentado na seção 3.5.6. As mensagens *OpenFlow* usadas para coletar as estatísticas foram apresentadas no Quadro 5, devidamente explicadas na seção 3.5.4.

Conforme já explanado, as estatísticas de tráfego no *OpenFlow*

podem ser extraídas em diferentes níveis de abstração (e.g., porta, fila, fluxo) a partir de *switches* individuais, com base nos contadores nas tabelas de fluxo. Contudo essas estatísticas precisam receber um significado, assim, para obter o valor de cada métrica individual foram adotadas algumas equações já validadas na literatura, apresentadas na seção 3.5.6.

**b) QoS-Router:** utiliza as informações das métricas de QoS obtidas pelo módulo *QoS-Monitor* para tomar decisões de Engenharia de Tráfego de acordo com os requisitos de QoS das adaptações de políticas (Tabela 18). Desta forma, esta aplicação fornece informações do caminho da disponibilidade dos recursos para que a aplicação de gerenciamento *QoE-ManApp* possa planejar e solicitar a troca de rota.

A *QoS-Router* foi inspirada pela aplicação de engenharia de tráfego proposto por Santos, Ribeiro e Cesar (2015). A diferença do nosso algoritmo é que a troca de rota acontece quando se conhece a priori a métrica de QoS violada e uma ação de adaptação de política planejada deve ser executada. No algoritmo proposto por Santos, Ribeiro e Cesar (2015) é ofertada a melhor rota de acordo com a relevância do parâmetro de QoS para a aplicação (e.g., se o fluxo ativo é de uma aplicação VoIP, o atraso é crítico; se o fluxo é de FTP, a largura de banda é crítica). Em ambos os casos é usado o algoritmo clássico de *Dijkstra Shortest Path Algorithm* (SPA).

A função de custo é definida de acordo com os parâmetro de QoS dos *links*. Os parâmetros de QoS de cada *link* fornecidos pelo *QoS-Monitor* são transformados em um parâmetro de custo positivo e a função de custo tem três elementos (SANTOS; RIBEIRO, CESAR, 2015): identificação do *link*; o valor do custo atual; estatísticas atuais, usadas para calcular o custo.

Uma vez que a adaptação de política considera a métrica de QoS individual é necessário uma função de custo por métrica de QoS para o cálculo da rota. As funções de custos usadas na *QoS-Router* são aquelas propostas por Santos, Ribeiro e Cesar (2015), descritas na seção 3.5.7.

De forma sintetizada o algoritmo de roteamento funciona da seguinte forma:

- i. Passo 1: Encontre todos os caminhos da origem para o destino usando o algoritmo de *Dijkstra*.
- ii. Passo 2: Selecione os valores dos parâmetros de QoS da aplicação *QoS-Monitor* para calcular cada função de custo.
- iii. Passo 3: Calcule cada função de custo por parâmetro de QoS no caminho da origem para o destino.

- iv. Passo 4: Classifique os caminhos em ordem crescente por função de custo de acordo com o tipo de parâmetro de QoS.
- v. Passo 5: Armazene o custo de cada caminho, com informações que permitam mantê-los atualizados (identificação do *link*; o valor do custo atual; estatísticas atuais) na KB.

Todos os caminhos calculados pela *QoS-Router* são persistidos na KB para que a *QoE-ManApp* possa planejar uma ação por troca de rota. Da forma com que o algoritmo se comporta, o caminho da disponibilidade está sempre atualizado e disponível na KB, facilitando o planejamento de uma ação de adaptação de política por troca de rota.

#### 5.5.3.3 Algoritmo da Aplicação de Gerenciamento da QoE (QoE-ManApp)

A seguir são explicadas todas as etapas envolvidas na detecção, gerenciamento e finalização do fluxo. Inicialmente o fluxo é detectado. Essa tarefa é de responsabilidade da *thread* principal, realizada pelo *QoS-Control*. Para isso ela implementa um ouvinte de mensagem *OpenFlow* que escuta pacotes que chegam no controlador, analisa se eles estão mapeados na KB e cria um *thread* de gerenciamento para cada fluxo específico e o controle do fluxo passa a ser da *QoE-ManApp*. Esta *thread* é executada até que o fluxo seja finalizado. Ela é responsável por coletar, analisar, planejar e executar ações para otimizar a QoE. A Figura 47 sintetiza a lógica do algoritmo.

Figura 47 - Algoritmo de gerenciamento do fluxo

---

**Algorithm 1:** Flow's main thread algorithm

---

```

1  $pMap \leftarrow getPoliciesMapParams()$ ;
2  $coeffsMap \leftarrow getMOSCoeff()$ ;
3  $tryChangePathBand \leftarrow false$ ;
4  $tryChangePathLat \leftarrow false$ ;
5 while true do
6    $measureThroughput()$ ;
7    $measureLatency()$ ;
8    $measurePacketLoss()$ ;
9    $measureJitter()$ ;
10   $mos \leftarrow calculateMOS()$ ;
11   $insertDataKB(bandmax, lat, ploss, jit, mos)$ ;
12  if  $((b1 > chRateUp) \text{ and } (b2 > chRateUp)) \text{ or } ((b2 > chRateUp)$ 
    and  $(b3 > chRateUp))$  then
13    if not  $increaseBandwidth()$  then
14       $tryChangePathBand \leftarrow true$ ;
15    end
16  end
17  else if  $(flowLatencyCurrentPath > maxLatency)$  then
18     $tryChangePathLat \leftarrow true$ ;
19  end
20  else if  $(flowPacketLoss > maxPacketLoss)$  then
21     $tryChangePathBand \leftarrow true$ ;
22     $tryChangePathLat \leftarrow true$ ;
23  end
24  if  $(tryChangePathBand) \text{ or } (tryChangePathLat)$  then
25    if  $(tryChangePathBand)$  then
26       $dswitch \leftarrow (maxbandConsum/maxbandUsable)$ ;
27      if  $(dswitch \geq 0.99)$  then
28         $maChangePathBasedOnBandwidth(maxall)$ ;
29      end
30       $tryChangePathBand \leftarrow false$ ;
31    end
32    if  $(tryChangePathLat)$  then
33      if  $(\min_m Lat > maxLatency)$  then
34        if  $((mLat1 \leq mLat2) \text{ and } (mLat1 \leq mLat3))$  then
35           $maChangePathBasedOnLatency(latency, maxband)$ ;
36        end
37      end
38      else
39         $tryChangePathLat \leftarrow false$ ;
40      end
41    end
42  end
43 end

```

---



O algoritmo (Figura 47) pega as políticas dos usuários e os coeficientes para estimar o MOS da KB. Os coeficientes são providos pelo algoritmo de aprendizagem de máquina, derivados da análise de regressão linear, considerando um contexto similar.

Inicialmente todos os fluxos estão no mesmo caminho e as variáveis *'TryChangePathBand'* e *'TryChangePathLatency'* são inicializadas com valor *false*. Em seguida o algoritmo obtém as estatísticas para cada fluxo: coleta três medidas consecutivas do *throughput*, sendo persistida a de maior valor; latência no caminho atual; o *jitter*, considerando o acúmulo de medidas anteriores da latência e a taxa de perda de pacotes do fluxo (linhas 5-9). O MOS é estimado (linha - 10). Finalmente todas essas informações são persistidas na KB (linha 11).

Para otimizar a QoE foi determinado que quando o *throughput* do fluxo atingir pelo menos 80% da largura de banda máxima configurada (linhas 12-13), haverá a possibilidade do fluxo ter sua largura de banda incrementada (*'changeRateUp'*). O incremento depende de três condições:

- Se o caminho atual do fluxo tem largura de banda disponível;
- Se o aumento em concessão é maior que o último aumento concedido;
- Se o aumento em concessão é menor do que a largura de banda máxima permitida.

Se todas as condições forem satisfeitas, o algoritmo chama o *OFQuality* e a largura de banda é aumentada em 10%.

Se a largura de banda não pode ser aumentada, então o algoritmo sugere alteração na rota com base na largura de banda (*'tryChangePathBand' = true*), na próxima iteração do loop (linhas 13-14). A mudança de rota pode ocorrer de duas maneiras diferentes:

- Se o algoritmo detectar que a latência no caminho atual é maior do que o valor da latência máxima definida na política, então, a mudança de rota será baseada em caminhos com menor latência (*'tryChangePathLat' = true*), linhas 17-18.
- Se o algoritmo detectar que a taxa de perda de pacotes no caminho atual é maior do que o valor máximo definido na política, então, a mudança de rota será baseada em caminhos com maior largura de banda disponível (*'tryChangePathBand' = true*) ou caminhos com menor latência (*'tryChangePathLat' = true*), linhas 20-22.

Embora não seja mostrado no pseudocódigo, foi determinado que

se três medidas consecutivas do *throughput* atingirem 40% ou menos do que foi configurado na política, então o algoritmo usa o limitador do fluxo, reduzindo a largura de banda em 5%. Se em três medições consecutivas do *throughput* o valor for 0 (zero), o algoritmo assume que a conexão do fluxo foi finalizada.

#### 5.5.3.4 APIs da Camada de Gerenciamento

Considerando que as aplicações da camada de gerenciamento e de controle estão situados em diferentes níveis de abstração, as mesmas se comunicam usando API *Northbound*, assim, a API REST do *Floodlight* provê a comunicação entre as aplicações de gerenciamento e módulos de controle.

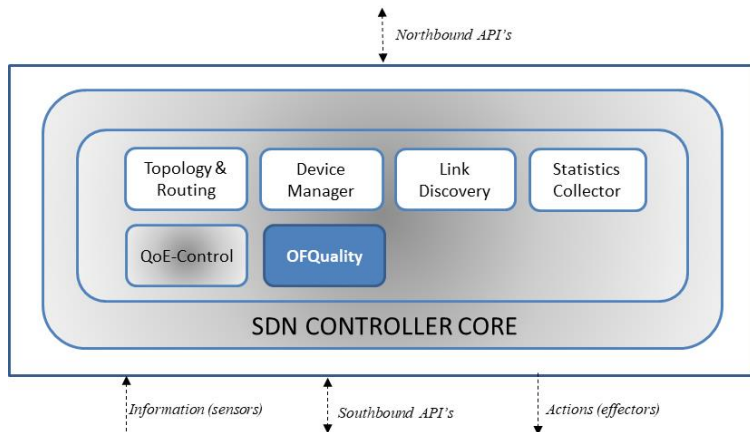
O módulos de controle *OFQuality* e *QoC-Control* implementam a REST API *Northbound*, permitindo que a aplicação de gerenciamento *QoE-ManApp* interaja com ambos. Essa API REST pode ser acessada usando métodos HTTP (ou seja, GET, POST e DELETE), em combinação com os URIs do recurso. Quando o controlador recebe uma chamada REST, o gerenciador do controlador redireciona a chamada para uma classe específica do módulo *OFQuality* ou do *QoC-Control*, que através de um método respectivo, faz a comunicação com o plano de dados.

#### 5.5.4 Camada de Controle

Provê todos os componentes para a configuração, monitoração e o controle dos recursos que provisionam e entregam o serviço cliente da QoE. São propostos e/ou utilizados módulos para configurar os parâmetros de QoS nos recursos de rede; coletar estatísticas; calcular a melhor rota de acordo com os parâmetros de QoS; controlar a admissão dos recursos. Todos esses componentes contribuem para a inteligência do controlador SDN.

A abordagem de provisão de serviços centrada no usuário, que permite que as demandas de alto nível (nível do usuário/aplicação) sejam realizadas em baixo nível (nível de hardware), é facilitada pela camada de controle. Neste caso, além do núcleo do controlador SDN e seus módulos genéricos, módulos de controle, que implementam diferentes funções de rede dão suporte ao gerenciamento da QoE. Os elementos que compõem a camada de controle são ilustrados na Figura 48.

Figura 48 - Componentes da camada de controle



Na Figura 47 os módulos de cor branca referem-se aos módulos nativos do controlador *Floodlight*, controlador ao qual a arquitetura foi projetada. O módulo de cor azul escuro (*OFQuality*) foi proposto e avaliado em nosso trabalho de pesquisa anterior (VOLPATO et al., 2017b). E finalmente, a nossa contribuição nesta camada consiste na implementação do módulo *QoE-Control*, proposto para identificar e gerenciar novos fluxos.

O *OFQuality*, o *QoEControl* e os demais módulos utilizados foram carregados no *Module Manager* do controlador *Floodlight*, descritos em um arquivo de propriedades que é lido pelo controlador durante a inicialização. Todos esses módulos e o núcleo do controlador são descritos a seguir.

#### 5.5.4.1 Núcleo do Controlador SDN

Oferece serviços de interesse de aplicações SDN e implementa vários módulos (PROJECT FLOODLIGHT, 2014), alguns deles estão sendo utilizados para o escopo deste trabalho (*Topology Manager & Routing*; *Device Manager*; *Discovery Link*). É de responsabilidade do núcleo ouvir os *sockets OpenFlow* e despachar os eventos para os módulos secundários (e.g., *OFQuality*, *QoE-Control*) que se registram para receber eventos.

**a) *Topology Manager and Routing*:** mantém as informações da topologia para o controlador e descobre as rotas disponíveis. O Serviço

de Topologia calcula topologias com base em informações de *links* que aprende do *ILinkDiscoveryService*. Todas as informações sobre a topologia atual são armazenadas em uma estrutura de dados imutável chamada instância de topologia. Se houver alguma alteração na topologia, uma nova instância é criada e a mensagem de notificação é chamada. Se outros módulos quiserem ouvir mudanças na topologia, eles podem implementar a interface *ITopologyListener* (PROJECT FLOODLIGHT, 2014).

**b) *Device Manager*:** adiciona e remove os dispositivos à medida que eles se movimentam na rede e, define o dispositivo de destino para um novo fluxo. Conhece o estado *up/down* dos *switches* e suas portas, atendendo as mensagens assíncronas (*Packet-In*, *Flow-Removed*, *Port-Status*) trocadas entre o controlador e o *switch* (PROJECT FLOODLIGHT, 2014).

**c) *Serviço de Descoberta do Link*:** é responsável pela descoberta e manutenção das ligações de dispositivos na rede *OpenFlow*. Este serviço usa o protocolo LLDP (*Data Link Discovery Protocol*) e faz *broadcast* de pacotes para detectar as ligações (PROJECT FLOODLIGHT, 2014).

**d) *Statistics Collector*:** coleta estatísticas da utilização da banda de um respectivo *switch* ou porta. Não é um módulo nativo do controlador *Floodlight*, foi disponibilizado posteriormente pela comunidade científica, publicado na documentação do desenvolvedor (PROJECT FLOODLIGHT, 2014) e disponibilizado no pacote `net.floodlightcontroller.statistics`. Para facilitar a obtenção das métricas de QoS pelo *QoS-Monitor* (aplicação de gerenciamento proposta), implementamos novos métodos neste módulo para obter estatística por fluxo e estimar a taxa de utilização do *link*.

#### 5.5.4.2 Quality of Service Configuration Module – OFQuality

O *OFQuality* (VOLPATO et. al, 2017b) é um módulo para configuração de QoS nos dispositivos do plano de dados para que aplicações (de gerenciamento e de controle) e especialistas de domínio possam criar, atualizar, remover e consultar filas e configurações de políticas de QoS nos *switches* que suportam OVSDB.

O *OFQuality* implementa dois tipos de REST APIs: *Northbound* e *Southbound* APIs. A API *Northbound* permite que outros sistemas

(e.g., *QoE-ManApp*) interajam com o módulo e, portanto, com o SDNC e *switches*. Essas APIs REST podem ser acessadas usando os métodos CRUD do HTTP em combinação com as URIs do recurso. A API *southbound* é responsável pela comunicação cliente-servidor OVSDDB (PFAFF; DAVIE, 2013), considerada a ponte entre o SDNC e o servidor OVSDDB.

O módulo *OFQuality* possui uma implementação do protocolo OVSDDB (PFAFF; DAVIE, 2013) relativa a QoS. Ele implementa um cliente que envia e recebe informações, sendo possível um cliente OVSDDB se conectar a várias instâncias de servidor OVSDDB. Cada instância de *switch* OVS possui um servidor OVSDDB que pode ser gerenciado pelo módulo *OFQuality*.

O módulo *OFQuality* conta com a funcionalidade de *cache* a fim de manter os endereços dos recursos de QoS atualizados. Para estabelecer conexões com vários *switches*, precisa-se manter um mapeamento sobre elementos/recursos de cada *switch* com o seu respectivo endereço de rede. Por isso, o módulo *OFQuality*, por padrão, mantém um *cache* que consulta as instâncias de *switches* em um tempo predeterminado. Isso garantirá a consistência das informações no módulo, permitindo localizar cada elemento (e.g. portas, interfaces, filas).

O controlador possibilita que os módulos secundários exportem serviços para outros módulos ou aplicações. Assim, o *OFQuality* exporta serviços relacionados à QoS para o controlador, bem como para outros módulos do controlador, incluindo o módulo *QoC-Control*. Para isso, definiu-se uma interface com a descrição dos métodos fornecidos como um serviço. Em seguida, implementou-se esses métodos. Finalmente, todos os módulos que utilizem o módulo *OFQuality* terão acesso a implementação e documentação.

#### 5.5.4.3 Quality of Experience Control (QoE-Control)

A *QoEControl* é uma aplicação de gerenciamento compilada como um módulo no controlador *Floodlight*. Seu principal objetivo consiste em identificar e gerenciar fluxos, selecionando um caminho de acordo com as políticas de QoS definidas na KB. Desta forma, possui características tanto de uma aplicação de gerenciamento quanto de controle. Não sendo possível sua implementação na camada de gerenciamento, justamente porque uma das funções de controle usada para auxiliar na detecção de novos fluxos (canal de recebimento de mensagens PACKET\_IN enviadas dos *switches* ao controlador), não

está disponível para aplicações. Assim, as funcionalidades deste módulo são executadas em dois níveis:

- i. Nível de controle, para a identificação de novos fluxos e seleção de caminho inicial, de acordo com as políticas da KB;
- ii. Nível de gerenciamento, no qual o gerenciamento do fluxo é realizado pela *QoE-ManApp* durante toda a provisão do serviço.

Para a operacionalização destes dois níveis, a *QoE-Control* foi estruturada conforme segue.

**a) Estrutura da Aplicação QoC-Control:** a estrutura do *QoE-Control* foi definida de acordo com a interface *IFloodlightModule*. Nesta interface, estão presentes os métodos que: (i) permitem exportar variáveis a serem usadas por outros módulos; (ii) inicializam variáveis; (iii) permitem criar *threads* para outros trabalhos do módulo; (iv) definem os módulos que se pretende acessar. Foi necessário, também, implementar a interface *IOFMessageListener* para subscrever mensagens do tipo `PACKET_IN`. A interface *IOFMessageListener* tem um método *receive* para que o *QoC-Control* ao receber uma mensagem `PACKET_IN` entregue pela classe *Controller*, a trate de forma conveniente. O método espera como retorno um comando `CONTINUE` ou `STOP` para que o controlador propague o pacote para os próximos módulos interessados ou pare a sua propagação.

**b) Funcionalidades da Aplicação QoC-Control:** a operacionalização dos dois níveis da *QoE-Control* foi implementada com o recurso de multi *threads*, sendo uma *thread* principal para o controle e *threads* secundárias para o gerenciamento dos fluxos. Essa forma de operacionalização foi avaliada em nosso trabalho de pesquisa anterior (VOLPATO et al., 2017c), porém todo o controle estava sendo realizado na mesma camada, ou seja, na camada de controle.

(i) *Thread principal:* de acordo com a estrutura da *QoC-Control* definida anteriormente, a *thread* principal recebe do controlador mensagens `PACKET_IN`, enviadas ao controlador pelos *switches*. Analisando as informações das mensagens recebidas, insere fluxos de protocolos como ARP, ICMP, TCP ou UDP no *switches*, afim de prover a comunicação entre os *hosts*. A *thread* principal analisa a KB e se existir política para o fluxo em questão (TCP ou UDP), inicializa-se uma *thread* secundária com as informações pertinentes de cada fluxo,

caso contrário, os fluxos são descartados.

(ii) *Threads secundárias*: são geradas assim que os fluxos (TCP ou UDP) forem detectados pela *thread* principal. Cada *thread* secundária é responsável por monitorar um determinado fluxo e o acompanha durante todo o seu tempo de vida. O monitoramento do fluxo pela *thread* secundária acontece através de um *loop* do tipo *while true*. As *threads* secundárias são gerenciados pela *QoE-ManApp*.

**c) Tratamento dos Fluxos pela QoE-Control**: o tratamento dos fluxos feito pela *QoE-Control* é realizado de acordo com o proposto em nosso trabalho de pesquisa anterior, Volpato et al. (2017b), ou seja, a aplicação associa um fluxo com uma fila. Os parâmetros iniciais utilizados na fila são previamente definidos conforme a informação da KB. A fila é então inserida numa configuração de QoS que estará em todas as portas dos *switches* por onde o fluxo precisar passar. Havendo modificação da rota do fluxo, a aplicação remove a fila das portas dos *switches* no caminho anterior e insere nas portas do novo caminho.

#### 5.5.4.4 APIs da Camada de Controle

Na camada de controle são usadas duas APIs do tipo *Southbound* para prover a comunicação com o plano de dados. Primeiro, para fazer o gerenciamento das regras das tabelas de fluxo, o controlador se comunica com os *switches* do plano de dados através do protocolo *OpenFlow*. Em segundo lugar, para configurar os recursos de QoS dos *switches*, o controlador usa a aplicação *OFQuality*, que se comunica com o plano de dados através do protocolo OVSDDB.

### 5.5.5 Camada de Dados SDN

A quinta camada compreende a infraestrutura de dispositivos (*switches* virtuais e/ou físicos) que operam com a filosofia de redes definida por software, o que lhes confere independência de fornecedor, interoperabilidade e programação das tabelas de encaminhamento sob demanda. Essa camada é composta pelos dispositivos de rede com os protocolos *OpenFlow* e OVSD habilitados.

O controlador usa a API *OpenFlow* para instalar regras de encaminhamento nas tabelas de fluxo dos *switches*, descobrir a topologia de rede, gerar estatísticas dos fluxos, visualizar a topologia da rede, descobrir o status *up/down* dos dispositivos, controlar os fluxos

ativos nos dispositivos.

O OVSDB é responsável pela configuração dos recursos do *switch* (por exemplo, pontes, interfaces, portas, filas, políticas QoS e coleta de estatísticas). Os pacotes de rede pertencentes a um fluxo podem ser enfileirados em uma fila particular da interface de saída de um *switch*. Essas filas são previamente criadas com o módulo *OFQuality* (VOLPATO et al., 2017b) que contém uma implementação do protocolo OVSDB.

## 5.6 CONSIDERAÇÕES DO CAPÍTULO

Neste capítulo foi apresentado o modelo de gerenciamento de QoE, composto pela taxonomia, pelo modelo de representação do conhecimento e pela arquitetura de provisão de serviços.

A taxonomia, composta por quatro dimensões (fatores humanos, contexto, conteúdo, tecnológica) foi definida com base nos trabalhos e propostas existentes na literatura, apresentados no capítulo 3. As dimensões evidenciadas permitiram identificar o quanto o gerenciamento da QoE é complexa, uma vez que existem muitos fatores que podem impactar a UX no uso de um serviço.

A análise da taxonomia permitiu observar que nem todas as dimensões se aplicam ao mesmo propósito e irão depender do tipo de serviço que se pretende avaliar, ou seja, os fatores de influência de um serviço VoIP, por exemplo, diferem de um serviço de buscador de hotéis, pois o sucesso na UX do VoIP está condicionado a QoS do sistema de comunicação, enquanto que o sucesso da UX ao usar o buscador de hotéis, pode estar atrelada a ergonomia e usabilidade do sistema; a qualidade do conteúdo disponibilizado (e.g. atual, preciso, significado e valor do conteúdo para o usuário); tempo necessário para o usuário completar a tarefa (encontrar um hotel com as características desejadas); tempo de resposta do sistema para localizar o hotel com as restrições do usuário.

A partir da taxonomia, o modelo de representação do conhecimento foi proposto e formalizado com o uso de ontologia. O modelo foi construído, usando o processo metodológico proposto por Rautenberg e Todesco (2009), composto pelas etapas de especificação, aquisição do conhecimento, implementação e verificação.

Buscando inspiração na metodologia de desenvolvimento ágil, inicialmente um número limitado de questões de competências foi especificado e uma primeira iteração da ontologia foi criada para avaliar a QoE do usuário no uso de um serviço e gerar alertas de violações de



políticas. Posteriormente o *framework* de referência foi revisitado, novas questões de competências foram especificadas e novas iterações da ontologia foram produzidas.

Para a avaliação do modelo de representação do conhecimento da QoE foi sugerido a incorporação do mesmo numa plataforma de prestação de serviços orientada as necessidades dos usuários. Para isso foi apresentada uma arquitetura conceitual, que permite uma abordagem voltada para atender a UX. A arquitetura, estruturada em cinco camadas (contexto, serviço, gerenciamento, controle e dados) segue o paradigma da SDN (ONF, 2012), uma das promessas para a provisão de serviços na Internet do Futuro.

A contribuição desta tese para com a arquitetura consistiu, principalmente, da aplicação de gerenciamento da QoE (*QoE-ManApp*), projetada com o *loop* de controle do MAPE-K (IBM, 2005), incorporada na camada de gerenciamento, no módulo de Gerenciamento da QoE. A aplicação interage com as aplicações de gerenciamento e de controle, com o objetivo de aprender a experiência do usuário no uso de um serviço, verificar violações de políticas de QoS e autonomicamente solicitar à uma aplicação de controle a restauração da QoE.

Após a apresentação das três contribuições para o gerenciamento da QoE (taxonomia da QoE, modelo de representação do conhecimento e a arquitetura), incluindo o motor semântico proposto, conclui-se que, o segundo objetivo foi atingido completamente. O terceiro objetivo foi atendido completamente, pois todas as questões de competências foram respondidas. O quarto objetivo também foi contemplado na sua totalidade, uma vez que o modelo de representação do conhecimento foi inserido na aplicação de gerenciamento da QoE. A incorporação do modelo na camada de gerenciamento da QoE, permitiu o compartilhamento do conhecimento usado pelos mecanismos da SDN (aplicações de gerenciamento e controle). O conhecimento disponível da KB, juntamente com a base de regras possibilitou a classificação dos alertas das degradações da QoE e assim, as funcionalidades MAPE-K (da aplicação de gerenciamento da QoE) atuaram de forma autônoma para restaurar a QoE.

No próximo capítulo é feita a avaliação do modelo proposto aplicável em um cenário de Ambiente de Vida Assistido.



## 6 AVALIAÇÃO DO MODELO DE GERENCIAMENTO DA QOE

Neste capítulo o modelo de gerenciamento da QoE é avaliado. Para verificar a sua viabilidade é apresentado um cenário de provisão de serviços de *eHealth* em um AAL (*Ambient Assisted Living*), usando uma infraestrutura de rede definida por software. A avaliação consiste em verificar se o mecanismo de inteligência (aplicação de gerenciamento da QoE) é capaz de detectar, planejar e desencadear ações para restaurar a QoE, usando o conhecimento disponível na KB.

### 6.1 CENÁRIO EXPERIMENTAL

O cenário consiste de um AAL onde um paciente asmático é assistido remotamente por cuidadores da saúde. Com frequência, o paciente apresenta crises asmáticas e é necessário monitorar os dados as condições ambientais e sinais vitais.

Para o sensoriamento do AAL são monitorados os valores da temperatura, umidade e partir destes, calculado o ponto de orvalho (ver seção 5.1.2). No que se refere aos sinais biomédicos são feitas leituras da temperatura corporal, pressão arterial (sistólica e diastólica), pulso e oxigênio no sangue. Os dados sensoriados são enviados para uma base de dados do provedor de serviço de *eHealth*. Sempre que necessário o paciente é atendido por teleconsulta e havendo necessidade, imagens médicas do paciente são enviadas para o centro de atendimento remoto.

Através da descrição do cenário observa-se que na provisão do serviço de *eHealth* podem ser utilizados quatro variações do serviço (monitoramento de sinais vitais, monitoramento das condições ambientais, teleconsulta e envio de imagens médicas). Levando em consideração que as informações são críticas, há a necessidade de acompanhar, com frequência, a qualidade do serviço provido, assim, a Aplicação de Gerenciamento da QoE (*QoE-ManApp*) será utilizada para avaliar o nível de satisfação do serviço, considerando as políticas de QoS definidas na KB. Percebendo anormalidades na provisão do serviço, o sistema aciona as aplicações de gerenciamento e controle da rede SDN para restaurar a QoE.

#### 6.1.1 Correlações da Temperatura e Umidade Relativa do Ar com Crises Asmáticas

Segundo Teixeira et al. (2012), desde os primórdios a Umidade Relativa (UR) do ar tem sido relacionada a problemas de saúde, devendo

ficar entre 40% a 70%. Em seus estudos, Collins (1986) evidenciou que tanto a UR baixa quanto a alta podem contribuir para o aparecimento de doenças respiratórias, incluindo a asma.

Os fatores que podem desencadear ou agravar os sintomas da doença incluem infecções virais, alérgicos domésticos ou profissionais (e.g., ácaros, pólenes, barata), o fumo do tabaco, exercício e estresse (GIMA, 2015). Além destes, os parâmetros meteorológicos (e.g., umidade relativa do ar, precipitação, temperatura do ponto de orvalho, horas de sol e temperatura de bulbo seco) têm sido estudados como indicadores da frequência de asma brônquica (FLEISCHER, 1978).

Para a OMS (Organização Mundial da Saúde) a asma é uma das doenças respiratórias que tem sido apontadas como problema mundial de saúde pública. Motivados por este indicativo iniciamos nossas pesquisas com a idealização de um cenário AAL aplicável ao contexto em questão (SILVA et al., 2016b).

### 6.1.2 Correlação da Temperatura do Ponto de Orvalho como Indicador de Desconforto Térmico Humano

“O ponto de orvalho refere-se à temperatura na qual o vapor de água presente no ar ambiente passa ao estado líquido na forma de pequenas gotas por via da condensação”<sup>7</sup>. Ele pode ser usado como indicador de desconforto térmico fazendo-se uma relação com a temperatura, pressão e a UR.

A Tabela 19 apresenta os níveis de sensação no ser humano em função do cálculo da temperatura de orvalho, levando em consideração a UR a uma temperatura ambiente de 32° C.

Tabela 19 - Sensação térmica em função da temperatura do ponto de orvalho

Ponto de Orvalho em °C	Sensação Térmica no ser Humano	UR a 32°C	Sensação Térmica dos 32°C com UR
> 29 °C	Extremamente sufocante. Casos de morte por hipertermia	> = 85%	54°C-47°C
26-28 °C	Altíssima. Mortal para doenças relacionadas à asma	84-73%	46°C-42°C
24-26 °C	Extremamente desconfortável, consideravelmente sufocante	72-62%	41°C-38°C
21-24 °C	Muito úmido, desconfortável	61-52%	37°C-35°C
18-21 °C	Desconfortável para a maioria das pessoas no limite superior	51-44%	35°C-33°C
16-18 °C	Aceitável para a maioria, porém todos percebem a umidade no limite superior	43-37%	32°C

<sup>7</sup> [https://pt.wikipedia.org/wiki/Ponto\\_de\\_orvalho](https://pt.wikipedia.org/wiki/Ponto_de_orvalho)

Ponto de Orvalho em °C	Sensação Térmica no ser Humano	UR a 32°C	Sensação Térmica dos 32°C com UR
13-16°C	Confortável	36-31%	32°C-31°C
10-12°C	Muito confortável	30-26%	31°C-30°C
< 10°C	Um pouco seco para algumas pessoas	<=25%	30°C-27°C

Fonte: Horstmeyer (2006)

A Tabela 20 apresenta as equações necessárias para obter o índice de desconforto térmico e demais parâmetros usados no sensoriamento dos sinais ambientais (SILVA et al., 2016b).

Tabela 20 - Equações para obtenção dos valores das variáveis meteorológicas

Cálculo	Equações	Variáveis	Autores
Índice de Desconforto Térmico (ID)	$ID = 0.99 \times Ta + 0.36 \times To + 41.5$ (1)	ID = índice de desconforto Ta = temperatura em graus °C To = temperatura de orvalho em °C	Horstmeyer (2006), adaptado de (Thom 1959)
Temperatura do Ponto de Orvalho (To) em °C	$To = \frac{B_1 \times \left[ \ln\left(\frac{Ur}{100}\right) + \left(\frac{A_1 \times Ta}{B_1 + Ta}\right) \right]}{A_1 - \ln\left(\frac{Ur}{100}\right) - \left(\frac{A_1 \times Ta}{B_1 + Ta}\right)}$ (2)	A1 = 17,625 B1 = 243,04 Ln = log2,718 Ur = umidade relativa	Lawrence (2005)
Umidade Relativa (UR) em %	$Ur = 100 \times \left(\frac{e}{e_s}\right)$ (3)	e = pressão atual de vapor es = pressão saturada de vapor	Lawrence (2005)
Pressão Saturada do Vapor (es)	$e_s = E \times 10^{\left(\frac{7,5 \times Ta}{237,3 + Ta}\right)}$ (4)	E = parâmetro equivalente a 0,6108 para resultados em kilopascal (kPa) e Ta em graus °C.	Lier et al. (2008)

Fonte: Silva et al. (2016)

### 6.1.3 Parâmetros e Valores de Referência dos Sinais Vitais

No AAL em estudo foram transmitidos dados do monitoramento dos seguintes sinais vitais: temperatura corporal, pulso e oxigênio no sangue e pressão arterial. Para cada sinal vital existe uma terminologia e um valor de referência para inferir sua classificação, conforme sumarizado na Tabela 21.

Tabela 21 - Terminologia, classificação e valores de referência dos Sinais vitais

Sinal Vital	Terminologia e/ou Classificação	Valores de Referência
Temperatura	Hipotermia: Temperatura abaixo de 35°C Afebril: 36,1°C a 37,2°C Febril: 37,3°C a 37,7°C Febre: 37,8°C a 38,9°C Pirexia: 39°C a 40°C Hiperpirexia: acima de 40°C	Temperatura axilar: 35,8°C a 37°C Temperatura bucal: 36,3°C a 37,4°C Temperatura retal: 37°C a 38°C
Pulso e Oxigênio no	Pulso normocárdico: Batimento cardíaco normal. Pulso rítmico: os intervalos entre os batimentos são	Adultos – 60 a 100 bpm.

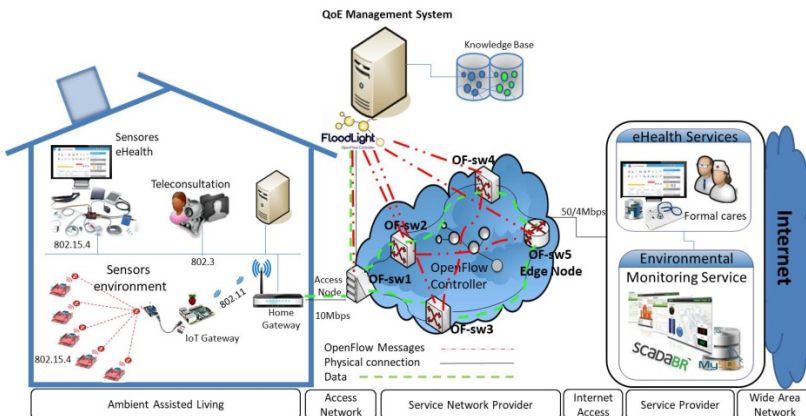
<i>Sinal Vital</i>	<i>Terminologia e/ou Classificação</i>	<i>Valores de Referência</i>
Sangue	iguais. Pulso arritmico: os intervalos entre os batimentos são desiguais. Pulso dierótico: dá impressão de dois batimentos. Taquisfigmia: pulso acelerado. Bradisfigmia: frequência abaixo da faixa normal. Pulso filiforme: indica redução da força ou do volume do pulso periférico	Crianças – 80 a 120 bpm. Bebês – 100 a 160 bpm.
Pressão Arterial (PA)	Hipertensão: PA acima da média. Hipotensão: PA inferior à média. Convergente: a sistólica e a diastólica se aproximam. Divergente: a sistólica e a diastólica se afastam.	Hipotensão – inferior a 100 x 60. Normotensão – 120 x 80. Hipertensão limite – 140 x 90. Hipertensão moderada – 160 x 100. Hipertensão grave – superior a 180 x 110.

Fonte: <http://aenfermagem.com.br/procedimentos/sinais-vitais/>

## 6.2 AMBIENTE EXPERIMENTAL

Esta seção apresenta o ambiente usada para realizar os experimentos, incluindo a infraestrutura física e todos os recursos utilizados para emular a comunicação do AAL com a unidade de atendimento remota. A infraestrutura física com seus componentes é apresentada na Figura 49.

Figura 49 - Infraestrutura utilizada nos experimentos



No ambiente experimental está sendo considerado que toda infraestrutura para a provisão, gerenciamento, transporte e

armazenamento de dados é fornecida e gerenciada pelo provedor de serviços. Essa prática tem sido adotada em grandes corporações e em diferentes contextos de aplicações, conforme relatado em Jain et al. (2013) e Philips (2015). Nossa primeira contribuição de aplicação da SDN em *eHealth*, apoiada pela *Philips Health Care* pode ser consultada em Assuiti et al. (2016).

Para emular a comunicação remota de um AAL com um centro de atendimento remoto, alguns serviços e recursos foram configurados e incorporados num ambiente local, conforme segue.

#### 6.2.1.1 Configuração dos Serviços no AAL

Para a transmissão das condições ambientais foi configurada uma rede de sensores *zigBee*, com 11 nodos escravos e 1 mestre. O mestre captura os dados dos sensores DHT-11, via comunicação serial, e os envia ao gateway IoT (*Internet of Things*). Toda a estrutura da rede para obter as leituras das medições e a troca de mensagens entre coordenador e os nodos *ZigBee* foram avaliadas em Silva et al. (2015b).

Para a captura dos sinais biomédicos utilizou-se a plataforma de *eHealth* (Cooking-hacks). Usando uma aplicação de *eHealth* (NAZÁRIO; DANTAS, 2014) os dados são capturados e transmitidos via comunicação serial do terminal do usuário para o *Home Gateway*. Inicialmente os dados foram enviados pela plataforma de *eHealth* (SILVA et al., 2016c) posteriormente foram gerados dados sintéticos, emulando medições de 11 sensores para cada tipo de sinal biomédico (temperatura, pressão arterial, pulso e oxigênio no sangue).

Para a emulação do envio de imagens foi utilizada uma série de arquivos com características similares de imagens médicas.

#### 6.2.1.2 Home Gateway

Foi usado o roteador TP-LINK TL-WR1043ND para conectar com o *access node* na rede do ISP (*Internet Service Provider*), emulando a conexão com a Internet através de um *link* banda larga (DSL, Cable, or PON) de 10Mbps.

### 6.2.2 Provedor do Serviço e Provedor de Rede

O provedor de rede e de serviços tratam-se da mesma entidade. Para emular o provedor de rede foi usada uma malha de *switches OpenFlow*, instalado o SDNC *Floodlight* e os principais módulos das

camadas de gerenciamento e de controle, apresentadas na arquitetura proposta. Estamos considerando que o provedor de serviço de *eHealth* provê e gerencia toda a infraestrutura física.

#### 6.2.2.1.1 Malha de Switches OpenFlow

Foram utilizados cinco OVS (*Open Virtual Switch*) instalados, inicialmente, no sistema operacional Ubuntu 14.04-64bits<sup>8</sup>. O OVS-1 foi usado para emular o dispositivo de acesso ao ISP. Cada OVS é executado em uma máquina física separada interligado por túneis GRE (*Generic Routing Encapsulation*). O comando *tc* do Linux foi usado para configurar as restrições do *delay*, *jitter*, percentual de pacotes descartados e capacidade do *link*. Para acessar a Internet foi utilizado um *link* de 50/4Mbps.

#### 6.2.2.1.2 Sistema de Gerenciamento da QoE

É composto pela KB, as aplicações de gerenciamento, *QoS-Monitor*, *QoS-Router* e *QoE-ManApp* e os módulos de controle, *OFQuality* e *QoE-Control*. Em resumo, o sistema de gerenciamento da QoE representa toda a proposta apresentada no capítulo 5, com a programação dos componentes no controlador *Floodlight*.

### 6.2.3 Protocolos de Criptografia e Roteamento

Para o envio dos dados do AAL para o provedor de serviço foi configurada uma VPN (*Virtual Private Network*) usando protocolos de criptografia e autenticação. Nos experimentos foi utilizado o protocolo de roteamento OSPF (*Open Shortest Path First*), nativo do *Floodlight*.

Para o envio dos dados foram configuradas três rotas alternativas. Na rota 1, os dados são enviados pelos *switches* s1-s2-s4, na rota 2 pelos *switches* s1-s3-s4 e na rota 3 pelos *switches* s1-s5-s4. A Tabela 22 apresenta a configuração dos *links* utilizados nos experimentos.

Tabela 22 - Configuração das rotas dos *Links* usadas nos experimentos

<i>Rota</i>	<i>Origem</i>	<i>Destino</i>	<i>Delay (ms)</i>	<i>Perda (%)</i>	<i>Capacidade do Link</i>
1	S1	S2	50	2	10 Mbps
	S2	S4	50	2	10 Mbps

<sup>8</sup> Ao longo dos experimentos o sistema operacional foi atualizado para o Ubuntu 16.04.2 LTS.



<i>Rota</i>	<i>Origem</i>	<i>Destino</i>	<i>Delay (ms)</i>	<i>Perda (%)</i>	<i>Capacidade do Link</i>
2	S1	S3	10	1	10 Mbps
	S3	S4	10	1	10 Mbps
3	S1	S5	100	0	10 Mbps
	S5	S4	100	0	10 Mbps

### 6.2.3.1.1 Parâmetros e Valores de Referência para as Políticas de QoS

Uma vez que os experimentos foram realizados em laboratório, não dispúnhamos de parâmetros de QoS de referência, as políticas de QoS para os serviços de *eHealth* foram definidas com base em pesquisas de vários autores, principalmente de Skorin-Kapov e Matijasevic (2010), sendo as mesmas apresentadas na Tabela 23.

Tabela 23 - Resumo dos requisitos de QoS dos serviços de e-Health.

<i>Tipo de Serviço de eHealth</i>	<i>Exemplo de Aplicação</i>	<i>Principal Conteúdo Midiático</i>	<i>Requisitos de QoS</i>		
			<i>Throughput</i>	<i>Delay</i>	<i>Perda</i>
Teleconsulta baseada em vídeo conversacional em tempo real.	Vídeo conferência paciente e médico para um exame de rotina	Vídeo	Média (4Mbps)	250 – 400ms	até 1%
Telemonitoramento em tempo não real.	Transmissão de sinais vitais de pacientes para atendimento domiciliar	Dados biomédicos coletados dos sensores.	Baixo (1,0 a 2,0Mbps)	300-500ms	0%.
Telemonitoramento em tempo real.	Transmissão de sinais ambientais	Dados ambientais coletados dos sensores.	Baixo (2,0 a 2,5Mbps)	<300ms	Até 2%
Telediagnóstico em tempo não real	Diagnóstico remoto não emergencial: Transferências de imagens médicas para um local remoto, onde especialistas analisam e retornam um relatório diagnóstico.	Imagens, texto e dados.	Média, mas, depende do tipo de imagem (3Mbps)	250 – 400ms	até 1%

Fonte: Adaptações de Cavaro-Menard, Zhang-Ge e Callet (2013); Skorin-Kapov e Matijasevic (2010); Ullah, Fiedler e Wac (2004); Vouyioukas, Maglogiannis e Komnakos (2007).

Após a análise da Tabela 23 foram definidas as políticas de QoS para os serviços de *eHealth* (sinais biomédicos em tempo real, sinais ambientais em tempo real, imagens médicas), ilustradas na Tabela 24. A

prioridade foi definida pelo tipo de transmissão, no caso os sinais ambientais receberam maior prioridade por serem transmitidos em tempo real. Não foi definida política para o *Jitter*, pois não encontramos parâmetros de referência na literatura.

Neste experimento foram gerados quatro tipos de fluxos que simulam os serviços de *eHealth* (sinais biomédicos, sinais vitais, transferência de imagens médicas e teleconsulta), sendo uma conexão ativa para cada serviço. As políticas de QoS configuradas para os experimentos são dispostas na Tabela 24.

Tabela 24 - Políticas de QoS para os serviços de eHealth

Id do Fluxo	Protocolo Transporte	Serviço	Prioridade	Principal Conteúdo Midiático	Throughput (Mbps)		Latência (ms)		Perda de Pacote (%)	
					min	max	min	max	min	max
1	UDP	Sinais Biomédicos	1	Dados	1,0	2,0	300	500	0	0
2	UDP	Sinais Ambientais	2	Dados	1,5	3,0	0	200	0	2
3	TCP	Imagens Médicas	3	Imagem	3,0	5,0	200	400	0	1
4	UDP	Teleconsulta	4	Áudio e vídeo	3,0	5,0	200	400	0	1

### 6.3 RESULTADOS EXPERIMENTAIS

Neste banco de ensaio foram realizados dois experimentos, com duas rodadas, com e sem a ativação do algoritmo de gerenciamento da QoE, com três objetivos:

- a) Analisar a estatística descritiva e intervalos de confiança dos parâmetros monitorados.
- b) Verificar e comparar estatisticamente o desempenho das duas abordagens, ou seja, com e sem o algoritmo de gerenciamento de QoE.
- c) Verificar as funcionalidades do algoritmo na restauração da QoE.

Após as rodadas foram coletadas 12 amostras (para cada abordagem), composta por  $n$  observações, sendo que o número de observações varia em função do tipo de serviço, pois as conexões iniciaram em tempos distintos. O número da amostra se justifica pelo fato de estarmos experimentando quatro serviços (sinais ambientais, sinais vitais, imagens médicas, teleconsulta), considerando que cada serviço tem 3 parâmetros de QoS (*throughput*, latência, descarte de

pacotes), logo foi necessário analisar um conjunto de dados composto de 12 amostras com o algoritmo proposto e 12 amostras com o algoritmo nativo.

Para todo o conjunto de dados observado foi realizada a mesma análise estatística, contemplando a estatística descritiva, intervalos de confiança e testes de hipóteses.

### 6.3.1 Análise Descritiva dos Dados

Nas Tabelas 25 a 36 são sumarizadas as estatísticas descritivas dos dados, com o algoritmo proposto (*QoEManApp*) e com algoritmo nativo do SDNC *Floodlight*. Os dados das duas abordagens são comparados, fazendo referência aos valores mínimos e máximos definidos na política de QoS (Tabela 24).

#### 6.3.1.1 Análise descritiva dos Sinais Ambientais

Tabela 25 - Estatística descritiva do throughput: sinais ambientais

<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Mínimo	1,5	Mínimo	1,5
Média	2,102068966	Média	1,500344828
Mediana	2,06	Mediana	1,5
Máximo	2,69	Máximo	1,51
Desvio Padrão da Média	0,045987735	Desvio Padrão da Média	0,000344828
Desvio Padrão	0,247651531	Desvio Padrão	0,001856953

Na Tabela 25 observa-se que a medida do mínimo da métrica *Throughput*, em ambos os casos, ficou com o mesmo valor, sendo preservado, portanto, o valor mínimo estabelecido na política de QoS (1,5Mbps). Já os valores da média, da mediana e o máximo observados são superiores quando o algoritmo de gerenciamento de QoE é ativado (*QoEManApp*). No entanto, quando o algoritmo está desativado apresenta melhor desvio padrão. O fato é justificado porque o serviço é provido (praticamente sempre) com o mesmo valor do *Throughput* (1,5Mbps), portanto, não houve variabilidade do *Throughput*. Ao passo que o algoritmo proposto foi programado para aumentar a banda dos fluxos para se aproximarem do valor máximo contratado (3,0Mbps), sempre que houver banda disponível.

Tabela 26 - Estatística descritiva da latência: sinais ambientais

<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Mínimo	2	Mínimo	3
Média	1163,172414	Média	5256,758621
Mediana	9	Mediana	8065
Máximo	5792	Máximo	9092
Desvio Padrão da Média	411,4804285	Desvio Padrão da Média	711,357778
Desvio Padrão	2215,889922	Desvio Padrão	3830,778871

Embora algumas medições da latência, com o algoritmo da *QoEManApp*, tenham ficado acima do limite máximo permitido na política de QoS, o resumo descritivo dos dados estatísticos da Tabela 26 demonstra que o algoritmo proposto é superior em todas as medidas observadas, portanto os sinais ambientais são providos com menores latências (atrasos), quando comparados com a provisão dos sinais com o algoritmo nativo. Essa melhoria se justifica porque a *QoEManApp* ao detectar valores altos para a latência tomou alguma ação de melhoria para prover o serviço com menor latência, conforme políticas estabelecidas na Tabela 18.

Tabela 27 - Estatística descritiva da perda de pacotes: sinais ambientais

<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Mínimo	0,01	Mínimo	0,28
Média	0,077241379	Média	0,293103448
Mediana	0,02	Mediana	0,28
Máximo	0,35	Máximo	0,55
Desvio Padrão da Média	0,020849546	Desvio Padrão da Média	0,009312625
Desvio Padrão	0,112278243	Desvio Padrão	0,050150021

O resumo descritivo do percentual de descarte de pacotes apresentado na Tabela 27 demonstra que a variabilidade dos dados, com o algoritmo nativo, é menor quando comparado com o algoritmo proposto, porém a taxa de perda de pacotes é inaceitável para o valor mínimo (28%), mediana (28%) e máximo (55%). Fica evidente, portanto, que o algoritmo da *QoEManApp* realiza um controle mais efetivo no gerenciamento de descarte de pacotes, pois conforme observado na Tabela 27, tanto a medida do mínimo (1%) quanto a

mediana (2%) estão dentro dos limites definidos na política de QoS; a exceção ficou no valor da média (7%), em função do elevado valor máximo (35%).

O valor máximo do percentual de descarte é justificado pela troca de rota do fluxo, ou seja, como o algoritmo da *QoEManApp* foi programado para aumentar a largura de banda dos fluxos, chega um momento em que toda largura de banda disponível é alocada, logo para obtenção de mais largura de banda para um fluxo, o algoritmo sugere a troca de rota. E no momento em que a ação de adaptação de política é sugerida (Tabela 18) faz se necessário configurar e alocar o caminho da disponibilidade do recurso, nesse ínterim até o reestabelecimento da conexão, ocorre o descarte de pacotes.

### 6.3.1.2 Análise descritiva dos Sinais Biomédicos

Tabela 28 - Estatística descritiva do throughput: sinais biomédicos

<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Mínimo	1,01	Mínimo	1
Média	1,548888889	Média	1
Mediana	1,54	Mediana	1
Máximo	2	Máximo	1
Desvio Padrão da Média	0,043107014	Desvio Padrão da Média	0
Desvio Padrão	0,223990613	Desvio Padrão	0

Conforme apresentado na Tabela 28, a medida do mínimo da métrica *Throughput*, nos dois algoritmos, ficou com o mesmo valor, sendo preservado, portanto, o valor mínimo definido na política de QoS (1,0Mbps). Já os valores da média, da mediana e o máximo observados são superiores quando o algoritmo de gerenciamento de QoE está ativo. Contudo o algoritmo nativo do SDNC *Floodlight* não apresentou variabilidade nos dados, justamente porque o serviço é provido em uma taxa constante de 1,0Mbps. Em contraste, o algoritmo proposto foi programado para aumentar a banda de acordo com a priorização dos fluxos, sempre que houvesse banda disponível; atuando desta forma, observa-se que em algum momento, algum fluxo, foi provido com o valor máximo contratado, ou seja, 2,0Mbps.

Tabela 29 - Estatística descritiva da latência: sinais biomédicos

<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Mínimo	2	Mínimo	3
Média	1048,444444	Média	5308,814815
Mediana	9	Mediana	8065
Máximo	5792	Máximo	9091
Desvio Padrão da Média	402,3541421	Desvio Padrão da Média	708,2010335
Desvio Padrão	2090,69345	Desvio Padrão	3679,920516

Embora algumas medições da latência, com o algoritmo ativo, tenham ficado acima do limite máximo permitido na política de QoS, o resumo descritivo dos dados estatísticos da Tabela 29 demonstra que o algoritmo proposto é superior em todas as medidas observadas, portanto o serviço de sinais biomédicos é provido com latências (atrasos) bem menores, quando comparado com o algoritmo nativo do SDNC *Floodlight*. Este fato comprova que as ações de adaptações de políticas planejadas pela *QoEManApp* (Tabela 18) são efetivas.

Tabela 30 - Estatística descritiva da perda de pacotes: sinais biomédicos

<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Mínimo	0,01	Mínimo	0,35
Média	0,105185185	Média	0,365925926
Mediana	0,03	Mediana	0,36
Máximo	0,41	Máximo	0,59
Desvio Padrão da Média	0,025230393	Desvio Padrão da Média	0,008791009
Desvio Padrão	0,13110097	Desvio Padrão	0,045679423

O percentual de descarte de pacotes sintetizado na Tabela 30 demonstra que a variabilidade dos dados do algoritmo nativo é menor, quando comparada com a variabilidade dos dados do algoritmo proposto, porém com o algoritmo nativo, a taxa de perda de pacotes é inaceitável para o valor mínimo (36%), mediana (36%) e máximo (59%). Por outro lado, com o algoritmo proposto, as taxas de descarte de pacotes para o mínimo (1%) e para a mediana (3%) são bem menores; a média (10%) e o máximo (41%) ficaram com valores altos, se comparados com a política de QoS contratada. A elevação destes valores (conforme já explicado na seção 5.3.1.1) é justificada pela troca

de rota sugerida pelo algoritmo proposto. Fica evidente, portanto, que o algoritmo proposto realiza um controle mais efetivo no gerenciamento de descarte de pacotes do que o algoritmo nativo do SDNC *Floodlight*.

### 6.3.1.3 Análise descritiva na Transferência de Imagens Médicas

Tabela 31 - Estatística descritiva do throughput: imagens médicas

<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Mínimo	3,01	Mínimo	3
Média	4,632608696	Média	3,026521739
Mediana	4,88	Mediana	3
Máximo	5,18	Máximo	3,52
Desvio Padrão da Média	0,118452167	Desvio Padrão da Média	0,022454246
Desvio Padrão	0,568076638	Desvio Padrão	0,10768678

O resumo descritivo da Tabela 31 permite observar que a medida do mínimo da métrica *Throughput*, em ambos os algoritmos, ficou com o mesmo valor, sendo preservado, portanto, o valor mínimo estabelecido na política de QoS (3Mbps). No entanto, os valores da média, mediana e máximo, são bem superiores com o algoritmo proposto.

Embora o algoritmo nativo tenha apresentado menor variabilidade nos dados, as imagens médicas são providas sempre na mesma taxa de transmissão, independente da prioridade do fluxo. Diferentemente, o algoritmo proposto foi programado para incrementar a largura de banda dos fluxos para se aproximarem do valor máximo contratado (5,0Mbps). Nesse aspecto, observa-se que o algoritmo foi efetivo, prova disso é o valor máximo observado (5Mbps), atingindo o definido na política de QoS.

Tabela 32 - Estatística descritiva da latência: imagens médicas

<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Mínimo	2	Mínimo	3000
Média	528,1304348	Média	5489521,739
Mediana	9	Mediana	8066000
Máximo	5790	Máximo	9091000
Desvio Padrão da Média	346,6006992	Desvio Padrão da Média	828239,5919
Desvio Padrão	1662,238559	Desvio Padrão	3972097,544

Embora algumas medições da latência, com o algoritmo proposto, tenham ficado acima do limite máximo permitido na política de QoS, o resumo descritivo dos dados estatísticos da Tabela 32 demonstra que o algoritmo proposto é efetivo no gerenciamento da latência dos fluxos, uma vez que em todas as medidas observadas (mínimo, mediana, máximo, desvio padrão), as imagens médicas são transferidas com menores latências (atrasos), quando comparadas com as transferências providas pelo algoritmo nativo do SDNC *Floodlight*. Portanto, as ações de melhorias nos elementos de rede, planejadas pela *QoEManApp* (Tabela 18) são de fato executadas pelas aplicações de controle.

Tabela 33 - Estatística descritiva da perda de pacotes: imagens médicas

<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Mínimo	0,15	Mínimo	0,15
Média	0,207391304	Média	0,227826087
Mediana	0,17	Mediana	0,19
Máximo	0,5	Máximo	0,5
Desvio Padrão da Média	0,018060257	Desvio Padrão da Média	0,018554018
Desvio Padrão	0,08661395	Desvio Padrão	0,088981945

O resumo descritivo do percentual de descarte de pacotes, apresentado na Tabela 33, demonstra que os dois algoritmos se comportaram de modo bem similar no gerenciamento de descarte de pacotes para transferir as imagens médicas. A similaridade observada nos dados pode ser justificada porque as imagens médicas são transferidas com o uso do protocolo FTP e neste caso, o protocolo de transporte TCP é usado, provendo transmissão com confirmação de dados.

#### 6.3.1.4 Análise descritiva da Teleconsulta

Tabela 34 - Estatística descritiva do throughput: teleconsulta

<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Mínimo	1,54	Mínimo	2
Média	3,447894737	Média	2,998947368
Mediana	4,01	Mediana	3



<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Máximo	4,87	Máximo	3,63
Desvio Padrão da Média	0,244757403	Desvio Padrão da Média	0,066358475
Desvio Padrão	1,066872787	Desvio Padrão	0,289249887

Através do resumo descritivo da Tabela 34 observa-se que o algoritmo proposto se comportou conforme o desejado, uma vez que foi programado para incrementar o *throughput* dos fluxos, levando em consideração a prioridade e a banda total disponível. Em todos os serviços, inclusive para a Teleconsulta, o algoritmo mostrou-se eficiente provendo incrementos de banda para os fluxos. Por este motivo obteve os melhores valores para o máximo, média e mediana. Naturalmente que com os incrementos de banda, ao longo da provisão do serviço, haverá maior variabilidade no *throughput*, motivo pelo qual o valor do desvio padrão é sempre superior, quando comparado com o desvio padrão do algoritmo nativo.

Tabela 35 - Estatística descritiva da latência: teleconsulta

<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Mínimo	2	Mínimo	2
Média	8,684210526	Média	4205,473684
Mediana	9	Mediana	1027
Máximo	18	Máximo	9091
Desvio Padrão da Média	1,005983943	Desvio Padrão da Média	960,990651
Desvio Padrão	4,384982346	Desvio Padrão	4188,861134

O algoritmo de gerenciamento proposto, *QoEManApp*, consegue gerenciar a latência dos fluxos de forma bem superior ao algoritmo nativo. Através da análise dos dados da Tabela 35 constata-se que uma consulta médica não pode ser provida com os valores apresentados, pois pode haver ecos e até mesmo perda de dados com valores de atrasos tão elevados. Considerando que a política de QoS para a latência deve ficar entre 200 e 400ms, nota-se que em todos os valores (mínimo, mediana, média e máximo) o serviço de Teleconsulta foi provida com excelente qualidade. Inclusive a variabilidade dos dados foi pequena se distanciando de forma demasiada da variabilidade dos dados com o algoritmo padrão do *Floodlight*.

Tabela 36 - Estatística descritiva da perda de pacotes: teleconsulta

<i>Algoritmo Proposto</i>		<i>Algoritmo Nativo</i>	
Mínimo	0,02	Mínimo	0,03
Média	0,275263158	Média	0,438421053
Mediana	0,28	Mediana	0,48
Máximo	0,35	Máximo	0,69
Desvio Padrão da Média	0,014835717	Desvio Padrão da Média	0,05186162
Desvio Padrão	0,06466739	Desvio Padrão	0,226059558

O percentual de descarte de pacotes sumarizado na Tabela 36 demonstra que o algoritmo proposto realiza um controle efetivo no gerenciamento de descarte de pacotes, pois em todas as análises (mínimo, máximo, média, mediana, desvio padrão) o algoritmo proposto provou melhor desempenho que o algoritmo nativo do SDNC *Floodlight*.

### 6.3.2 Inferência Estatística: Intervalo de Confiança

Em muitas situações, uma estimativa pontual, como por exemplo a média, não fornece informação suficiente sobre um parâmetro, então faz-se necessário o uso de outra abordagem, tal como a inferência estatística. A inferência estatística permite conhecer a precisão da estimativa pontual obtida. Uma maneira de se fazer isso é através de uma estimativa intervalar do parâmetro, denominado Intervalo de Confiança (IC). A seguir são apresentados os IC obtidos dos experimentos realizados com os componentes da abordagem proposta e com os componentes nativos do SDNC *Floodlight*.

#### 6.3.2.1 Intervalo de Confiança dos Sinais Ambientais

Os gráficos apresentados nas Figura 50 a 55 mostram as médias amostrais, os limites inferiores, limites superiores e a dispersão das medições do IC dos parâmetros de QoS, provenientes da monitoração dos sinais ambientais. A análise foi realizada em 29 observações.

Figura 50 - IC do throughput: sinais ambientais (abordagem proposta)

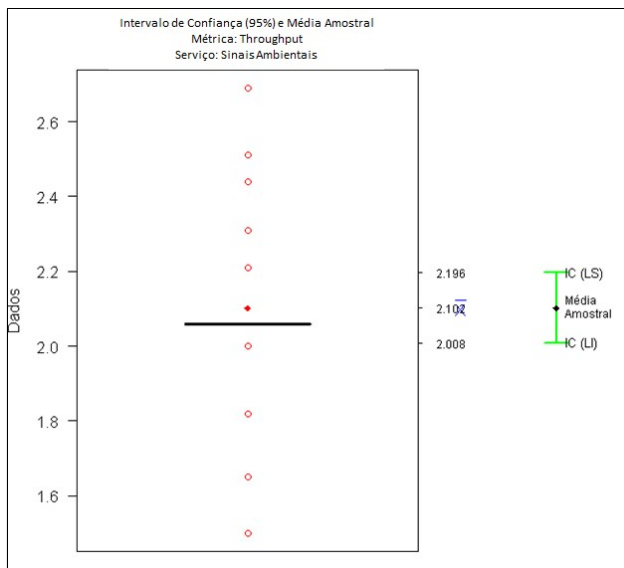
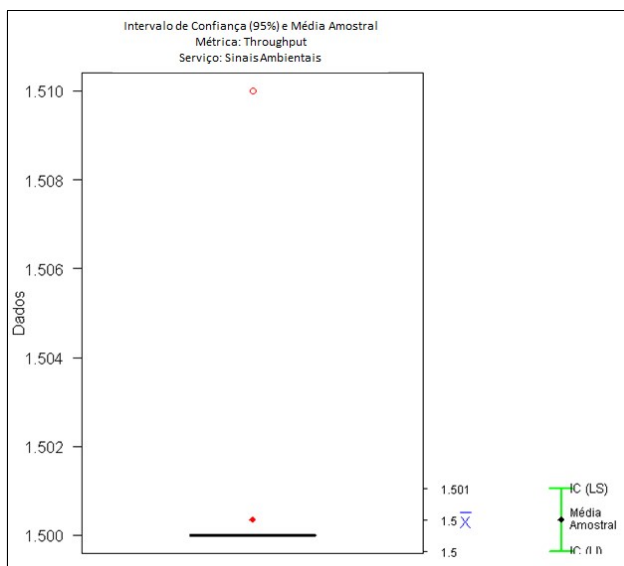


Figura 51 - IC do throughput: sinais ambientais (abordagem nativa)



A análise do gráfico da Figura 50 permite evidenciar que para um

intervalo de confiança de 95%, o *throughput* dos sinais ambientais ficou entre 2,008Mbps e 2,196Mbps; sendo que os dados se concentraram um pouco abaixo da média (2,1Mbps). Enquanto que os resultados obtidos com o uso do algoritmo padrão, exibidos no gráfico da Figura 51, demonstraram que o serviço de monitoramento de sinais ambientais foi provido sempre em 1,5Mbps. Nos gráficos é possível observar também a dispersão das medições de todas as observações e o quanto os valores monitorados se distanciam dos limites inferiores e superiores.

Figura 52 - IC da latência: sinais ambientais (abordagem proposta)

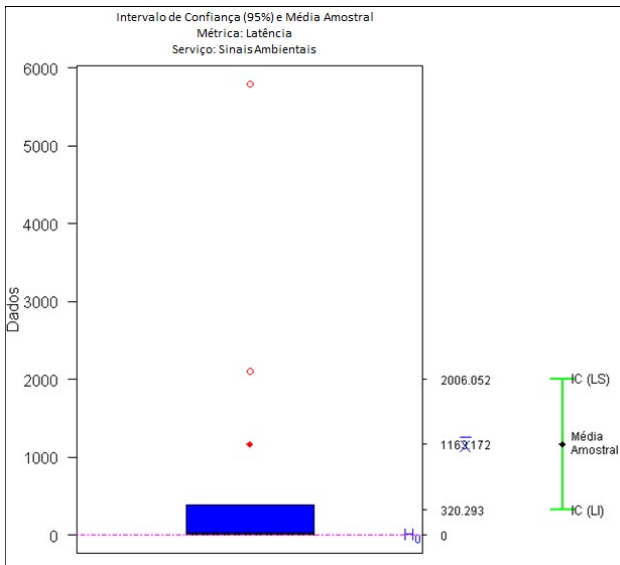
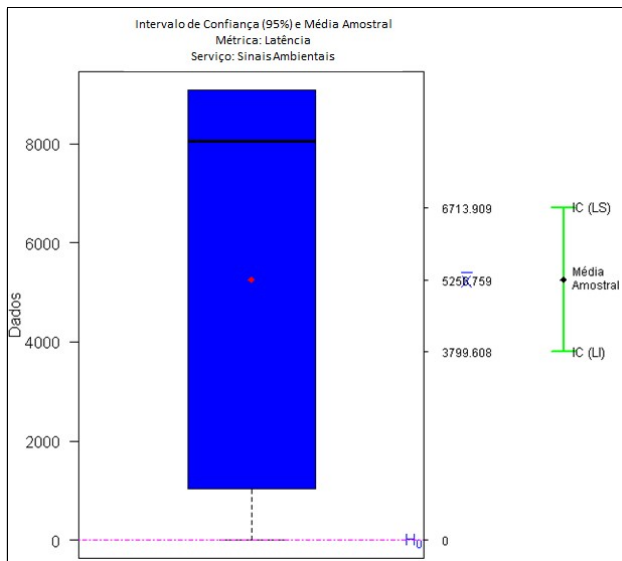


Figura 53 - IC da latência: sinais ambientais (abordagem nativa)



A análise do gráfico da Figura 52 permite evidenciar, dado um intervalo de confiança de 95%, que a latência monitorada ficou entre 320 milissegundos e 2 segundos, porém os dados monitorados se concentraram abaixo do limite inferior. Os maiores valores observados foram medidos no caminho inicial do fluxo, no entanto, conforme análise dos eventos do algoritmo da *QoE-ManApp*, observou-se que para contornar o problema da alta latência a aplicação de gerenciamento planejou a alteração de rota por um caminho com menor latência e o fluxo foi desviado para uma rota alternativa, com menor latência, normalizando o tempo da latência do fluxo. O que não ocorreu com o algoritmo padrão, pois o gráfico da Figura 53 evidencia que com o uso do algoritmo padrão, o serviço de monitoramento de sinais ambientais foi provido com valores de latência extremamente elevados, o que pode inviabilizar o uso desses dados por aplicações de *eHealth* sensível ao contexto.

Figura 54 - IC da perda de pacotes: sinais ambientais (abordagem proposta)

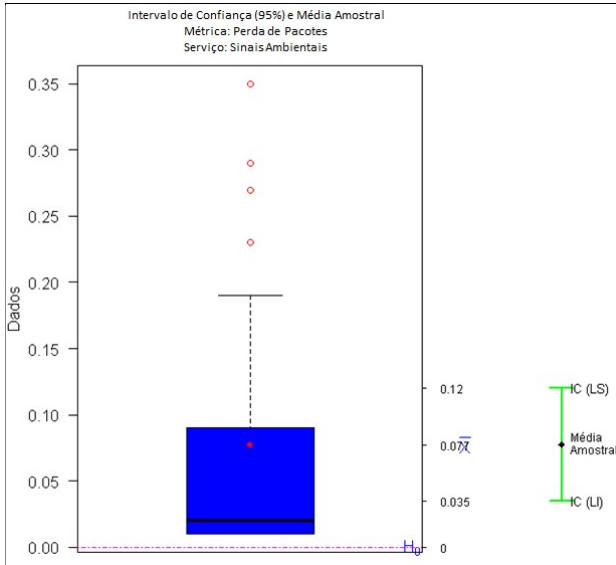
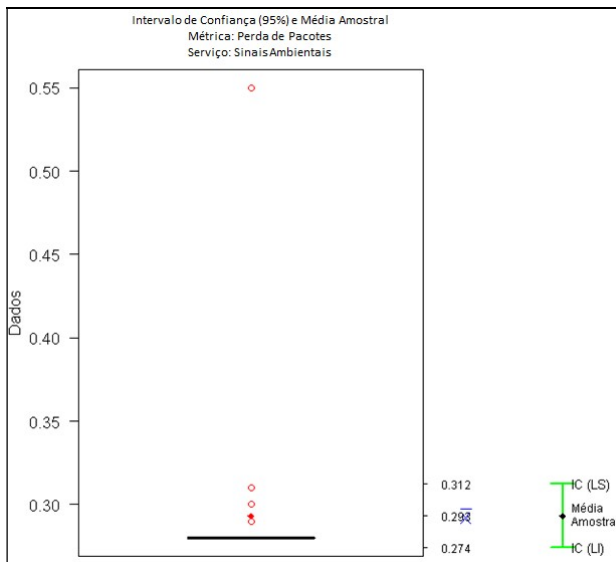


Figura 55 - IC da perda de pacotes: sinais ambientais (abordagem nativa)



O gráfico da Figura 54, considerando um intervalo de confiança

de 95%, demonstra que o serviço de sinais ambientais foi provido com taxa de descarte entre 3,5% e 12%; sendo que os dados se concentraram entre a média e abaixo do limite inferior. Já o gráfico da Figura 55 demonstra que nas mesmas condições experimentadas, a taxa de descarte de pacotes monitorada com o uso do algoritmo nativo ficou entre 27% e 31%. Como o algoritmo nativo não toma decisões, a tendência é que o serviço seja provido com as taxas de descarte apresentadas.

### 6.3.2.2 Intervalo de Confiança dos Sinais Biomédicos

Os gráficos apresentados nas Figura 56 a 61 mostram as médias amostrais, os limites inferiores, limites superiores e a dispersão das medições do IC dos parâmetros de QoS, provenientes da monitoração dos sinais biomédicos. A análise foi realizada em 27 observações.

Figura 56 - IC do throughput: sinais biomédicos (abordagem proposta)

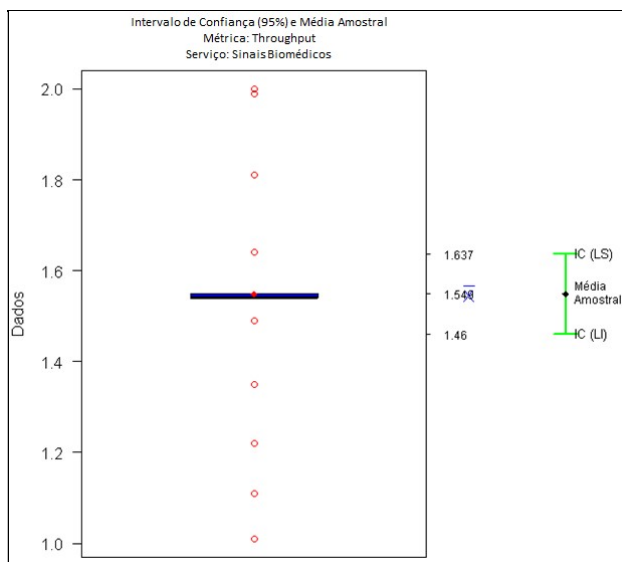
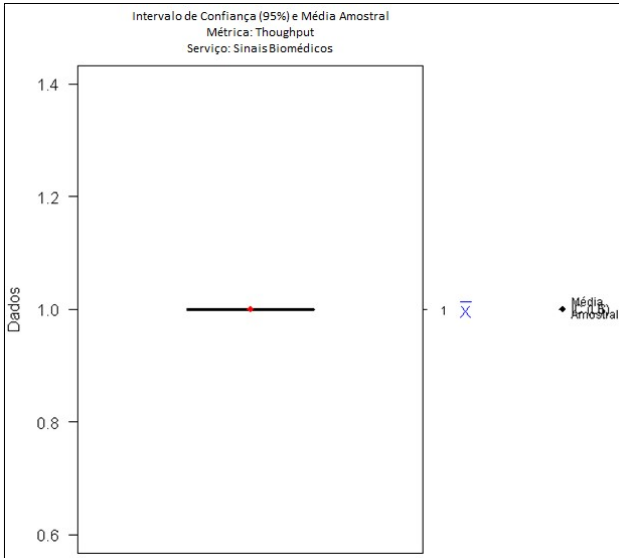


Figura 57 - IC do throughput: sinais biomédicos (abordagem nativa)



Na Figura 56, para um intervalo de confiança de 95%, nota-se que a média amostral das medições da *throughput* ficou em 1,54Mbps, oscilando de 1,46Mbps a 1,63Mbps. Esse aumento e incremento no *throughput* evidencia que o algoritmo proposto tomou decisões de engenharia de tráfego, otimizando o máximo possível a banda para os fluxos. Por outro lado, como o algoritmo nativo não toma decisões, o serviço de sinais biomédicos é provido a uma taxa de bits constante, ou seja, 1Mbps, conforme apresentado na Figura 57.



Figura 58 - IC da latência: sinais biomédicos (abordagem proposta)

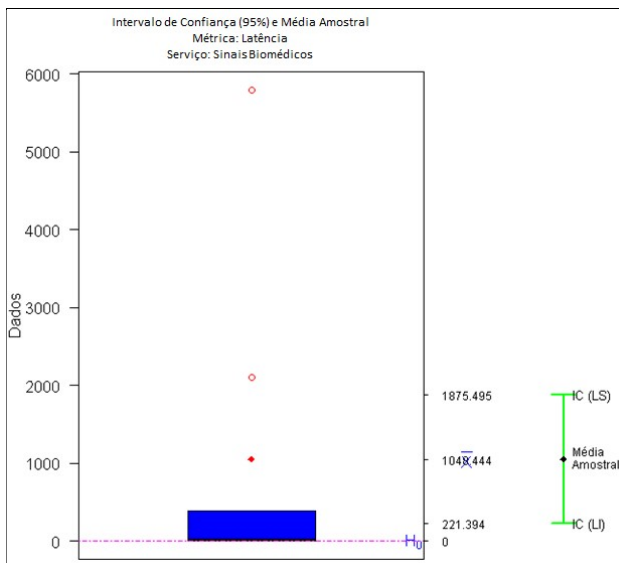
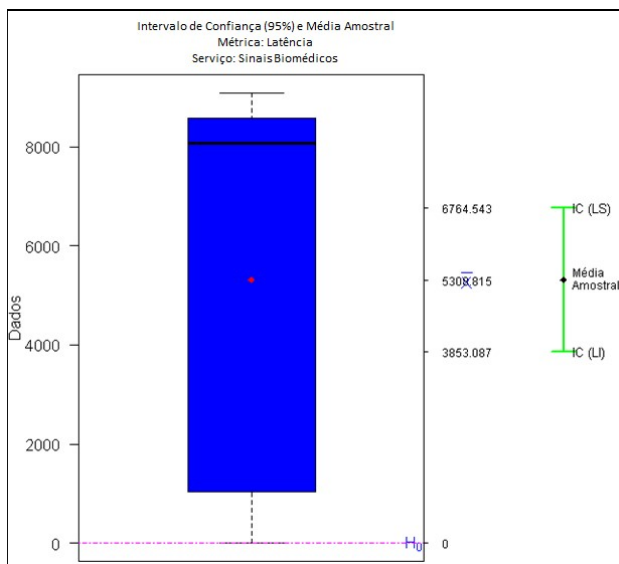


Figura 59 - IC da latência: sinais biomédicos (abordagem nativa)



O gráfico da Figura 58 apresenta os valores do intervalo de

confiança usando os componentes da abordagem proposta. A partir dos resultados experimentais, considerando um intervalo de confiança de 95%, o limite inferior ficou em 221 milissegundos e limite superior em 1,8 segundos. As medições demonstram, através do gráfico, alguns picos para a latência, com valores próximos de 2 e 4 segundos. Contudo, conforme análise dos eventos do algoritmo da *QoE-ManApp*, observou-se que para contornar o problema da alta latência, a aplicação de gerenciamento planejou a alteração de rota por um caminho com menor latência e os fluxos com valores acima do definido na política de QoS foram desviados por um caminho com menor latência, normalizando portanto, os tempos de provisão dos fluxos. Por este motivo, a maior concentração dos dados ficou entre 2 e 221 milissegundos. Em contraste, no gráfico da Figura 59, o limite inferior da latência ficou em 3,8 segundos e o limite superior em 6,7 segundos, isso implica em dizer que para um intervalo de confiança de 95%, a latência na transmissão do sinais biomédicos ficará entre 3,8 e 6,7 segundos, inviabilizando, portanto, o uso do algoritmo nativo do SDNC *Floodlight* na provisão deste tipo de sinal.

Figura 60 - IC da perda de pacotes: sinais biomédicos (abordagem proposta)

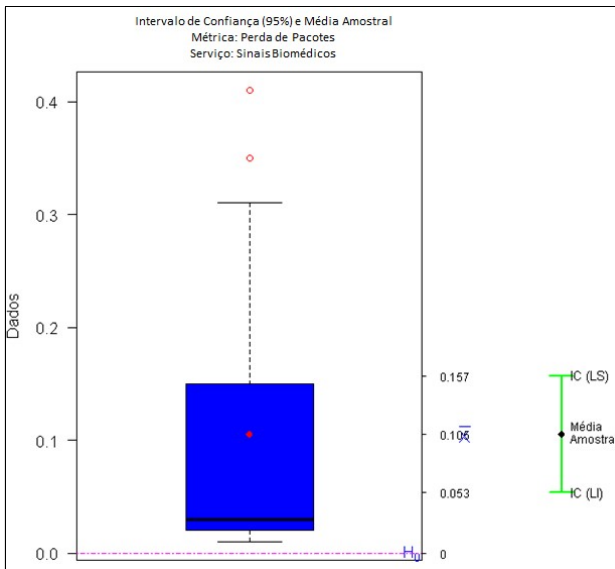
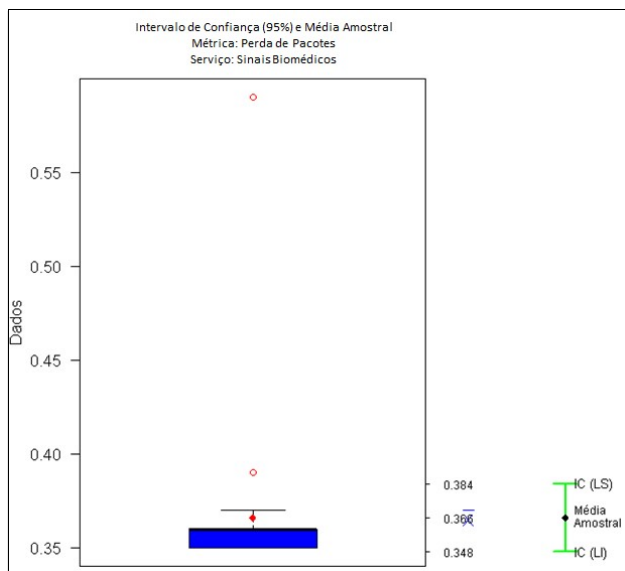


Figura 61 - IC da perda de pacotes: sinais biomédicos (abordagem nativa)



O gráfico da Figura 60 mostra que o limite inferior, a média amostral e o limite superior da taxa de descarte de pacotes ficaram, respectivamente, em 5%, 10% e 15%. Enquanto que com o algoritmo nativo (Figura 61), considerando o mesmo intervalo de confiança, o descarte de pacotes ficou entre 34% e 38%. Novamente observa-se que o algoritmo proposto é mais efetivo que o algoritmo nativo, no entanto, as taxas de descarte ainda são consideradas altas e novos experimentos, com novas configurações devem ser realizados visando reduzir a taxa de descarte de pacotes.

### 6.3.2.3 Intervalo de Confiança na Transferência de Imagens Médicas

Os gráficos apresentados nas Figuras 62 a 67 mostram as médias amostrais, os limites inferiores, limites superiores e a dispersão das medições do IC dos parâmetros de QoS, provenientes da monitoração da transferência de imagens médicas. A análise foi realizada em 23 observações.

Figura 62 - IC do throughput: imagens médicas (abordagem proposta)

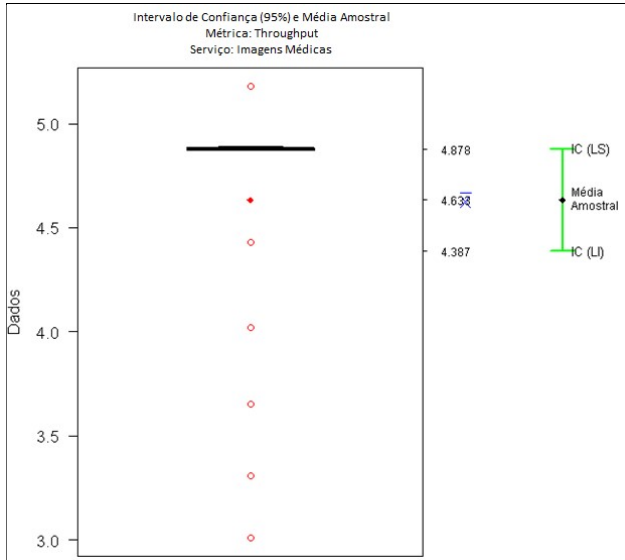
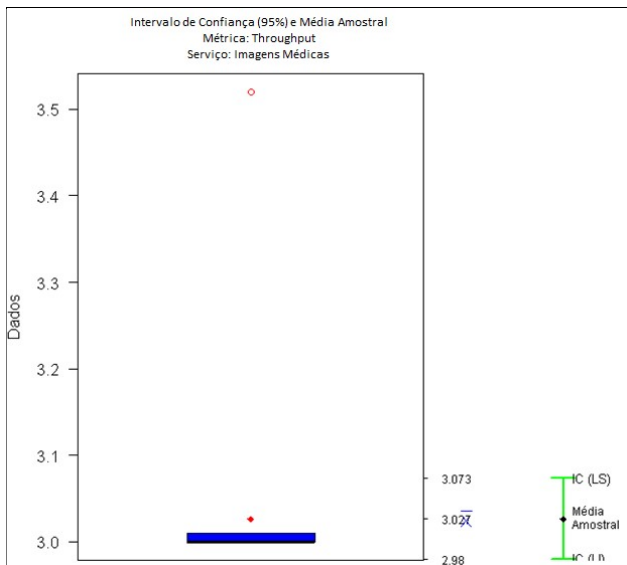


Figura 63 - IC do throughput: imagens médicas (abordagem nativa)



A análise do gráfico da Figura 62 permite evidenciar que para um

intervalo de confiança de 95%, o *throughput* na transmissão de imagens médicas ficou entre 4,3Mbps e 4,8Mbps; sendo que os dados se concentraram próximo do limite superior. Enquanto que os resultados obtidos com o uso do algoritmo padrão, exibidos no gráfico da Figura 63, considerando o mesmo intervalo de confiança, mostram que as imagens médicas foram transmitidas com taxas entre 2,98Mbps e 3,07Mbps; sendo que os dados observados se concentraram abaixo da média.

Figura 64 - IC da latência: imagens médicas (abordagem proposta)

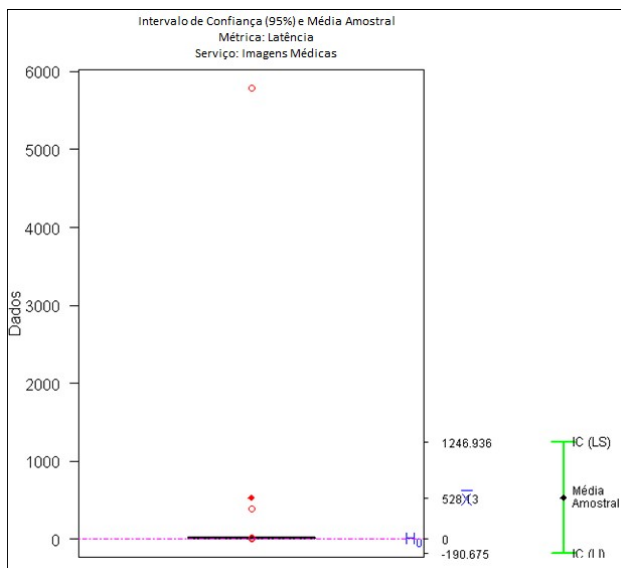
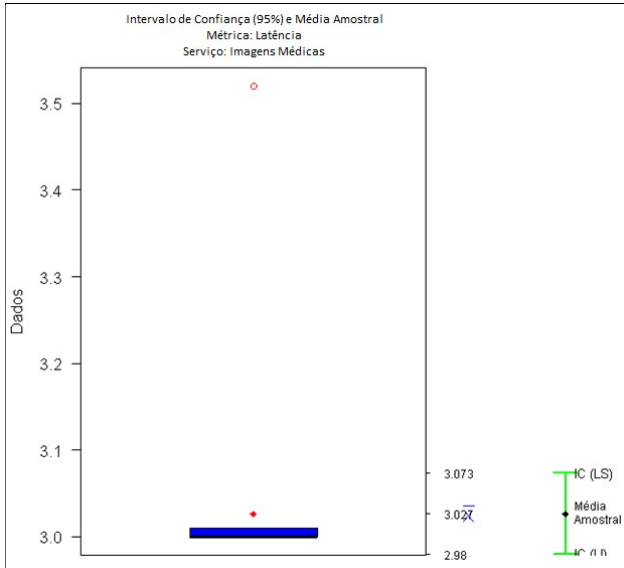


Figura 65 - IC da latência: imagens médicas (abordagem nativa)



O gráfico da Figura 64 apresenta os valores do intervalo de confiança usando os componentes da abordagem proposta. A partir dos resultados experimentais, considerando um intervalo de confiança de 95%, o limite inferior ficou em 1 milissegundo e limite superior em 1,2 segundos. O gráfico apresenta alguns picos para a latência, com valores acima de 5 segundos. Esse valor refere-se ao caminho inicial do fluxo, que após detectado pelo algoritmo, ações de adaptações foram executadas na rede e com isso, pode-se afirmar que o algoritmo mostrou-se efetivo no gerenciamento da latência, pois a grande concentração dos valores medidos ficou próximo do limite inferior. Em contraste, no gráfico da Figura 65, o limite inferior da latência ficou em 2,9 segundos e o limite superior em 3 segundos, isso implica em dizer que para um intervalo de confiança de 95%, com o algoritmo nativo, a latência na transmissão das imagens médicas ficará entre 3,8 e 6,7 segundos.

Figura 66 - IC da perda de pacotes: imagens médicas (abordagem proposta)

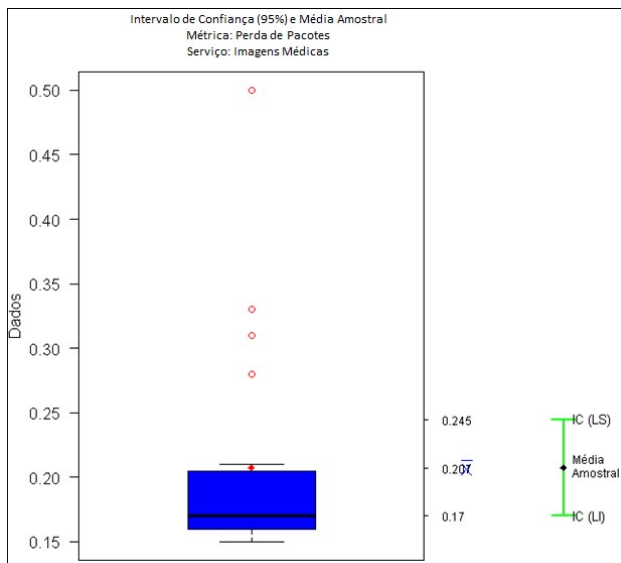
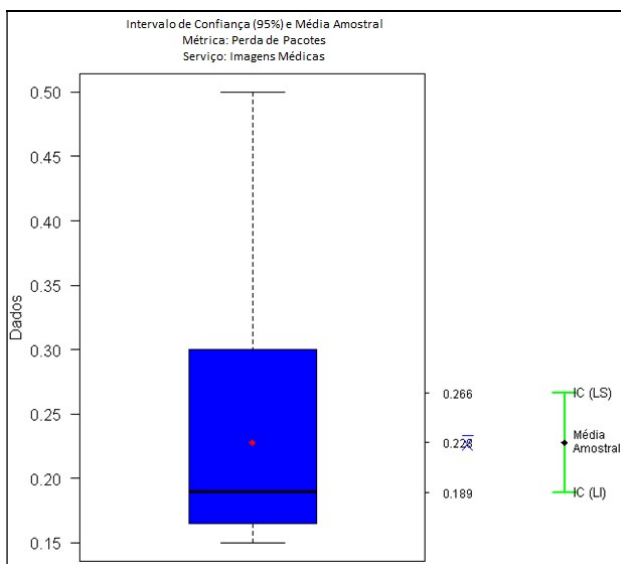


Figura 67 - IC da perda de pacotes: imagens médicas (abordagem nativa)



O gráfico da Figura 66 mostra que o limite inferior, a média

amostral e o limite superior da taxa de descarte de pacotes ficaram, respectivamente, em 17%, 20% e 24%. Enquanto que com o algoritmo nativo (Figura 67), considerando o mesmo intervalo de confiança, o descarte de pacotes ficou entre 18% e 26%. As medidas dos intervalos de confiança para os dois algoritmos ficaram com valores bem próximos, porém, ao analisar a concentração dos dados nos gráficos, fica evidente que o algoritmo nativo, mesmo com taxas de descartes consideradas altas, ainda provê um melhor desempenho. Contudo novos experimentos com o algoritmo proposto devem ser realizados para verificar se a taxa de descarte pode ser reduzida.

#### 6.3.2.4 Intervalo de Confiança da Teleconsulta

Os gráficos apresentados nas Figura 68 a 73 mostram as médias amostrais, os limites inferiores, limites superiores e a dispersão das medições do IC dos parâmetros de QoS provenientes da monitoração da Teleconsulta. A análise foi realizada em 19 observações.

Figura 68 - IC do throughput: teleconsulta (abordagem proposta)

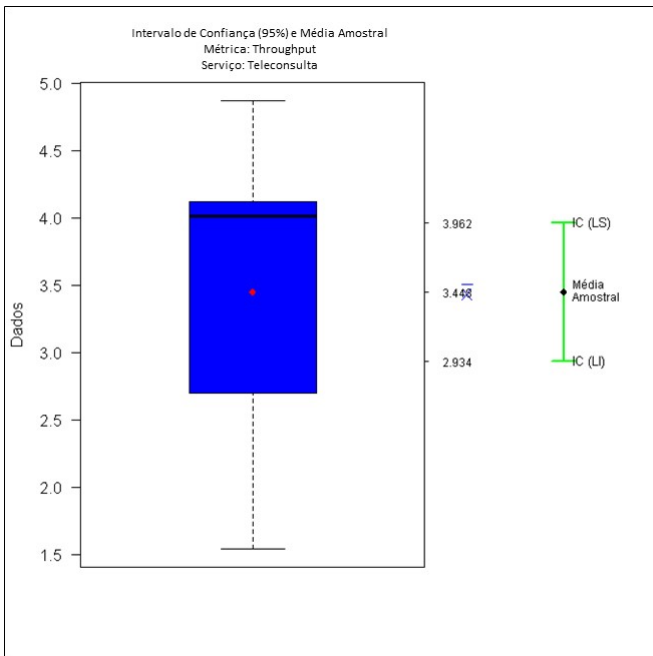
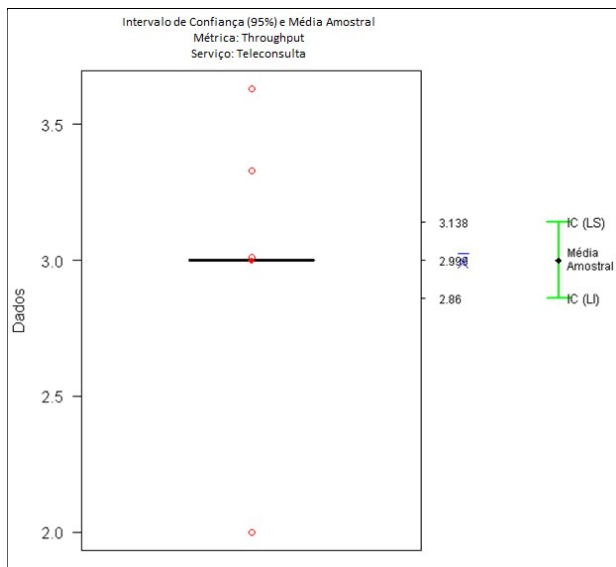




Figura 69 - IC do throughput: teleconsulta (abordagem nativa)



O gráfico da Figura 68 mostra o intervalo de confiança para os dados monitorados com o algoritmo proposto, sendo que o limite inferior, média e limite superior ficaram com os valores de 2,9Mbps, 3,4Mbps e 3,9Mbps, respectivamente. Isso implica em dizer que para um intervalo de confiança de 95% a teleconsulta com o algoritmo proposto será provida com taxas entre 2,9 Mbps e 3,9 Mbps. No entanto, com o algoritmo nativo, gráfico da Figura 69, observa-se que a média amostral das medições ficou em 2,9Mbps, com limite superior de 3,1Mbps e limite inferior de 2,8Mbps. Com estes valores, considerando o intervalo de confiança de 95%, pode-se inferir que a teleconsulta com o algoritmo nativo será transmitida com taxas entre 2,8 Mbps e 3,1 Mbps.

Figura 70 - IC da latência: teleconsulta (abordagem proposta)

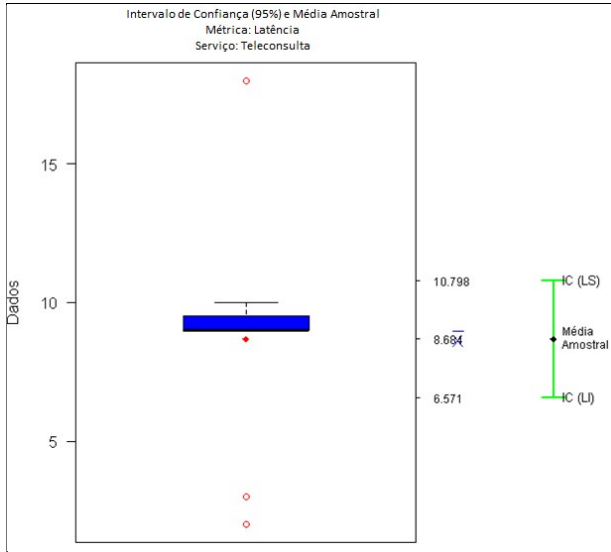
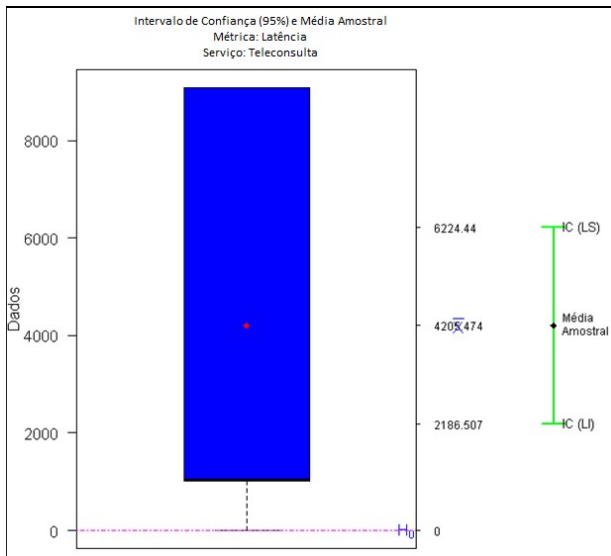


Figura 71 - IC da latência: teleconsulta (abordagem nativa)



No gráfico da Figura 70, para um intervalo de confiança de 95%,

observa-se que a média amostral da latência ficou em 8 milissegundos, com limite superior de 10 milissegundos e limite inferior de 6 milissegundos. É possível observar ainda, que a maioria das medições ficou concentrada acima da média. E para o gráfico da Figura 71 observa-se que, para um intervalo de confiança de 95%, o serviço de teleconsulta foi provisionado com com variações da latência de 2 a 6 segundos. A observação visual dos gráficos permite constatar a eficiência do algoritmo proposto no gerenciamento da latência quando comparado com o algoritmo nativo.

Figura 72 - IC da perda de pacotes: teleconsulta (abordagem proposta)

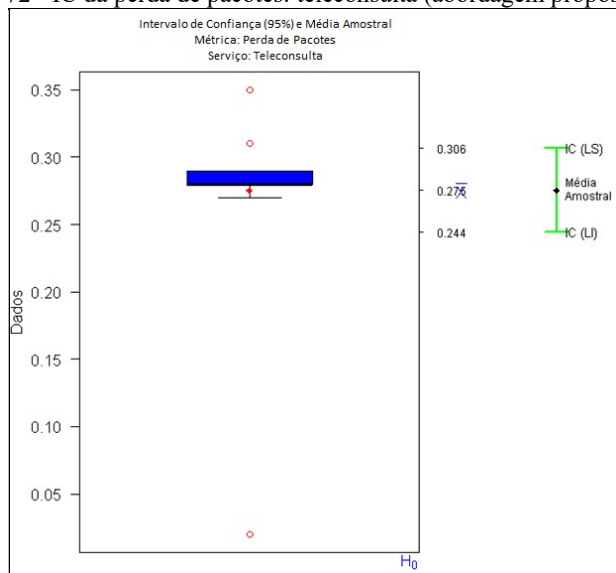
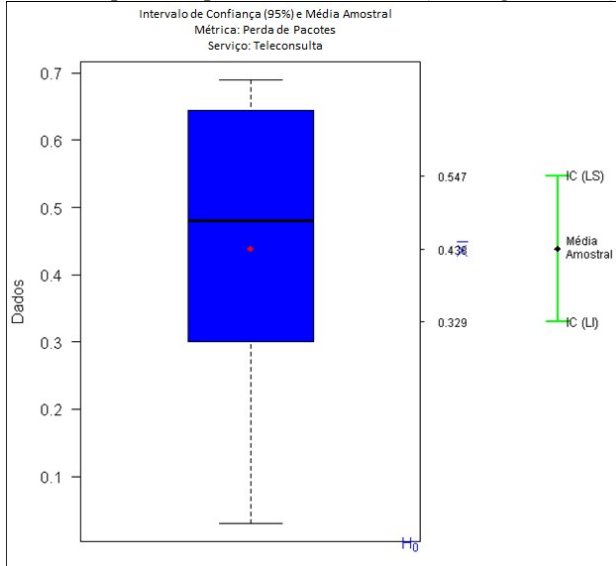


Figura 73 - IC da perda de pacotes: teleconsulta (abordagem nativa)



No gráfico da Figura 72, para um intervalo de confiança de 95%, observa-se que a média amostral das medições ficou em 24%, com limite superior de 30% e limite inferior de 24%. É possível observar ainda, que a dispersão das medições de todas as observações se concentraram entre o valor médio. Enquanto que os resultados experimentais da taxa de descarte de pacotes do gráfico da Figura 73 apresentam média de 43%, limite inferior de 32% e limite superior de 57%. Apesar dos valores elevados no descarte de pacotes para o serviço de Teleconsulta, observa-se através dos gráficos que algoritmo proposto provê uma melhor gestão do descarte comparado com o algoritmo nativo, contudo novos experimentos precisam ser realizados para averiguar se a taxa de descarte de pacotes pode ser reduzida.

### 6.3.3 Inferência Estatística: Teste de Hipóteses

Para verificar se o algoritmo, juntamente com os módulos propostos, provêm alguma melhoria no desempenho da rede quando comparado com o algoritmo e componentes padrão do SDNC *Floodlight*, foi feita a inferência estatística por teste de hipóteses. Hipóteses são afirmações sobre parâmetros populacionais, como por exemplo, o algoritmo proposto provê uma menor latência no

monitoramento de sinais biomédicos, quando comparado com o algoritmo padrão? Após a formulação, as hipóteses foram testadas para verificar a aceitação ou rejeição das mesmas.

### 6.3.3.1 Procedimentos usados no Teste de Hipóteses

A inferência foi realizada com o *Teste t de Student* com amostras pareadas, uma vez que os experimentos foram realizados nas mesmas condições referente ao ambiente, parametrização etc, com a única diferenciação na ativação e desativação do algoritmo.

Este tipo de planejamento é usual quando se deseja determinar o nível de uma certa medida (e.g., latência, perda, *throughput*) antes (componentes nativo) e depois da melhoria de um processo (componentes propostos). Desta forma é calculada a diferença entre as duas amostras e trabalha-se sobre as diferenças das amostras. Os testes foram realizados seguindo as etapas:

1. Formular as hipóteses:
  - ✓ Hipótese nula ( $H_0$ ): aquela considerada verdadeira e que será testada.
  - ✓ Hipótese alternativa ( $H_1$ ): aquela complementar a  $H_0$ , se  $H_0$  é rejeitada, por consequência  $H_1$  é aceita.
2. Definir o nível de significância,  $\alpha$ , utilizado:
  - ✓ No estudo consideramos o valor mais usual na literatura  $\alpha = 0,05$ , este valor foi usado em todos os testes.
3. Verificar qual o teste adequado e calcular a estatística de teste:
  - ✓ Foi usado o *Teste t de Student* com amostras pareadas. A estatística de teste baseia-se nas diferenças DEPOIS – ANTES para cada elemento da amostra.
4. Decidir pela aceitação ou rejeição da hipótese de nulidade:
  - ✓ A decisão é feita com base no *p-value*.
5. Conclusão experimental
  - ✓ Uma explicação com “palavras simples” sobre o resultado do teste de hipóteses.

### 6.3.3.2 Teste de Hipótese do Throughput na Monitoração de Sinais Ambientais

1. Formulação das hipóteses:

- ✓ Hipótese nula ( $H_0$ ): o algoritmo proposto não melhorou o *throughput* no monitoramento de sinais ambientais, ou seja, a diferença média entre o depois e o antes é “0”.
- ✓ Hipótese alternativa ( $H_1$ ): o algoritmo proposto melhorou o *throughput*, ou seja, a diferença média entre o depois e o antes é **maior** do que “0”.

2. Definição do nível de significância utilizado.

- ✓  $\alpha = 0,05$ , para análise unilateral a direita.

3. Cálculo da estatística de teste: a Tabela 37 apresenta o resultado do Teste t para a métrica *Throughput* ao comparar as amostras das medições com e sem o algoritmo ativo.

Tabela 37 - Resultado do Teste t do throughput: sinais ambientais

	<i>throughput</i> (algoritmo proposto)	<i>Throughput</i> (algoritmo nativo)
Média	2,102068966	1,500344828
Variância	0,061331281	3,44828E-06
Observações	29	29
Hipótese da diferença de média	0	
gl	28	
<b>Stat t</b>	<b>13,04977427</b>	
P(T<=t) uni-caudal	1,00452E-13	
<b>t crítico uni-caudal</b>	<b>1,701130934</b>	

4. Decisão da aceitação ou rejeição da hipótese de nulidade:

- ✓ Como a estatística do Teste  $t=13,04$  está dentro da Região Crítica,  $RC: \{t > 1,70\}$ , rejeita-se  $H_0$ .

5. Conclusão experimental

- ✓ Através do resultado do teste tem-se evidências suficientes para afirmar que o algoritmo novo traz um efeito positivo no *Throughput* ao nível de significância de 5%, logo  $t \in RC$ , então  $H_1$  é verdadeira.

### 6.3.3.3 Teste de Hipótese da Latência na Monitoração de Sinais Ambientais

#### 1. Formulação das hipóteses:

- ✓ Hipótese nula ( $H_0$ ): o algoritmo proposto não melhorou a latência no monitoramento de sinais ambientais, ou seja, a diferença média entre o depois e o antes é “0”.
- ✓ Hipótese alternativa ( $H_1$ ): o algoritmo proposto melhorou a latência, ou seja, a diferença média entre o depois e o antes é **menor** do que “0”.

#### 2. Definição do nível de significância utilizado.

- ✓  $\alpha = 0,05$ , para análise unilateral a esquerda.

3. Cálculo da estatística de teste: a Tabela 38 apresenta o resultado do Teste t para a métrica latência ao comparar as amostras das medições com e sem o algoritmo ativo.

Tabela 38 - Resultado do Teste t da latência: sinais ambientais

	<i>Latência (algoritmo proposto)</i>	<i>Latência (algoritmo nativo)</i>
Média	1163,172414	5256,758621
Variância	4910168,148	14674866,76
Observações	29	29
Hipótese da diferença de média	0	
gl	28	
<b>Stat t</b>	<b>-4,304385804</b>	
P(T<t) uni-caudal	9,24455E-05	
<b>t crítico uni-caudal</b>	<b>-1,701130934</b>	

#### 4. Decisão da aceitação ou rejeição da hipótese de nulidade:

- ✓ Como a estatística do Teste  $t = -4,30$  está dentro da Região Crítica,  $RC: \{t < -1,70\}$ , rejeita-se  $H_0$ .

#### 5. Conclusão experimental

- ✓ Através do resultado do teste tem-se evidências suficientes para afirmar que o algoritmo novo melhora a latência ao nível de significância de 5%, logo  $t \in RC$ , então  $H_1$  é verdadeira.

### 6.3.3.4 Comparação da Perda de Pacotes na Monitoração de Sinais Ambientais

#### 1. Formulação das hipóteses:

- ✓ Hipótese nula ( $H_0$ ): o algoritmo proposto não melhorou a taxa de perda de pacotes no monitoramento de sinais ambientais, ou seja, a diferença média entre o depois e o antes é “0”.
- ✓ Hipótese alternativa ( $H_1$ ): o algoritmo proposto melhorou a taxa de perda de pacotes, ou seja, a diferença média entre o depois e o antes é **menor** do que “0”.

#### 2. Definição do nível de significância utilizado.

- ✓  $\alpha = 0,05$ , para análise unilateral a esquerda.

3. Cálculo da estatística de teste: a Tabela 39 apresenta o resultado do Teste t para a métrica de perda de pacotes ao comparar as amostras das medições com e sem o algoritmo ativo.

Tabela 39 - Resultado do Teste t da perda de pacotes: sinais ambientais

	<i>Perda de Pacotes (algoritmo proposto)</i>	<i>Perda de Pacotes (algoritmo nativo)</i>
Média	0,077241379	0,293103448
Variância	0,012606404	0,002515025
Observações	29	29
Hipótese da diferença de média	0	
gl	28	
<b>Stat t</b>	<b>-12,63238679</b>	
P(T<=t) uni-caudal	2,19657E-13	
<b>t crítico uni-caudal</b>	<b>-1,701130934</b>	

#### 4. Decisão da aceitação ou rejeição da hipótese de nulidade:

- ✓ Como a estatística do Teste  $t = -12,63$  está dentro da Região Crítica,  $RC: \{t < -1,70\}$ , rejeita-se  $H_0$ .

#### 5. Conclusão experimental

- ✓ Através do resultado do teste tem-se evidências suficientes para afirmar que o algoritmo novo melhora a taxa de perda de pacotes ao nível de significância de 5%, logo  $t \in RC$ , então  $H_1$  é verdadeira.



### 6.3.3.5 Comparação do *Throughput* na Monitoração de Sinais Biomédicos

#### 1. Formulação das hipóteses:

- ✓ Hipótese nula ( $H_0$ ): o algoritmo proposto não melhorou o *throughput* no monitoramento de sinais biomédicos, ou seja, a diferença média entre o depois e o antes é “0”.
- ✓ Hipótese alternativa ( $H_1$ ): o algoritmo proposto melhorou o *throughput* no monitoramento de sinais biomédicos, ou seja, a diferença média entre o depois e o antes é **maior** do que “0”.

#### 2. Definição do nível de significância utilizado.

- ✓  $\alpha = 0,05$ , para análise unilateral a direita.

3. Cálculo da estatística de teste: a Tabela 40 apresenta o resultado do Teste t para a métrica *Throughput* ao comparar as amostras das medições com e sem o algoritmo ativo.

Tabela 40 - Resultado do Teste t do throughput: sinais biomédicos

	<i>throughput</i> (algoritmo proposto)	<i>throughput</i> (algoritmo novo)
Média	1,548888889	1
Variância	0,050171795	0
Observações	27	27
Hipótese da diferença de média	0	
gl	26	
<b>Stat t</b>	<b>12,73316898</b>	
P(T<=t) uni-caudal	5,56776E-13	
<b>t crítico uni-caudal</b>	<b>1,70561792</b>	

#### 4. Decisão da aceitação ou rejeição da hipótese de nulidade:

- ✓ Como a estatística do Teste  $t=12,73$  está dentro da Região Crítica,  $RC: \{t > 1,70\}$ , rejeita-se  $H_0$ .

#### 5. Conclusão experimental

- ✓ Através do resultado do teste tem-se evidências suficientes para afirmar que o algoritmo novo traz um efeito positivo no *Throughput* ao nível de significância de 5%, logo  $t \in RC$ , então  $H_1$  é

verdadeira.

### 6.3.3.6 Comparação da Latência na Monitoração de Sinais Biomédicos

#### 1. Formulação das hipóteses:

- ✓ Hipótese nula ( $H_0$ ): o algoritmo proposto não melhorou a latência no monitoramento de sinais biomédicos, ou seja, a diferença média entre o depois e o antes é “0”.
- ✓ Hipótese alternativa ( $H_1$ ): o algoritmo proposto melhorou a latência no monitoramento de sinais biomédicos, ou seja, a diferença média entre o depois e o antes é **menor** do que “0”.

#### 2. Definição do nível de significância utilizado.

- ✓  $\alpha = 0,05$ , para análise unilateral a esquerda.

3. Cálculo da estatística de teste: a Tabela 41 apresenta o resultado do Teste t para a métrica latência ao comparar as amostras das medições com e sem o algoritmo ativo.

Tabela 41 - Resultado do Teste t da latência: sinais biomédicos

	<i>Latência (algoritmo proposto)</i>	<i>Latência (algoritmo nativo)</i>
Média	1048,444444	5308,814815
Variância	4370999,103	13541815
Observações	27	27
Hipótese da diferença de média	0	
gl	26	
<b>Stat t</b>	<b>-4,521613948</b>	
P(T<=t) uni-caudal	5,94142E-05	
<b>t crítico uni-caudal</b>	<b>1,70561792</b>	

#### 4. Decisão da aceitação ou rejeição da hipótese de nulidade:

- ✓ Como a estatística do Teste  $t = -4,52$  está dentro da Região Crítica,  $RC: \{t < -1,70\}$ , rejeita-se  $H_0$ .

#### 5. Conclusão experimental

- ✓ Através do resultado do teste tem-se evidências suficientes para afirmar que o algoritmo novo melhorou a latência ao nível de significância de 5%, logo  $t \in RC$ , então  $H_1$  é verdadeira.

### 6.3.3.7 Comparação da Perda de Pacotes na Monitoração de Sinais Biomédicos

#### 1. Formulação das hipóteses:

- ✓ Hipótese nula ( $H_0$ ): o algoritmo proposto não melhorou a taxa de perda de pacotes no monitoramento de sinais biomédicos, ou seja, a diferença média entre o depois e o antes é “0”.
- ✓ Hipótese alternativa ( $H_1$ ): o algoritmo proposto melhorou a taxa de perda de pacotes no monitoramento de sinais biomédicos, ou seja, a diferença média entre o depois e o antes é **menor** do que “0”.

#### 2. Definição do nível de significância utilizado.

- ✓  $\alpha = 0,05$ , para análise unilateral a esquerda.

3. Cálculo da estatística de teste: a Tabela 42 apresenta o resultado do Teste t para a métrica de perda de pacotes ao comparar as amostras das medições com e sem o algoritmo ativo.

Tabela 42 - Resultado do Teste t da perda de pacotes: sinais biomédicos

	<i>Perda de Pacotes (algoritmo proposto)</i>	<i>Perda de Pacotes (algoritmo nativo)</i>
Média	0,105185185	0,365925926
Variância	0,017187464	0,00208661
Observações	27	27
Hipótese da diferença de média	0	
gl	26	
<b>Stat t</b>	<b>-12,29367068</b>	
P(T<=t) uni-caudal	1,22406E-12	
<b>t crítico uni-caudal</b>	<b>-1,70561792</b>	

#### 4. Decisão da aceitação ou rejeição da hipótese de nulidade:

- ✓ Como a estatística do Teste  $t = -12,29$  está dentro da Região Crítica,  $RC: \{t < -1,70\}$ , rejeita-se  $H_0$ .

#### 5. Conclusão experimental

- ✓ Através do resultado do teste tem-se evidências suficientes para afirmar que o algoritmo novo melhorou a taxa de descarte de pacotes ao nível de

significância de 5%, logo  $t \in RC$ , então  $H_1$  é verdadeira.

### 6.3.3.8 Comparação do *Throughput* na Transferência de Imagens Médicas

#### 1. Formulação das hipóteses:

- ✓ Hipótese nula ( $H_0$ ): o algoritmo proposto não melhorou o *throughput* na transferência de imagens médicas, ou seja, a diferença média entre o depois e o antes é “0”.
- ✓ Hipótese alternativa ( $H_1$ ): o algoritmo proposto melhorou o *throughput* na transferência de imagens médicas, ou seja, a diferença média entre o depois e o antes é **maior** do que “0”.

#### 2. Definição do nível de significância utilizado.

- ✓  $\alpha = 0,05$ , para análise unilateral a direita.

3. Cálculo da estatística de teste: a Tabela 43 apresenta o resultado do Teste t para a métrica *Throughput* ao comparar as amostras das medições com e sem o algoritmo ativo.

Tabela 43 - Resultado do Teste t do throughput: imagens médicas

	<i>throughput</i> (algoritmo proposto)	<i>throughput</i> (algoritmo nativo)
Média	4,632608696	3,026521739
Variância	0,322711067	0,011596443
Observações	23	23
Hipótese da diferença de média	0	
gl	22	
<b>Stat t</b>	<b>13,84861644</b>	
P(T<=t) uni-caudal	1,21119E-12	
<b>t crítico uni-caudal</b>	<b>1,717144374</b>	

#### 4. Decisão da aceitação ou rejeição da hipótese de nulidade:

- ✓ Como a estatística do Teste  $t=13,84$  está dentro da Região Crítica,  $RC: \{t > 1,71\}$ , rejeita-se  $H_0$ .

#### 5. Conclusão experimental

- ✓ Através do resultado do teste tem-se evidências suficientes para afirmar que o algoritmo novo traz

um efeito positivo no *Throughput* ao nível de significância de 5%, logo  $t \in RC$ , então  $H_1$  é verdadeira.

### 6.3.3.9 Comparação da Latência na Transferência de Imagens Médicas

#### 1. Formulação das hipóteses:

- ✓ Hipótese nula ( $H_0$ ): o algoritmo proposto não melhorou a latência na transferência de imagens médicas, ou seja, a diferença média entre o depois e o antes é “0”.
- ✓ Hipótese alternativa ( $H_1$ ): o algoritmo proposto melhorou a latência na transferência de imagens médicas, ou seja, a diferença média entre o depois e o antes é **menor** do que “0”.

#### 2. Definição do nível de significância utilizado.

- ✓  $\alpha = 0,05$ , para análise unilateral a esquerda.

3. Cálculo da estatística de teste: a Tabela 44 apresenta o resultado do Teste t para a métrica latência ao comparar as amostras das medições com e sem o algoritmo ativo.

Tabela 44 - Resultado do Teste t da latência: imagens médicas

	<i>Latência (algoritmo proposto)</i>	<i>Latência (algoritmo nativo)</i>
Média	528,1304348	5489521,739
Variância	2763037,028	1,57776E+13
Observações	23	23
Hipótese da diferença de média	0	
gl	22	
Stat t	<b>-6,626048887</b>	
P(T<=t) uni-caudal	5,79267E-07	
t crítico uni-caudal	<b>-1,717144374</b>	

#### 4. Decisão da aceitação ou rejeição da hipótese de nulidade:

- ✓ Como a estatística do Teste  $t = -6,62$  está dentro da Região Crítica,  $RC: \{t < -1,71\}$ , rejeita-se  $H_0$ .

#### 5. Conclusão experimental

- ✓ Através do resultado do teste tem-se evidências suficientes para afirmar que o algoritmo novo

melhorou a latência ao nível de significância de 5%, logo  $t \in RC$ , então  $H_1$  é verdadeira.

### 6.3.3.10 Comparação da Perda de Pacotes na Transferência de Imagens Médicas

#### 1. Formulação das hipóteses:

- ✓ Hipótese nula ( $H_0$ ): o algoritmo proposto não melhorou a taxa de perda de pacotes na transferência de imagens médicas, ou seja, a diferença média entre o depois e o antes é “0”.
- ✓ Hipótese alternativa ( $H_1$ ): o algoritmo proposto melhorou a taxa de perda de pacotes na transferência de imagens médicas, ou seja, a diferença média entre o depois e o antes é **menor** do que “0”.

#### 2. Definição do nível de significância utilizado.

- ✓  $\alpha = 0,05$ , para análise unilateral a esquerda.

3. Cálculo da estatística de teste: o resultado do Teste t ao comparar as amostras das medições da perda de pacotes, com e sem o algoritmo ativo, é ilustrado na Tabela 45.

Tabela 45 - Resultado do Teste t da perda de pacotes: imagens médicas

	<i>Perda de Pacotes (algoritmo proposto)</i>	<i>Perda de Pacotes (algoritmo nativo)</i>
Média	0,207391304	0,227826087
Variância	0,007501976	0,007917787
Observações	23	23
Hipótese da diferença de média	0	
gl	22	
<b>Stat t</b>	<b>-2,606726371</b>	
P(T<=t) uni-caudal	0,008051845	
<b>t crítico uni-caudal</b>	<b>-1,717144374</b>	

#### 4. Decisão da aceitação ou rejeição da hipótese de nulidade:

- ✓ Como a estatística do Teste  $t = -2,60$  está dentro da Região Crítica,  $RC: \{t < -1,71\}$ , rejeita-se  $H_0$ .

#### 5. Conclusão experimental

- ✓ Através do resultado do teste tem-se evidências

suficientes para afirmar que o algoritmo novo melhora a taxa de descarte de pacotes ao nível de significância de 5%, logo  $t \in RC$ , então  $H_1$  é verdadeira.

### 6.3.3.11 Comparação do *Throughput* para a Teleconsulta

#### 1. Formulação das hipóteses:

- ✓ Hipótese nula ( $H_0$ ): o algoritmo proposto não melhorou o *throughput* para a teleconsulta, ou seja, a diferença média entre o depois e o antes é “0”.
- ✓ Hipótese alternativa ( $H_1$ ): o algoritmo proposto melhorou o *throughput* para a teleconsulta, ou seja, a diferença média entre o depois e o antes é **maior** do que “0”.

#### 2. Definição do nível de significância utilizado.

- ✓  $\alpha = 0,05$ , para análise unilateral a direita.

3. Cálculo da estatística de teste: o resultado do Teste t ao comparar as amostras das medições do *Throughput*, com e sem o algoritmo ativo, é ilustrado na Tabela 46.

Tabela 46 - Resultado do Teste t do throughput: teleconsulta

	<i>Throughput</i> (algoritmo proposto)	<i>throughput</i> (algoritmo nativo)
Média	3,447894737	2,998947368
Variância	1,138217544	0,083665497
Observações	19	19
	0,194639412	
Hipótese da diferença de média	0	
gl	18	
<b>Stat t</b>	<b>1,864360258</b>	
P(T<=t) uni-caudal	0,03933181	
<b>t crítico uni-caudal</b>	<b>1,734063607</b>	

#### 4. Decisão da aceitação ou rejeição da hipótese de nulidade:

- ✓ Como a estatística do Teste  $t=1,86$  está dentro da Região Crítica,  $RC: \{t > 1,73\}$ , rejeita-se  $H_0$ .

#### 5. Conclusão experimental

- ✓ Através do resultado do teste tem-se evidências suficientes para afirmar que o algoritmo novo traz um efeito positivo no *Throughput* ao nível de significância de 5%, logo  $t \in RC$ , então  $H_1$  é verdadeira.

### 6.3.3.12 Comparação da Latência para a Teleconsulta

#### 1. Formulação das hipóteses:

- ✓ Hipótese nula ( $H_0$ ): o algoritmo proposto não melhorou a latência para a teleconsulta, ou seja, a diferença média entre o depois e o antes é “0”.
- ✓ Hipótese alternativa ( $H_1$ ): o algoritmo proposto melhorou a latência para a teleconsulta, ou seja, a diferença média entre o depois e o antes é **menor** do que “0”.

#### 2. Definição do nível de significância utilizado.

- ✓  $\alpha = 0,05$ , para análise unilateral a esquerda.

3. Cálculo da estatística de teste: o resultado do Teste t ao comparar as amostras das medições da *Latência*, com e sem o algoritmo ativo, é ilustrado na Tabela 47.

Tabela 47 - Resultado do Teste t da latência: teleconsulta

	<i>Latência (algoritmo on)</i>	<i>Latência (algoritmo off)</i>
Média	8,684210526	4205,473684
Variância	19,22807018	17546557,6
Observações	19	19
Correlação de Pearson	0,019317453	
Hipótese da diferença de média	0	
gl	18	
<b>Stat t</b>	<b>-4,367235039</b>	
P(T<=t) uni-caudal	0,00018572	
<b>t crítico uni-caudal</b>	<b>-1,734063607</b>	

#### 4. Decisão da aceitação ou rejeição da hipótese de nulidade:

- ✓ Como a estatística do Teste  $t = -4,36$  está dentro da Região Crítica,  $RC: \{t < -1,73\}$ , rejeita-se  $H_0$ .



### 5. Conclusão experimental

- ✓ Através do resultado do teste tem-se evidências suficientes para afirmar que o algoritmo novo melhora a latência ao nível de significância de 5%, logo  $t \in RC$ , então  $H_1$  é verdadeira.

### 6.3.3.13 Comparação da Perda de Pacotes para a Teleconsulta

#### 1. Formulação das hipóteses:

- ✓ Hipótese nula ( $H_0$ ): o algoritmo proposto não melhorou a taxa de perda de pacotes para a teleconsulta, ou seja, a diferença média entre o depois e o antes é “0”.
- ✓ Hipótese alternativa ( $H_1$ ): o algoritmo proposto melhorou a taxa de perda de pacotes para a teleconsulta, ou seja, a diferença média entre o depois e o antes é **menor** do que “0”.

#### 2. Definição do nível de significância utilizado.

- ✓  $\alpha = 0,05$ , para análise unilateral a esquerda.

3. Cálculo da estatística de teste: a Tabela 48 apresenta o resultado do Teste t para a métrica de perda de pacotes ao comparar as amostras das medições com e sem o algoritmo ativo.

Tabela 48 - Resultado do Teste t da perda de pacotes: teleconsulta

	<i>Perda de Pacotes (algoritmo on)</i>	<i>Perda de Pacotes (algoritmo off)</i>
Média	0,275263158	0,438421053
Variância	0,004181871	0,051102924
Observações	19	19
Correlação de Pearson	0,110429074	
Hipótese da diferença de média	0	
gl	18	
<b>Stat t</b>	-3,117086908	
P(T<t) uni-caudal	0,002976261	
<b>t crítico uni-caudal</b>	-1,734063607	

#### 4. Decisão da aceitação ou rejeição da hipótese de nulidade:

- ✓ Como a estatística do Teste  $t = -3,11$  está dentro da

Região Crítica,  $RC: \{t < -1,73\}$ , rejeita-se  $H_0$ .

#### 5. Conclusão experimental

- ✓ Através do resultado do teste tem-se evidências suficientes para afirmar que o algoritmo novo melhorou a taxa de descarte de pacotes ao nível de significância de 5%, logo  $t \in RC$ , então  $H_1$  é verdadeira.

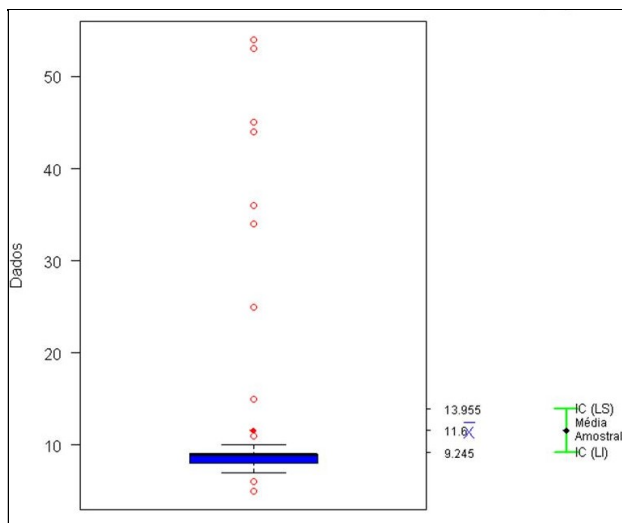
### 6.3.4 Tempo Médio para Restaurar a QoE

No experimento com o algoritmo de gerenciamento da QoE ativo foi possível constatar que o mecanismo de inteligência (aplicação de gerenciamento da QoE – *QoE-ManApp*) é capaz de detectar, planejar e desencadear ações para restaurar a QoE, usando o conhecimento disponível na KB. Assim, como também foi possível evidenciar que os mecanismos de controle, *OFQuality* e *QoC-Control*, atuaram juntamente com a *QoE-ManApp* para que a política de QoS planejada fosse executada.

Com o uso de uma tabela que guarda todos os eventos realizados pela aplicação de gerenciamento observou-se que as indagações iniciais sobre as funcionalidades do algoritmo puderam ser respondidas. Um fato curioso na análise dos registros do experimento foi em relação ao tempo da detecção e restauração da QoE, foi observado valores altos, valores muitos baixos e valores muito próximos. Este fato motivou a exploração da tabela de eventos para analisar com mais cuidado, o tempo de reação do mecanismo de inteligência, pois o fator tempo é crucial se tratando de serviços de *eHealth*. Assim, formulou-se a seguinte pergunta para esta análise dos dados: - qual o tempo médio em segundos para a restauração da QoE?

O gráfico da Figura 74 apresenta alguns dados estatísticos do tempo médio da restauração da QoE. Conforme pode ser observado a média amostral ficou em 11,6 segundos, com limite superior de 13,955 segundos e 9,245 segundos de limite inferior, isso implica em dizer que para um intervalo de confiança de 95%, o tempo de reação ficará entre 9,245 e 13,955 segundos. No entanto, conforme mostra o gráfico, a maior concentração dos dados ficou entre o limite inferior.

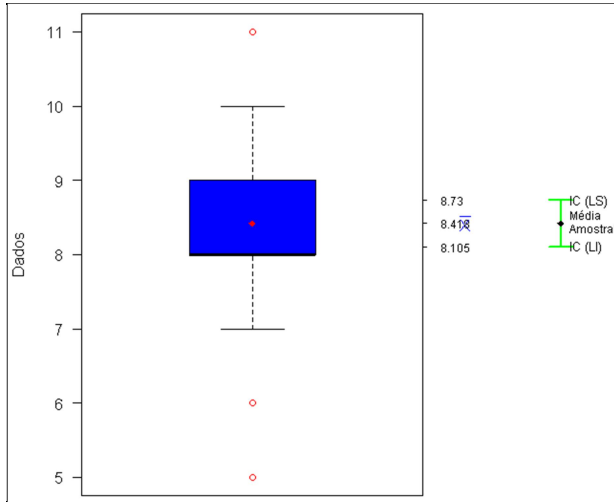
Figura 74 - IC na restauração da QoE, sem limpeza nos dados



No gráfico da Figura 74 pode ser observado os *outliers*, pontos muito distantes das demais observações. Se for realizada uma limpeza nos dados removendo os pontos fora da curva, então, pode-se afirmar com 95% de confiança que o tempo médio de restauração da QoE é 8,41791, com limite inferior de 8,105457 e 8,730364, conforme mostra a Figura 75.

Consideramos que o tempo médio de reação do algoritmo proposto é aceitável, uma vez que a QoE é restaurada sem impactar na qualidade percebida pelo usuário. Portanto, a solução proposta é capaz de detectar e restaurar a QoE, sem interferir na UX.

Figura 75 - IC na restauração da QoE, com limpeza nos dados



A estatística descritiva dos dois casos (Figura 74 e 75) está resumida na Tabela 49. Na tabela é possível observar o tempo máximo e mínimo necessário para o algoritmo detectar e restaurar a QoE, usando um intervalo de confiabilidade de 95%.

Tabela 49 - Informações estatísticas do tempo de restauração da QoE

<i>Informações Estatísticas da Figura 74</i>		<i>Informações Estatísticas da Figura 75</i>	
Estatística T	9,815165	Estatística T	53,79011
Graus de Liberdade	74	Graus de Liberdade	66
P-valor	4,88E-15	P-valor	3,09E-56
Média Amostral	11,6	Média Amostral	8,41791
Desvio Padrão Amostral	10,23507	Desvio Padrão Amostral	1,280971
Tamanho da Amostra	75	Tamanho da Amostra	67
Hipótese Alternativa Diferente de	0	Hipótese Alternativa Diferente de	0
Nível de Confiança	95%	Nível de Confiança	95%
Limite Inferior	9,245123	Limite Inferior	8,105457
Limite Superior	13,95488	Limite Superior	8,730364

## 6.4 CONSIDERAÇÕES DO CAPÍTULO

Para contemplar o quinto objetivo do trabalho, neste capítulo foi apresentado o cenário e o ambiente experimental, com o propósito de avaliar a abordagem de gerenciamento de QoE. Embora o modelo possa ser utilizado em outros contextos de prestação de serviços, o cenário foi idealizado por acreditar que num futuro próximo as pessoas serão atendidas de suas residências por centrais de atendimento a saúde, desta forma atuando como provedoras e consumidoras de seus próprios contextos.

As aplicações da área da saúde tem emergido em muitos países, sobretudo em função da longevidade e qualidade de vida das pessoas, redução de custos de apoio à saúde e assistência social. Nesse cenário, as iniciativas públicas e privadas têm investido nos *Homecare Services*, para que as pessoas possam ser assistidas remotamente em suas próprias casas ou em centros de acolhimento. Com base nisto vislumbramos uma arquitetura genérica de gerenciamento da QoE que pode ser usada na previsão dos serviços de *eHealth*. A solução de gerenciamento da QoE proposta é capaz de:

- a) Detectar as conexões dos serviços e alocar caminhos de acordo com as definições de política de QoS disponíveis na KB.
- b) Monitorar os fluxos ativos, coletando e persistindo os dados na KB.
- c) Classificar os alertas com o uso das regras semânticas.
- d) Planejar uma ação de melhoria na rede e acionar os mecanismos de controle para restaurar ou otimizar a QoE, com base em adaptações de políticas.

A solução proposta foi experimentada num AAL, usando uma infraestrutura SDN. Os experimentos foram realizados com os componentes nativos do controlador SDNC e com os componentes da abordagem proposta. Para avaliar os ganhos obtidos, os dados dos resultados experimentais foram avaliados através de estatística descritiva, intervalo de confiança e *Teste t*. Com esta avaliação foi possível observar melhorias significativas no desempenho da solução proposta quando comparada com a abordagem de provisão do serviço de forma tradicional. A inferência estatística com aplicação do *Test t* permitiu evidenciar, com intervalo de confiança de 95%, que todos os serviços foram providos com qualidade superior ao do algoritmo nativo do SDNC *Floodlight*.

Os resultados experimentais demonstraram que a proposta é

viável e funcional, pois os mecanismos de gerenciamento atuaram de forma eficiente e autônoma, uma vez que o tempo entre a detecção de degradação na QoE e a adaptação de políticas nos recursos de rede não influenciou na qualidade percebida pelo usuário.

## 7 CONSIDERAÇÕES FINAIS

Neste capítulo são descritas as conclusões e as contribuições da pesquisa. O capítulo também relaciona os objetivos propostos *versus* alcançados, como forma de evidenciar o cumprimento dos objetivos e metas do trabalho. Por último são apresentadas as pretensões de trabalhos futuros e algumas publicações já realizadas visando a melhoria e extensão da pesquisa.

### 7.1 CONCLUSÕES

Este trabalho foi motivado pela forma com que os provedores de serviços provisionam os serviços aos seus usuários. Na sua grande maioria, os serviços digitais são providos pensando, prioritariamente, no servidor de banco de dados, na rede de transporte, cabeamento utilizado, banda contratada, retorno financeiro, entre outros aspectos relacionados com a tecnologia.

Quando um serviço com restrições de qualidade é provido, como por exemplo, VoIP, IPTV, entre outros, é assumida a mesma classe de serviço para todos os usuários, mas nem sempre esta situação é a ideal. Em muitos casos é necessário diferenciar o serviço, em função do contexto de uso. Tomando como base um exemplo de um paciente que é atendido por teleconsulta, usando o VoIP, a QoS dada deveria ser diferente de um usuário que usa o VoIP para conversar com um amigo, por exemplo. No entanto, definir políticas com valores padronizados impede que o usuário participe e dê a sua opinião sobre a qualidade do serviço que está sendo provido.

Os provedores que se preocupam em prover serviços de acordo com a UX, tem um grande potencial em manterem-se competitivos, pois os usuários sempre esperam que o serviço seja ofertado com a melhor qualidade possível. A provisão de serviços centradas na UX é custosa, mas, necessária para que o conhecimento do nível de satisfação do usuário seja correlacionado com outros aspectos, incluindo a rede, o contexto, conteúdo e fatores humanos. Contudo, existe uma grande complexidade em mapear e relacionar todos esses fatores para comporem uma KB. Uma vez conhecida a UX e suas relações, o mecanismo de inteligência será capaz de aprender a UX no uso do serviço e ofertar serviços diferenciados de acordo com o contexto.

Este trabalho deu os primeiros passos rumo a uma abordagem de provisão de serviços ciente da UX. E para elucidar a contribuição do mesmo são descritos os objetivos traçados *versus* alcançados.

*Objetivo 1 - Selecionar uma abordagem semântica para a representação do conhecimento da QoE.*

O embasamento conceitual do segundo capítulo contemplou o primeiro objetivo proposto, uma vez que a documentação do mesmo permitiu identificar que a ontologia é uma alternativa viável para a representação do conhecimento da QoE, pois além de fornecer interoperabilidade, provê uma estrutura conceitual comum sobre a qual foi possível desenvolver a KB da QoE, compartilhá-la com todas as entidades envolvidas na provisão do serviço, sendo possível reutilizá-la em outros contextos. Além disso, a ontologia juntamente com as regras semânticas, permitem a construção de sistemas e aplicações inteligentes. No segundo capítulo, além da abordagem conceitual e técnica, foram apresentadas as principais metodologias para a engenharia de ontologias.

*Objetivo 2 – Identificar os fatores de influência na QoE, formas de mensuração e representação do conhecimento, gerando uma taxonomia das dimensões a partir das pesquisas existentes.*

No terceiro capítulo foi realizada uma revisão sistemática da literatura para evidenciar o domínio de aplicação da QoE, as bases conceituais das propostas, as formas de mensuração, as dimensões que foram consideradas, como o conhecimento tem sido representado e quais arquiteturas foram utilizadas para avaliar a QoE. Após toda a contextualização do segundo, terceiro e quarto capítulos foi possível compreender o contexto da provisão de serviços, os tipos de conhecimentos, a abordagem interdisciplinar como condição necessário para o estudo e as abordagens tecnológicas emergentes para a gerenciamento da QoE.

Através das pesquisas realizadas ao longo do capítulo 3 ficou evidente que as propostas que envolvem a experiência do usuário devem ir além da tecnologia e conhecimento do perfil. A experiência do usuário deve ser tratada de forma mais ampla, incluindo aspectos cognitivos, estéticos e necessidades hedônicas. Essa abordagem só é possível se tratada de forma interdisciplinar.

A interdisciplinaridade decorre da interação de elementos e busca de artefatos de diferentes disciplinas, em prol do mesmo objeto de estudo. Ao analisar os trabalhos apresentados ao longo deste documento, observou-se que a proposição dos modelos existentes tem sido viabilizada, mesmo que de forma implícita, pela aplicação de pelo menos duas das seguintes bases teóricas: Psicologia, Sociologia, Ciência da Cognição, Design de Interface, Engenharia do Conhecimento, Computação Sensível ao Contexto e TIC. Nessa perspectiva e com base



na análise dos conceitos e propostas de QoE apresentadas, foi possível gerar a taxonomia das dimensões de influência na QoE.

A análise da taxonomia permitiu observar que nem todas as dimensões se aplicam ao mesmo propósito e irão depender do tipo de serviço que se pretende avaliar, ou seja, os fatores de influência de um serviço VoIP, por exemplo, diferem de um serviço de buscador de hotéis, pois o sucesso na UX do VoIP está condicionado a QoS do sistema de comunicação, enquanto que o sucesso da UX ao usar o buscador de hotéis, pode estar atrelada a ergonomia e usabilidade do sistema; a qualidade do conteúdo disponibilizado (e.g., atual, preciso, significado e valor do conteúdo para o usuário); tempo necessário para o usuário completar a tarefa (encontrar um hotel com as características desejadas); tempo de resposta do sistema para localizar o hotel com as restrições do usuário.

*Objetivo 3 - Especificar um modelo de representação do conhecimento da QoE, com base na taxonomia gerada.*

O modelo de representação do conhecimento da QoE, proposto através de uma ontologia, foi construído usando o processo metodológico proposto por Rautenberg e Todesco (2009). O modelo de representação do conhecimento consistiu na construção de uma KB com formalismo semântico para que o mecanismo de inteligência da arquitetura proposta aprenda a UX no uso do serviço, usando o conhecimento da KB.

Considerando que os testes com usuários são difíceis de realizar, manter e tem um alto custo, a KB foi construída com as fontes de conhecimentos já existentes, incluindo o conhecimento explícito dos especialistas; resultados de testes realizados e validados; pesquisa de campo; imersão na literatura. A partir desse conhecimento formalizado, novos fatos foram derivados usando as regras semânticas, como foi o caso do motor semântico, que usou as regras para inferir e classificar alertas de degradação da QoE, com base no conhecimento da KB.

*Objetivo 4 - Elaborar um mecanismo que faz uso do modelo, projetando-o numa arquitetura orientada a UX, com o uso da SDN.*

O modelo de representação do conhecimento da QoE foi incorporado numa plataforma de prestação de serviços orientada as necessidades do usuário. Para isso foi apresentada uma arquitetura conceitual, que permite uma abordagem voltada para atender a UX. A arquitetura, estruturada em cinco camadas (contexto, serviço, gerenciamento, controle e dados) segue o paradigma da SDN (ONF, 2012), uma das promessas para a provisão de serviços na Internet do Futuro.

A contribuição desta tese para com a arquitetura consistiu, principalmente, da aplicação de gerenciamento da QoE (*QoE-ManApp*), projetado com o *loop* de controle do MAPE-K (IBM, 2005) e incorporado na camada de gerenciamento, no módulo de Gerenciamento da QoE. A aplicação interage com as aplicações de gerenciamento e de controle, com o objetivo de aprender a experiência do usuário no uso de um serviço, verificar violações de políticas de QoS na KB e autonomicamente solicitar à uma aplicação de controle a restauração da QoE.

Após a apresentação das três contribuições do modelo de gerenciamento da QoE (taxonomia da QoE, modelo de representação do conhecimento e a arquitetura), o segundo objetivo foi concluído. O terceiro objetivo foi atendido completamente, pois todas as questões de competências foram respondidas. O quarto objetivo também foi contemplado na sua totalidade, uma vez que o modelo de representação do conhecimento, juntamente com as regras semânticas, foi inserido na aplicação de gerenciamento da QoE. A incorporação do modelo na camada de gerenciamento da QoE, permitiu o compartilhamento do conhecimento usado pelos mecanismos da SDN (aplicações de gerenciamento e controle). O conhecimento disponível da KB, juntamente com a base de regras possibilitou a classificação dos alertas das degradações da QoE e assim, as funcionalidades MAPE-K (da aplicação de gerenciamento da QoE) atuaram de forma autônoma para restaurar a QoE.

*Objetivo 5 - Idealizar um cenário para a avaliação da abordagem proposta num Ambiente de Vida Assistido (AAL - Ambient Assisted Living).*

No sexto capítulo foi apresentado o cenário, o ambiente experimental e toda a estrutura utilizada para emular a provisão de serviços entre um AAL e uma unidade de atendimento remota. A solução proposta foi experimentada num ambiente de SDN. Os experimentos foram realizados nas mesmas condições, tanto com a abordagem proposta, quanto com a abordagem nativa. Com os resultados experimentais foi realizada a mesma análise estatística, contemplando a estatística descritiva, intervalos de confiança e teste de hipóteses. A inferência estatística, com aplicação do *Test t*, permitiu evidenciar com intervalo de confiança de 95%, que todos os serviços foram providos com qualidade superior ao do algoritmo nativo do SDNC *Floodlight*. Portanto, com base em todas as análises realizadas e apresentadas no sexto capítulo tem-se evidências suficientes para afirmar que o modelo de gerenciamento da QoE pode ser usado em

plataformas de provisão de serviços cientes de contexto. Destaca-se ainda, que embora, o cenário tenha sido idealizado para o serviço de *eHealth*, o modelo é genérico e pode ser experimentado na provisão de outros serviços.

## 7.2 TRABALHOS FUTUROS

Os componentes do modelo de gerenciamento da QoE podem ser melhorados em vários aspectos. Uma das melhorias refere-se a modelagem dos dados da ontologia. Após a concretização da ontologia, dado o grande número de classes e diferentes funções (modelagem da QoE em dimensões, experiência do usuário, monitoramento de parâmetros de QoS, classificação de alertas, políticas de QoS) sugere-se considerar a viabilidade da mesma ser estruturada em sub modelos, assim, como proposto em Laghari et al. (2010) e Latré et al. (2008).

A dimensão “fatores humanos” da taxonomia proposta carece de um estudo aprofundado, tanto nos fatores objetivos quanto nos subjetivos. Uma vez que o pesquisador tenha interesse de avaliar em profundidade esses fatores, sugere-se que a abordagem interdisciplinar seja preservada, pelos motivos já apresentados no quinto capítulo. Como ponto de partida pode-se buscar aparatos em teorias da Psicologia, IHC e Design de Interface para o estudo dos fatores subjetivos e, buscar o apoio nas bases conceituais da Ciência da Cognição, Psicologia Cognitiva e Modelos Mentais para o estudo dos fatores objetivos.

Outra questão que merece ser aprofundada refere-se a experiência do usuário no uso do serviço. Para que o mecanismo de inteligência possa inferir a UX, a KB deve ser efetivamente populada com os *feedbacks* dos usuários correlacionados com o contexto e não numa KB com conhecimento pré-definido por especialistas. Para isso, duas contribuições tecnológicas devem ser implementadas e colocadas em produção (módulo de captura do contexto do usuário e módulo de captura da experiência do usuário). Assim, a verdadeira KB ciente da UX pode ser construída. E para que o mecanismo de inteligência possa efetivamente aprender a UX, algoritmos de aprendizagem de máquinas, além do algoritmo de regressão linear, devem ser adotados, fazendo uma avaliação de qual deles é o mais assertivo, não descartando a possibilidade de usar uma combinação de algoritmos.

Na parte experimental foram evidenciadas demandas tanto para a Engenharia do Conhecimento quanto para a Ciência da Computação, referente a integração do modelo de conhecimento com os componentes da SDN. O modelo de conhecimento proposto usa a linguagem OWL,

no entanto, a troca de informações entre os componentes da arquitetura SDN usam, em sua grande maioria API REST, programadas em JSON, por exemplo. Assim, para minimizar o número de adaptações tecnológicas na integração dos componentes (ontologia e aplicações de controle e gerenciamento da SDN) deve-se considerar o uso da ontologia em JSON. No *framework* Protégé uma ontologia especificada em OWL pode ser transformada em ontologia JSON-DL, mas é preciso fazer uma análise quanto a preservação da expressividade da ontologia, bem como a estrutura hierárquica da mesma.

Outro problema técnico visualizado como demanda para a Ciência da Computação, refere-se ao intercâmbio de informações entre o módulo de controle da SDN que gerencia o fluxo, mas que no entanto, depende do conhecimento da QoE/QoS persistido na KB controlada pela aplicação de gerenciamento da QoE.

Como a arquitetura proposta foi avaliada num AAL, muito embora, o tráfego tenha sido feito com o uso de VPN, nenhum protocolo de segurança da informação adicional foi pesquisado, mas sabe-se que a rede precisa adotar protocolos padronizados para que os fluxos de informações de *eHealth* possam ser transmitidos. Outros aspectos que devem ser considerados na expansão da proposta referem-se a escalabilidade dos AAL e a mobilidade dos usuários.

Na avaliação do modelo, realizada no sexto capítulo, observou-se que a taxa de descarte de pacotes ainda precisa ser reduzida, portanto, novos experimentos, com alterações do ambiente experimental, devem ser realizados para verificar se o algoritmo proposto é capaz de prover tal melhoria visando obter um padrão de comportamento. Quanto a este aspecto estamos produzindo novas pesquisas para a publicação de um material conclusivo, incluindo esta demanda.

A KB usada nos experimentos foi construída e avaliada usando os parâmetros de QoS e QoC, pois estes são parâmetros diretamente relacionados com QoE. Porém, por ser genérico, o modelo pode ser utilizado e avaliado em outros contextos de provisão de serviços. E com isso novos parâmetros, notoriamente aqueles relacionados a fatores humanos, devem ser analisados.

Partindo do pressuposto que os AAL já fazem parte da realidade de muitas pessoas (idosos e pacientes que necessitam de monitoramento remoto a partir de suas residências) e que num futuro próximo múltiplos AALs irão formar uma rede de assistência a saúde das pessoas, uma intenção de trabalho futuro da tese consiste em prover uma infraestrutura de agregação e armazenamento de dados em bases noSQL na nuvem. Portanto, em Vanelli et al. (2017b) iniciamos as pesquisas

nesta direção, propondo uma infraestrutura para agregar e armazenar dados de diferentes fontes de dispositivos da IoT de um AAL para a nuvem.

E por último, em Vanelli et al. (2017c) apresentamos uma proposta promissora de provisão de serviço em nuvem. O artigo apresenta uma arquitetura para robôs como um serviço (RaaS) na nuvem, onde um protótipo robótico foi desenvolvido e controlado para avaliar métricas de confiabilidade e qualidade do serviço do protocolo MQTT (*Message Queuing Telemetry Transport*). A arquitetura pode acomodar algumas aplicações, tais como, utilizar os robôs no auxílio de pacientes de um AAL (alertar o paciente o horário de tomar o medicamento); alertar um sistema de controle sobre possível queda de um idoso; detectar a entrada e saída de um paciente num determinado espaço físico do AAL.



## REFERÊNCIAS

ABEL, Mara; FIORINI, Sandro Rama. **Uma revisão da Engenharia do Conhecimento: Evolução, Paradigmas e Aplicações**. IJKEM, Florianópolis, v. 2, n. 2, p.1-35, mar. 2013. Quadrimestral.

ADRICHEM, N. L. M. van; DOERR C.; KUIPERS F. A. **OpenNetMon: Network monitoring in OpenFlow Software-Defined Networks**, 2014. IEEE Network Operations and Management Symposium (NOMS), Krakow, 2014, pp. 1-8.

AJZEN, I. **The Theory of Planned Behavior**. Organizational Behavior and Human Decision Processes, California, v. 50, n. 2, 1991.

ALMEIDA, M. B; SOUZA, R. R. **Avaliação do espectro semântico de instrumentos para organização da informação**, Encontros Bibli - Revista Eletrônica de Biblioteconomia e Ciência da Informação, v. 16, p. 25-50, 2011.

ALMES, G.; KALIDINDI, S.; ZEKAUSKAS, M. **A One-way Packet Loss Metric for IPPM**. EUA, 1999a. IETF RFC. Disponível em: <<http://tools.ietf.org/html/rfc2680>>. Acesso em: 04 jul. 2013.

ALMES, G.; KALIDINDI, S.; ZEKAUSKAS, M. **A One-way Delay Metric for IPPM**. EUA, 1999b. IETF RFC. Disponível em: <<http://tools.ietf.org/html/rfc2679>>. Acesso em: 10 jul. 2013.

ALRESHOODI, M. and WOODS, J. **Survey on QoE\QoS Correlation Models For Multimedia Services**, in International Journal of Distributed and Parallel Systems (IJDPS) Vol.4, No.3, May 2013.

AL-SHABIDI A.; MCCAULEY, M. **POX-Wiki**. 2015. Disponível em: <<https://openflow.stanford.edu/display/ONL/POX+wiki>>. Acesso em: 12 ago. 2014.

AMEIGEIRAS, P.; RAMOS-MUNOZ, J. J.; NAVARRO-ORTIZ, J.; MOGENSEN, P.; LOPEZ-SOLER, J. M. **QoE oriented cross-layer design of a resource allocation algorithm in beyond 3G systems**. Comput. Commun. 33, 5 (March 2010), 571-582. Comput. Commun. 33 (5), 571-582.

ASSUITI, M.; VOLPATO, F.; SILVA, Madalena Pereira da; DANTAS, M. A. R. **A Software-Defined Network Configuration Providing Differentiated QoS to an eHealth Environment**. In: Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, PDPTA, 2016. WorldComp 2016. Las Vegas: CSREA Press, 2016. p. 237-243.

BABIARZ, J.; CHAN, K.; BAKER, F. **Configuration Guidelines for DiffServ Service Classes**. EUA, 2006. IETF RFC. Disponível em: <<http://tools.ietf.org/html/rfc4594>>. Acesso em: 20 fev 2013.

BARAKOVIĆ, S.; J. BARAKOVIĆ, J.; BARAKOVIĆ, H. **QoE dimensions and QoE measurement of NGN services**, in Proceedings of the 18th Telecommunications Forum (TELFOR '10), Belgrade, Serbia, November 2010.

BARAKOVIĆ, Sabina; SKORIN-KAPOV, Lea. **Survey and Challenges of QoE Management Issues in Wireless Networks**. Journal of Computer Networks and Communications, vol. 2013, Article ID 165146, 28 pages, 2013.

BARI, M. F.; CHOWDHURY, S. R.; AHMED, R.; BOUTABA, R. (2013), **PolicyCop: An Autonomic QoS Policy Enforcement Framework for Software Defined Networks**, Future Networks and Services (SDN4FNS), IEEE SDN for, vol., no., pp.1,7, 11-13.

BELLAVISTA, P. et al. **A Survey of Context Data Distribution for Mobile Ubiquitous Systems**, ACM Computing Surveys, vol. 44, no. 4, pp. 1–45, 2012.

BERGMAN, Mike. **Metamodeling in Domain Ontologies**. 2010. Disponível em: <<http://www.mkbergman.com/913/metamodeling-in-domain-ontologies/>>. Acesso em: 12 jul. 2013.

BERNERS-LEE, T., HENDLER, J.; O. LASSILA, O. **The Semantic Web**. *Scientific American*, v.284, n.5, p.34-43. 2001.

BLAKE, S. et al. **An Architecture for Differentiated Services**. EUA, 1998. IETF RFC. Disponível em: <<http://tools.ietf.org/html/rfc2475>>. Acesso em: 27 fev. 2013.



BMIR. **The Protégé Ontology Editor and Knowledge Acquisition System**. Disponível em: <<http://protege.stanford.edu/>>. Acesso em: 20 mar. 2013.

BONNIE E.; David, J.; KIERAS E. **Using GOMS for user interface design and evaluation: which technique?** ACM Transactions on Computer-Human Interaction (TOCHI), v.3 n.4, p.287-319, Dec. 1996.

BORGES, Fábio Roberto; VEIGA, Ricardo Teixeira; GONÇALVES FILHO, Cid; FERNANDES, Izabella Bueno; TORRES JÚNIOR, Noel. **Qualidade em serviços e gestão da experiência do cliente no comércio eletrônico**. Pensamento Contemporâneo em Administração, ISSN 1982-2596, v. 8, n. 2, 2014.

BORST, Willem N. **Construction of Engineering Ontologies for Knowledge Sharing**

**and Reuse**. PhD Thesis, University of Twente, Enschede, 1997.

Disponível em:

<<http://www.ub.utwente.nl/webdocs/inf/1/t0000004.pdf>> Acesso em: 12 mar. 2016.

BRADEN, R.; CLARK, D.; SHENKER, S. **Integrated Services in the Internet Architecture: an Overview**. 1994. IETF RFC. Disponível em: <<https://tools.ietf.org/html/rfc1633>>. Acesso em: 12 fev 2013.

BRICKLEY, Dan; MILLER, Libby. **FOAF Vocabulary Specification 0.99**. 2000. Disponível em: <<http://xmlns.com/foaf/spec/>>. Acesso em: 14 abr. 2014.

BRICKLEY, Dan. **W3C Semantic Web Interest Group: Basic Geo (WGS84 lat/long) Vocabulary**. 2003. Disponível em: <<https://www.w3.org/2003/01/geo/>>. Acesso em: 4 maio 2014.

BROEKSTRA, J.; KAMPMAN, A.; HARMELEN, F. V. **Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema**, Proceedings of the First International Semantic Web Conference, 54-68, 2002.

BROOKS, P.; HESTNES, B. **User measures of quality of experience: why being objective and quantitative is important**, IEEE Network, vol. 24, no. 2, pp. 8-13, 2010.

BUCHANAN, B. e E. SHORTLIFFE. **Rule-based expert systems: the MYCIN experiments**. Reading: Addison-Wesley. 1984

CALLET, Le, MOLLER S., PERKIS A., Eds., **Qualinet White paper on Definitions of Quality of Experience (QoE)**. May 2012. Disponível em: <<http://www.qualinet.eu/>>. Acesso em 12 mai. 2013.

CARDONE, G. et al. **Socio-technical awareness to support recommendation and efficient delivery of IMS-enabled mobile services**, IEEE Communications Magazine, June 2012.

CAVARO-MENARD, C.; ZHANG-GE, L.; CALLET, P. Le. **QoE for Telemedicine: Challenges and Trends**. Conference on Applications of Digital Image Processing, Aug 2013, San Diego, United States. 8856, 2013.

CÉ JÚNIOR, J., PRUDÊNCIO, A.C., WILLRICH, R. and da SILVA, M.P. 2013. **A semantic approach for QoS specification of communication services using QoE parameters**. Journal of the Brazilian Computer Society. 19, 3 (Sep. 2013), 207–221

CHEN, G.; KOTZ, D. **A survey of context-aware mobile computing research**, Hanover: NH, USA, 2000.

CHEN, K. T.; HUANG, C.Y.; HUANG, P. and Lei. C. L. **Quantifying Skype user satisfaction**. In Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM). ACM, 2006.

CHEN, K.; TU, C.; XIAO W. **Oneclick: A framework for measuring network quality of experience**. In INFOCOM 2009, IEEE International Conference on Computer Communications., pages 702–710.

CHEN, K. T.; CHANG, C. J.; WU, C. C.; CHANG, Y. C.; LEI, C. L. **Quadrant of euphoria: a crowdsourcing platform for QoE assessment**, IEEE Network, vol. 24, no. 2, pp. 28–35, 2010.

CHIMENTO, P.; ISHAC, J. **Defining Network Capacity**. EUA, 2008. IETF RFC. Disponível em: <<http://tools.ietf.org/html/rfc5136>>. Acesso em: Jul. 2013.

CHU, Y. and GANZ, A. **A Mobile Teletrauma System using 3G Networks**, IEEE Trans. Inform. Tech. Biomed., vol. 8, no. 4, pp. 456–562, 2004.

COLE, R.G., ROSENBLUTH, J.H., 2001. **Voice over IP performance monitoring**. ACM SIGCOMM Comput. Commun. Rev. 31 (2), 9–24.

COSTA, A. R. C.; BORTOLATO, M. M.; PERASSI, R. L. **Estudo semiótico da interface gráfico-digital interativa Picasso Head**, Estudos Semióticos, vol. 7, n. 1, p. 114-125, jun 2011.

CRESWELL, J. W. **Projeto de pesquisa: métodos qualitativo, quantitativo e misto**. Tradução Magda Lopes; consultoria, supervisão e revisão técnica desta edição Dirceu da Silva. – 3. Ed. – Porto Alegre: Artmed, 2010. 296p.

COLLINS, KJ. (1986). **Low indoor temperatures and morbidity in the elderly**, Oxford Journals Medicine Age and Ageing, v. 15, p. 212-220.

D' AGOSTINI, C. S.; FILETO, R. **Capturing Users' Preferences and Intentions in a Semantic Search System**, in: SEKE - International Conference on Software Engineering & Knowledge Engineering, 2009, Boston: 2009. p. 587-591.

DAVENPORT, T. H. **Ecologia da informação: por que só a tecnologia não basta para o sucesso na era da informação**. São Paulo: Futu-ra, 1998.

DAVIS, F. D. **Perceived Usefulness, Perceived Ease of Use and User Acceptance of Information Technology**, MIS Quarterly, 13,3 (September) (1989), 319-340.

DEMICHELI, C.; CHIMENTO, P. **IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)**. EUA, 2002. IETF RFC. Disponível em: <<http://tools.ietf.org/html/rfc3393>>. Acesso em: 14 abr. 2013.

DE MOOR, K. et al. **Bridging troubled water: Quality of experience in a mobile media context.** In *Terena Networking Conference*, Bruges, 2008.

DE MOOR, K.; KETYK'Ó, I.; JOSEPH W. et al., **Proposed framework for evaluating quality of experience in a mobile, testbedoriented living lab setting**, *Mobile Networks and Applications*, vol. 15, no. 3, pp. 378–391, 2010.

DEY, A. K. **Providing Architectural Support for Building Context-Aware Applications**, 2000. Georgia Institute of Technology, 2000.

DINDA, P. A.; MEMIK, G.; DICK, R. P. et al., **The user in experimental computer systems research**, in *Proceedings of the Workshop on Experimental Computer Science*, SanDiego, Calif, USA, June 2007.

DUDA, R. O., P. E. HART, N. J. NILSSON e G. L. Sutherland. **Semantic network representations in rule-based inference systems.** In: D. A. Waterman e F. Hayes-Roth (Ed.). *Pattern Direct Inference Systems*. New York: Academic Press, 1978. Semantic network representations in rule-based inference systems, p.203-221.

ETSI, TR 102 643 V1.0.2 (2010-01), Technical Report, Human Factors (HF); **Quality of Experience (QoE) requirements for real-time communication services.** 2010.

ETSI Guide 202 843 V1.1.2, **Definitions and Methods for Assessing the QoS Parameters of the Customer Relationship Stages Other than Utilization**, July 2011.

FADEL, L. M. 2014. **Experience-Centered Web Design Model.** In *Proceedings of the Third International Conference on Design, User Experience, and Usability. User Experience Design for Diverse Interaction Platforms and Environments - Volume 8518*, Aaron Marcus (Ed.), Vol. 8518. Springer-Verlag New York, Inc., New York, NY, USA, 92-103.

FALLON, L.; HUANG, Y. **GSQR: A generic service quality reporting protocol for terminals**, *Integrated Network Management*

(IM), IFIP/IEEE International Symposium on, Dublin, 2011, pp. 1122-1128.

FALLON, L.; O'SULLIVAN, D. **Using a semantic knowledge base for communication service quality management in Home Area Networks**, 2012 IEEE Network Operations and Management Symposium, Maui, HI, 2012, pp. 43-51.

FIEDLER, M., KILKKI K., REICHL, P. **From Quality of Service to Quality of Experience**. Dagstuhl Seminar Proceedings, Executive Summary, 2009.

FIEDLER, M.; HOSSFELD, T. e TRAN-GIA, P. **A generic quantitative relationship between quality of experience and quality of service**, IEEE Network, vol. 24, no. 2, pp. 36-41, March-April 2010.

FIRESTONE, J. M.; McELROY, M. W. **Key issues in the new knowledge management**. Amsterdam: Butterworth-Heinemann; Boston, MA: KMCI, 2001.

FISHBEIN, M.; AJZEN I. **Belief, Attitude, Intention and Behavior: An Introduction to Theory and Research**, Addison-Wesley Publishing Company, 1975.

FITZSIMMONS, J. A.; FITZSIMMONS, M. J. **Administração de serviços: operações, estratégias e tecnologia da informação**. Tradução de Jorge Ritter. 4 ed. Porto Alegre: Bookman, 2005.

FLEISCHMANN, Ana Marilza Pernas; DANTAS, Mario Antono Ribeiro. **Ontologias Aplicadas à Descrição de Recursos em Grids Computacionais**. 2004. 109 f. Dissertação (Mestrado) - Curso de Ciência da Computação, Informática e Estatística, Ufsc, Florianópolis, 2004. Disponível em: <<https://repositorio.ufsc.br/bitstream/handle/123456789/87777/208061.pdf?sequence=>>>. Acesso em: 04 set. 2016.

FLEISCHER, S.L.M. and ASNANI, G.C (1978). **The influence of weather on asthma in Nairobi**, International Journal of Biometeorology, v. 22, p. 263-270.

FLORES, C. D., C. L. HOHER, M. LADEIRA e R. M. VICARI. **Una Experiencia en el uso de redes probabilísticas para el diagnóstico médico : una experiencia brasileira Informatic@Médica**, v.2, n.8, March 2001, p.25-29, 2001.

GARVIN, David A. **Gerenciando a Qualidade: A visão estratégica e competitiva**. Rio de Janeiro: Qualitymark, 1992.

GHOBADIAN, Abby, SPELLER, Simon, JONES, Matthew. **Service quality: concepts and models. International Journal of Quality & Reliability Management**. UK, v.11, n. 9, p. 43-66, 1994.

GIACOBBE M., FAZIO M., CELESTI A., ABBATE T., VILLARI M. (2016) **A Scientometric Analysis of Cloud Computing and QoE Literature to Design a Cloud Platform of Experience for Digital Business**. In: Celesti A., Leitner P. (eds) *Advances in Service-Oriented and Cloud Computing. ESOC Workshops 2015. Communications in Computer and Information Science*, vol 567. Springer, Cham.

GIANESI, Irineu G. N., CORRÊA, Henrique Luiz. **Administração estratégica de serviços: operações para a satisfação do cliente**. São Paulo: Atlas, 1994. 233p.

GIMA, **Global Initiative for Asthma**. Available in <http://www.ginasthma.org/>. Access in July 2015.

GÓMEZ-PÉREZ, A.; CORCHO, O.; FERNÁNDEZ-LÓPEZ, M. **Ontologic Engineering: with examples from the areas of knowledge management, e-commerce and the semantic web**. Springer-Verlag, 2004.

GUEDES, D.; VIEIRA, L. F. M.; VIEIRA, M. A. M.; RODRIGUES, H.; NUNES, R. V. **Redes Definidas por Software: uma abordagem sistêmica para o desenvolvimento de pesquisas em Redes de Computadores**. Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos 2012 - Minicursos Livro Texto. Porto Alegre: Sociedade Brasileira de Computação, 2012, v. , p. 160-210.

GRUBER, T. R. **A translation approach to portable ontology. Knowledge Acquisition**, 5(2): 199-220, 1993.

GRUBER, T. R. **Toward Principles for the Design of Ontologies Used for Knowledge Sharing.** International Journal of Human-Computer Studies, N.43, P.907-928, 1995.

GUARINO, N. **Formal Ontology in Information Systems Formal Ontology in Information Systems**, FOIS'98. Trento, Italy: 6-8 June 1998.

GUDE, N. et al. (2008), **NOX: Towards an Operating System for Networks.** SIGCOMM Computer Communication, 1(38):105-110, 2008. 31, 32.

HEKTNER, J. M.; SCHMIDT, J. A.; CSIKSZENTMIHALYI, M. **Experience Sampling Method: Measuring the Quality of Everyday Life**, Sage Publications, 2006.

HOBBS, Jerry R.; PAN, Feng. 2004. **An ontology of time for the semantic web.** 3, 1 (March 2004), 66-85.

HOLANDA, O., et al. **JOINT: Java ontology integrated toolkit. Expert Systems with Applications**, v. 40, p. 6469-6477, 2013.

HORSTMAYER, S. (2006). **Relative Humidity Relative to What? The Dew Point Temperature a better approach**, in <http://www.shorstmeyer.com/wxfaq/humidity/humidity.html>. Access in July 2015.

HORROCKS, I., PATEL-SCHNEIDER, P. F., BOLEY, H., TABEL, S., Grosf, B., Dean, M. **SWRL: A Semantic Web Rule Language Combining OWL and RuleML.** W3C Member Submission 21 May 2004. Disponível em: <<http://www.w3.org/Submission/SWRL/>>. Acesso em: 10 jun. 2013.

HOßFELD, T.; SCHATZ, R.; EGGER S. **SOS: the MOS is not enough!**, in Proceedings of the 3rd International Workshop on Quality of Multimedia Experience (QoMEX '11), Machelen, Belgium, September 2011.

HOßFELD, T.; SEUFERT, M.; HIRTH, M.; ZINNER, T.; TRAN-GIA, P.; SCHATZ, R., **Quantification of YouTubeQoE via**

**Crowdsourcing**, in Proceedings of the IEEE International Symposium on Multimedia (ISM '11), December 2011.

HOFELD, T.; SCHATZ, R.; BIRSACK, E.W.; PLISSONNEAU, L. 2013. **Internet video delivery in YouTube: from traffic measurements to quality of experience**. In: Biersack, E., CALLEGARI, C., MATIJASEVIC, M. (Eds.), Data Traffic Monitoring and Analysis. Springer Berlin Heidelberg, 264–301.

HOFELD, T.; SEUFERT, M.; SIEBER, C.; ZINNER, T. 2014. **Assessing effect sizes of influence factors towards a QoE model for HTTP adaptive streaming**. In: Proceedings of the IEEE International Workshop on Quality of Multimedia Experience (QoMEX). pp. 111–116.

HUONG-TRUONG, THU et al. **QoE-aware Resource Provisioning and Adaptation in IMS-based IPTV Using OpenFlow**. Local & Metropolitan Area Networks (LANMAN), 2013 19th IEEE Workshop, 2013, p. 1-3.

IBM, **An architectural blueprint for autonomic computing**. IBM white paper, 2005. Disponível em: <<http://www-03.ibm.com/autonomic/pdfs/AC%20Blueprint%20White%20Paper%20V7.pdf>>. Acesso em: mar. 2014.

ICKIN, S.; WAC, K.; FIEDLER, M.; JANOWSKI, L. HONG, JIN-HYUK and DEY, A.K. **Factors influencing quality of experience of commonly used mobile applications**. Communications Magazine, IEEE, 50(4):48–56, april 2012.

IETF (2006a), **NETCONF Configuration Protocol**. 2006. Disponível em: <<http://datatracker.ietf.org/wg/netconf/>>. Acesso em: 12 dez. 2015.

IETF (2006b), **A Border Gateway Protocol 4 (BGP-4)**. 2006. Disponível em: <<http://www.ietf.org/rfc/rfc4271.txt>>. Acesso em: 14 dez. 2015.

IETF (2011a), **Operations, Administration, and Maintenance Framework for MPLS-Based Transport Networks**, 2011. Disponível em: <<http://tools.ietf.org/html/rfc6371>>. Acesso em: 15 dez. 2015.



IETF (2011b), **Extensible Messaging and Presence Protocol (XMPP): Core**. 2011. Disponível em: <<http://tools.ietf.org/html/rfc6120>>. Acesso em 17 dez. 2015.

IETF (2013), **The Open vSwitch Database Management Protocol**, 2013. Disponível em <<http://tools.ietf.org/html/rfc6371>>. Acesso em: 13 ago. 2015.

ISO – International Organization for Standardization. **Quality of Service Basic Framework** – outline. ISO / IEC JTC1 / SC21 / WG1 N1145, 1994.

ISOTANI, Seiji; BITTENCOURT, Ig Ibert. **Dados Abertos Conectados**, 2015. Núcleo de Informação e Coordenação do Ponto Br – NIC.br, Centro de Estudos sobre Tecnologia Web – CeWeb.br. Disponível em: <<http://ceweb.br/livros/dados-abertos-conectados/prefacio/>>. Acesso em: 19 ago. de 2016.

ITU-R Recommendation BS.1387, **Method for Objective Measurements of Perceived Audio Quality**, November 2001.

ITU-T Recommendation G.107, **The E-model: A Computational Model for Use in Transmission Planning**, December 2011.

ITU-T Recommendation G.107, **The E-model: A Computational Model for Use in Transmission Planning**, February 2014.

ITU-T Recommendation G.108, **Application of the E-model: A Planning Guide**, September 1999.

ITU-T Recommendation G.1010, **End-User Multimedia QoS Categories**, November 2001.

ITU-T Recommendation G.1011, **Reference Guide to Quality of Experience Assessment Methodologies**, June 2010.

ITU-T Recommendation G.1030, **Estimating End-to-End Performance in IP Networks for Data Applications**, November 2005.

ITU-T Recommendation G.1070, **Opinion Model for Video-Telephony Applications**, April 2007.

ITU-T Recommendation G.1070, **Opinion Model for Video-Telephony Applications**, July 2012.

ITU-T Recommendation G.1080, **Quality of Experience requirements for IPTV Services**, December, 2008.

ITU-T Recommendation P.563, **Single-ended Method for Objective Speech Quality Assessment in Narrow-Band Telephony Applications**, May 2004.

ITU-T Recommendation P.564, **Conformance Testing for Voice Over IP Transmission Quality Assessment Models**, November 2007.

ITU-T Recommendation P.800, **Methods for subjective determination of transmission quality**, 1996.

ITU-T Recommendation P.800.1, **MOS - Mean opinion score terminology**, Geneva, Switzerland, 2003.

ITU-T Recommendation P.862, **Perceptual Evaluation of Speech Quality (PESQ): An Objective Method for End-to-End Speech Quality Assessment of Narrow-Band Telephone Networks and Speech Codecs**, February 2001.

ITU-T Recommendation P.863, **Perceptual Objective Listening Quality Assessment**, January 2011.

ITU-T Recommendation Y.1541, **Network Performance Objectives for IP-Based Services**, May 2002.

JAIN, Sushant et al. B4: **Experience with a Globally-Deployed Software Defined WAN**. SIGCOMM'13, August 12–16, 2013, Hong Kong, China.

JOHNSTON, Robert. **The determinants of service quality: satisfiers and dissatisfiers**. International Journal of Service Industry Management. UK, v. 6, n. 5, p. 53-71, 1995.

JUMISKO-PYYKKÖ, S. and HANNUKSELA, M. M. **Does context matter in quality evaluation of mobile television?** In Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI 2008), pages 63–72. ACM.

KASSLER, A. SKORIN-KAPOV, L. DOBRIJEVIC, O. MATIJASEVIC, M. DELY, P. **Towards QoE-driven multimedia service negotiation and path optimization with software defined networking, Software**, Telecommunications and Computer Networks (SoftCOM), 2012 20th International Conference on, Split, 2012, pp. 1-5.

KHAN, A.; SUN, L.; IFEACHOR, E., 2009. **Content clustering based video quality prediction model for MPEG4 video streaming over wireless networks**. In: Proceedings of the IEEE International Conference on Communications (ICC). pp. 1– 5.

KHAN, A.; SUN, L., JAMMEH, E., IFEACHOR, E., 2010. **Quality of experience driven adaptation scheme for video applications over wireless networks**. IET Commun. 4 (11), 1337–1347.

KHIRMAN, S.; HENRIKSEN, P. **Relationship between Quality-of-service and Quality-of-Experience for Public Internet Service**, Proc. of the 3rd Workshop on Passive and Active Measurement, 2002.

KIM, H. and CHOI, S. **A study on a QoS/QoE Correlation Model for QoE Evaluation on IPTV Service**, Proc. of The 12th International Conference on Advanced communication technology, ICACT'10, pp. 1377-1382, 2010.

KIM, Sungsu. **Cognitive Model-Based Autonomic Fault Management in SDN**. 2013. 103 f. Tese (Doutorado) - Curso de Computer Science And Engineering, Electrical And Computer Engineering, Pohang University Of Science And Technology, Pohang, 2013. Disponível em:  
<[http://dpmn.postech.ac.kr/thesis/13/kiss/thesis\\_sungsu.pdf](http://dpmn.postech.ac.kr/thesis/13/kiss/thesis_sungsu.pdf)>. Acesso em: 12 maio 2014.

KUIPERS, F.; KOOIJ, R.; VLEESCHAUWER, De; BRUNNSTROM, K. **Techniques for measuring quality of experience**, *Wired/Wireless Internet Communications*, pp. 216–227, 2010.

LAGHARI K. U. R. et al. **Towards a Service Delivery Based on Customer eXperience Ontology: Shift from Service to eXperience**, R. Brennan, J. Fleck II, and S. van der Meer (Eds.): MACE 2010, LNCS 6473, pp. 51–61.

LAGHARI K. U. R. et al. **QoE aware Service Delivery in Distributed Environment**, *Advanced Information Networking and Applications (WAINA)*, 2011.

LATRÉ, S. et al., **Design for a Generic Knowledge Base for Autonomic QoE Optimization in Multimedia Access Networks**, *Network Operations and Management Symposium Workshops*, 2008. NOMS Workshops 2008. IEEE, Salvador da Bahia, 2008, pp. 335-342.

LAWRENCE, M. G. (2005) . **The Relationship between Relative Humidity and the Dewpoint Temperature in Moist Air**, *American Meteorological Society*, [s.l].

LEC-EGC-UFSC. **OntoKEM - Ontology for Knowledge Engineering and Management**. Disponível em: <<http://ontokem.egc.ufsc.br/ProjetoOntologias/index.html>>. Acesso em: 18 fevereiro 2014.

LI-YUAN, L., ZHOU, W; SONG, J. **The Research of Quality of Experience Evaluation Method in Pervasive Computing Environment**, *1st International Symposium on Pervasive Computing and Applications*, 2006.

MANHÃES, Maurício Cordeiro; VARVAKIS, Gregório. **A Inovação em serviços e o processo de criação do conhecimento: uma proposta de método para o design de serviço**. 1 v. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento, Florianópolis, 2010.

MANICA, Heloise; DANTAS, Mario Antônio Riberiro. **Modelo de recuperação e comunicação de conhecimento em emergência médica com utilização de dispositivos portáteis**. xv, [139] p. Tese (Doutorado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-graduação em Engenharia e Gestão do Conhecimento, Florianópolis, 2009.

MARTINS, P. G.; LAUGENI, F. P. **Administração da produção**. 2 ed. São Paulo: Saraiva, 2006.

MARTINS, D.; SILVA, Madalena Pereira da; DANTAS, M.A.R. **Um estudo de qualidade de serviço em ambientes SDN**. Revista Junior de Iniciação Científica em Ciências Exatas e Engenharia, v. 1, p. 23-28, 2016.

MATTÉ, Volnei Antônio. (2009). **O conhecimento da prática projetual dos designers gráficos como base para o desenvolvimento de materiais didáticos impressos**. Tese de doutorado EGC, UFSC.

MCKEOWN, N.; ANDERSON, T.; BALAKRISHNAN, H.; PARULKAR, G.; PETERSON, L.; REXFORD, J.; SHENKER, S. and TURNER, J.. 2008. **OpenFlow: enabling innovation in campus networks**. SIGCOMM Comput. Commun. Rev. 38, 2 (March 2008), 69-74.

MENKOVSKI, V.; EXARCHAKOS, G.; LIOTTA, A.; CUADRA-SANCHEZ, A. **Managing quality of experience on a commercial mobile TV platform**, International Journal on Advances in Telecommunications, vol. 4, no. 1-2, pp. 72–81, 2011.

MITRA, K.; ÅHLUND, C. and ZASLAVSKY, A. **A decision-theoretic approach for quality-of-experience measurement and prediction**. In Multimedia and Expo (ICME), 2011 IEEE International Conference on, pages 1–4, july, 2011a.

MITRA, K.; ZASLAVSKY, and C. ÅHLUND, C. **Dynamic bayesian networks for sequential quality of experience modelling and measurement**. In Smart Spaces and Next Generation Wired/Wireless Networking, volume 6869 of Lecture Notes in Computer Science, pages 135–146. Springer Berlin/Heidelberg, 2011b.

MITRA, K.; ZASLAVSKY, A. and ÅHLUND, C. **A probabilistic context-aware approach for quality of experience measurement in pervasive systems**. In Proceedings of the 2011 ACM Symposium on Applied Computing (SAC'11), pages 419–424, 2011c.

MITRA, K., ZASLAVSKY A., ÅHLUND, C. **Context-aware qoe modelling, measurement and prediction in mobile computing systems**. Mobile Computing, IEEE Transactions on, PP(99):1–1, 2014.

MITRA, K.; ZASLAVSKY, A. and ÅHLUND, C. **Context-Aware QoE Modelling, Measurement, and Prediction in Mobile Computing Systems**. IEEE Transactions on Mobile Computing Year: 2015, Volume: 14, Issue: 5, Pages: 920 – 936.

MÖLLER S. et al. **A Taxonomy of Quality of Service and Quality of Experience of Multimodal Human-Machine Interaction**, QoMEX, 2009.

MORAIS, E. A. M; AMBRÓSIO, A.P.L. **Ontologias: conceitos, usos, tipos, metodologias, ferramentas e linguagens**. Relatório Técnico - INF\_001/07, URGs, Dezembro, 2007.

MOTIK, B. et al. **OWL 2 Web Ontology Language: Profiles - 2nd edition**. W3C, 2012. Disponível em: < <https://www.w3.org/TR/owl2-profiles/> >. Acesso em: 20 ago. 2015.

NAZÁRIO, Débora Cabral; DANTAS, Mário Antonio Ribeiro. **CUIDA - Um modelo de conhecimento de qualidade de contexto aplicado aos ambientes ubíquos internos em domicílios assistidos**. 2015. 215 f. Tese (Doutorado) - Curso de Engenharia e Gestão do Conhecimento, Engenharia e Gestão do Conhecimento, Universidade Federal de Santa Catarina, Florianópolis, 2015.

NAZARIO, D. C.; SILVA, Madalena Pereira da; DANTAS, M. A. R.; TODESCO, J. L. **Quality of Context Knowledge Model Based on Ontology**. In: 12th International Conference on Management of Technology and Information Systems, 2015, São Paulo. 12th CONTECSI. São Paulo: TECSI EAC FEA USP, 2015. p. 2890-2913.

NONAKA, I. **The knowledge-creating company: how Japanese companies create the dynamics of innovation**. New York: Oxford University Press, 1995.

NONAKA, I; TAKEUCHI, H. **Criação de conhecimento na empresa**. 16. ed. Rio de Janeiro: Campus, 1997.

NORTH, Klaus. **Gestão do Conhecimento: Um Guia Prático Rumo a Empresa Inteligente**. Rio de Janeiro: Quality-mark, 2010. ISBN: 978-85-7303-900-2.

NOY, N. F.; MCGUINNESS, D. L. **Ontology Development 101: A Guide to Creating Your First Ontology**, 2008. Disponível em: <[http://protege.stanford.edu/publications/ontology\\_development/ontology101.pdf](http://protege.stanford.edu/publications/ontology_development/ontology101.pdf)>. Acesso em: 20 set. 2015.

ONF, Open Networking Foundation, **Software-Defined Networking: The New Norm for Networks**, ONF White Paper, April 13, 2012.

ONF, Open Networking Foundation. **OpenFlow Management and Configuration Protocol (OF-Config) 1.1.1**. 2014. Disponível em <<https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow-config/of-config-1-1-1.pdf>>. Acesso em: 02 jan. 2015.

OPEN SEMANTIC FRAMEWORK. **Metamodeling in Domain Ontologies**. 2014. Disponível em: <<http://www.mkbergman.com/913/metamodeling-in-domain-ontologies/>>. Acesso em: 21 jul. 2015.

OPENLINKSW. **Virtuoso Universal Server**. Disponível em: <<https://virtuoso.openlinksw.com/>>. Acesso em: 12 ago. 2015.

OWL API. **The OWL API**. Disponível em: <<http://owlapi.sourceforge.net/>>. Acesso em: 13 ago 2013.

OWL Working Group. (2004). **Web Ontology Language (OWL)**. Disponível em: <<http://www.w3.org/2004/OWL/>>. Acesso em: 5 jul 2013.

PACHECO, R. C. S.. **Coprodução em Ciência, Tecnologia e Inovação: Fundamentos e Visões**. Em Interdisciplinaridade: Universidade e Inovação Social e Tecnológica. Organizado por Joana Maria Pedro e Patrícia de Sá Freire, Curitiba, CRV., pp. 21-62. 2016.

PALADINI, Edson Pacheco. **Gestão da qualidade no processo: A qualidade na produção de bens e serviços**. São Paulo: Atlas, 1995. 286p.y.

PARASURAMAN, A., ZEITHAML, Valarie A., BERRY, Leonard L. **A conceptual model of service quality and its implications for future research**. Journal of Marketing. USA, v. 49, n. 4, p. 41-50, fall, 1985.

PARASURAMAN, A.; ZEITHAML, Valerie A.; BERRY, Leonard L. **SERVQUAL: A Multiple-Item Scale for measuring consumer perceptions of service quality**. Journal of Retailing. V. 64, p. 12-40, 1988.

PEFFERS, K. et al. **A design science research methodology for information systems research**. Journal of management information systems, v. 24, n. 3, p. 45-77, 2007. ISSN 0742-1222.

PERASSI, R.; PAVANATI, I.; NEVES JR, O. **Representação, realidade e conhecimento na mídia digital-eletrônica**, Texto Digital (UFSC), v. 6, p. 02-18, 2010.

PFAFF, B.; DAVIE, B. The open vSwitch database management protocol. [S.l.], 2013. 1-35 p. Disponível em: <<https://www.rfc-editor.org/rfc/rfc7047.txt>>.

PHEMIUS, K.; BOUET, M. **Monitoring latency with OpenFlow**. Proceedings of the 9th International Conference on Network and Service Management (CNSM 2013), Zurich, 2013, pp. 122-125.

PHILIPS. **Monitoramento de pacientes-Visibilidade dos sinais vitais a todo momento**. Disponível em: <<https://www.philips.com.br/healthcare>>. Acesso em: 01 out. 2015.



POLO, L., BERRUETA, D., (2008). **Measurement Units Ontology**, Working Draft April 2008. Disponível em <<http://idi.fundacionctic.org/muo/muo-vocab.html>>. Acesso em 10 junho de 2016.

POMBO, O. **Interdisciplinaridade e integração dos saberes**. Liinc em Revista, v.1, n.1, março 2005, p. 3 -15.

POMBO, O. **Epistemologia da interdisciplinaridade**. Revista do Centro de Educação e Letras. Unioeste, Foz do Iguaçu, v. 10 - nº 1 - p. 9-40, 2008.

POPADIUK, S.; CHOO, C. W. **Innovation and knowledge creation: how are these concepts related?**. International Journal of Information Management, v. 26, n. 4, p. 302-312, Aug. 2006.

PPGEGC/UFSC. **Engenharia e Gestão do Conhecimento**. Disponível em: <<http://www.egc.ufsc.br/pos-graduacao/>>. Acesso em: 10 agosto 2015

PRESSMAN, Roger S. **Engenharia de Software**. Mc Graw Hill, 6 ed, Porto Alegre, 2010.

PROJECT FLOODLIGHT (2014), **Open Source Software for Building Software-Defined Networks**, <http://www.projectfloodlight.org/floodlight/>, April.

QUINTELA, Hélder. (2005). **Sistemas de Conhecimento Baseados em Data Mining: Aplicação à Análise da Estabilidade de Estruturas Metálicas**. Dissertação de Mestrado. Universidade do Minho.

RAUTENBERG, Sandro; TODESCO, José Leomar. **Modelo de conhecimento para mapeamento de instrumentos da gestão do conhecimento e de agentes computacionais da engenharia do conhecimento baseado em ontologias**. 2009. 238 f. Tese (Doutorado) - Curso de Engenharia e Gestão do Conhecimento, Engenharia e Gestão do Conhecimento, Universidade Federal de Santa Catarina, Florianópolis, 2009.

RAUTENBERG, Sandro; TODESCO, José L.; GAUTHIER, Fernando A. O. **Processo de desenvolvimento de ontologias: uma proposta e uma ferramenta**. Rev. Tecnol., Fortaleza, v.30, n.1, p. 133-144, jun 2009.

REICHL, P.; EGGER, S.; SCHATZ, R. e D'ALCONZO, A. **The logarithmic nature of QoE and the role of the Weber-Fechner law in QoE assessment**, in Proceedings of IEEE International Conference on Communications (ICC '10), Cape Town, South Africa, May 2010.

REICHL, P., TUFFIN, B. e SCHATZ, R. **Logarithmic laws in service quality perception: where microeconomics meets psychophysics and quality of experience**, Telecommunication Systems Journal, vol. 55, no. 1, pp. 1–14, 2011.

HORSTMAYER, S. (2006). **Relative Humidity Relative to What? The Dew Point Temperature a better approach**, in <http://www.shorstmeyer.com/wxfaq/humidity/humidity.html>. Access in July 2015

ROSEN, E.; VISWANATHAN, A.; CALLON, R. (2001) **Multiprotocol Label Switching Architecture (MPLS)**, IETF RFC 3031.

ROSSATTO, M. A. **Gestão do conhecimento: a busca da humanização, transparência, socialização e valorização do intangível**. Rio de Janeiro: Interciência, 2002.

ROTHENBERG, C. E.; NASCIMENTO, M. R.; SALVADOR, M. R.; MAGALHÃES, M. F. **OpenFlow e redes definidas por software: um novo paradigma de controle e inovação em redes de pacotes**. Cad. CPqD Tecnologia, Campinas, v. 7, n.1, p. 65-76, jul. 2010/jun. 2011.

RUBINO, G.; TIRILLY, P.; VARELA, M. **Evaluating users satisfaction in packet networks using random neural networks**. In Artificial Neural Networks, ICANN 2006, volume 4131 of Lecture Notes in Computer Science, pages 303–312. Springer Berlin, 2006.

RYU SDN Framework Communit. **Component-based software defined networking framework Build SDN Agilely**, 2014. Disponível em: <<http://osrg.github.io/ryu/index.html>>. Acesso em: 04 abr 2014.

SANTOS, R. B.; RIBEIRO, T. R. CESAR, de A. C. **A network monitor and controller using only OpenFlow**. Network Operations and Management Symposium (LANOMS), 2015 Latin American, Joao Pessoa, 2015, pp. 9-16.

SAVI, Rafael; ULBRICH, Vania Ribas. **Avaliação de jogos voltados para a disseminação do conhecimento**, [238] p. Tese (Doutorado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento, Florianópolis, 2011.

SAWSDL, 2007. Semantic Annotations for WSDL and XML Schema (SAWSDL), W3C Recommendation 28 August 2007. Disponível em: <<http://www.w3.org/TR/sawSDL/>>. Acesso em: 05 nov. 2013.

SCADABR. Disponível em <<http://www.scadabr.com.br/>> Acesso em: 20 de maio de 2015.

SCHADOW, G.; MCDONALD, C. J. The Unified Code for Units of Measure. 2009. Disponível em: <<http://unitsofmeasure.org/>>. Acesso em 12 abril 2016.

SCHMID, B. F.; STANOEVSKA, K. **Knowledge media: An innovative concept and technology for knowledge management in the information age**. Proceedings of the 12th Biennial International Telecommunications Society Conference - Beyond Convergence. Stockholm, Sweden: IST, 1998.

SCHREIBER, A. T. **The KADS approach to knowledge engineering. Knowledge Acquisition**. [S.l.], 4: 1-4, 1992.

SCHREIBER, G., H. AKKERMANS, A. ANJEWIERDEN, R. D. HOOG, N. SHADBOLT, W. V. D. VELDE e B. WIELINGA. **Knowledge engineering and management: The CommonKADS Methodology**, Cambridge: The MIT Press. 2000.

SCHREIBER, G., H. AKKERMANS, et al. **Knowledge Engineering and Management: The CommonKADS Methodology**. MIT Press. Cambridge, 2002.

SCHULZRINNE, H.; CASNER, S.; FREDERICK, R.; JACOBSON, V. (2003, July). **RTP: A Transport Protocol for Real-Time Applications**, <https://www.ietf.org/rfc/rfc3550.txt>.

SEUFERT, M., EGGER, S., SLANINA, M., ZINNER, T., HOßFELD, T., TRAN-GIA, P., 2015. **A survey on quality of experience of HTTP adaptive streaming**. IEEE Commun. Surv. Tutor. 17 (1), 469–492.

SILVA; Daniela Lucas da; SOUZA, Renato Rocha; ALMEIDA, Maurício Barcellos. **Ontologias e vocabulários controlados: comparação de metodologias para construção**. Ci. Inf., Brasília, v. 37, n. 3, p. 60-75, set./dez. 2008.

SILVA, E. L. da; MENEZES, E. M. **Metodologia da pesquisa e elaboração de dissertação**. 3. ed. rev. atual. Florianópolis: Laboratório de Ensino a Distância da UFSC, 2001

SILVA, Madalena P. da; DANTAS, M.A.R. **Modelo de Qualidade de Experiência à Gestão do Conhecimento em Empresas Fornecedoras de Serviços Convergentes**. In: Congresso Internacional Conhecimento e Inovação, 2013, Porto Alegre. CIKI, 2013, 2013.

SILVA, Madalena Pereira da; DANTAS, M. A. R.; GONCALVES, A. L. **Uma Abordagem Interdisciplinar para a Proposição de um Modelo de Qualidade de Experiência em Serviços Multimídia**. In: Fernando Ostuni Gauthier; Selvino Assmann; Silvia Maria Puentes Bentancourt; Micheline Guerreiro Krause; Maricel Karina López Torres; Fernanda Martinhago; Jair Zandoná; Silvia Regina Pochmann de Quevedo. (Org.). **Interdisciplinaridade Teoria e Prática**. 1ª ed. Florianópolis: UFSC/EGC, 2014, v. 1, p. 401-430.

SILVA, Madalena Pereira da; DANTAS, M. A. R.; GONCALVES, A. L. **An Interdisciplinary Approach to Proposition of a Multidimensional Model of Quality of Experience**. In: **12th International Conference on Management of Technology and Information Systems**, 2015, São Paulo. 12th CONTECSI. São Paulo: TECSI EAC FEA USP, 2015. p. 4869-4876, 2015a.

SILVA, Madalena P. da; DANTAS, M. A. R.; GONCALVES, A. L. **An architecture with autonomic and semantic features for QoE management in SDN.** In: MODA 2015: Management of Ontology-Driven Architectures, 2015, Hammamet. MODA, 2015b.

SILVA, Madalena P.; DANTAS, Mario A. R.; GONÇALVES, Alexandre L.; PINTO, Alex R. **A Managing QoE Approach for Provisioning User Experience Aware Services Using SDN.** In: the 11th ACM Symposium, 2015, Cancun. Proceedings of the 11th ACM Symposium on QoS and Security for Wireless and Mobile Networks - Q2SWinet '15. v. 1. p. 51-58, 2015c.

SILVA, Madalena P. da; GONCALVES, A. L.; DANTAS, M. A. R.; Vanelli, B.; MANERICHI, G.; SANTOS, S. A. R. D.; FERRANDIM, M.; A.R. Pinto. **Implementation of IoT for Monitoring Ambient Air in Ubiquitous AAL Environments.** In: V Brazilian Symposium on Computing Systems Engineering, 2015, Foz do Iguacu. V Brazilian Symposium on Computing Systems Engineering. v. 1, 2015d.

SILVA, Madalena Pereira da; DANTAS, M. A. R.; GONCALVES, Alexandre L. **Autonomic Quality of Experience Management of Software-Defined Networks,** Revue des Nouvelles Technologies de l'information, v. RNTI-L-8, p. 79-98, 2016a.

SILVA, Madalena Pereira da; NAZARIO, D. C.; DANTAS, M. A. R.; GONCALVES, A. L.; A.R. Pinto; MANERICHI, G.; VANELLI, B. **Implementação da IoT para o Monitoramento das Variáveis Meteorológicas num AAL.** In: XXXVI Congresso da Sociedade Brasileira da Computação, 2016, Porto Alegre. CSBC 2016. Porto Alegre: Porto Alegre, 2016. p. 2577-2586, 2016b.

SILVA, M. Pereira da; NAZARIO, Debora Cabral; DANTAS, M. A. R.; GONCALVES, Alexandre Leopoldo; PINTO, A. R. MANERICHI, Guilherme; VANELLI, Bruno. **An eHealth Context Management and Distribution Approach in AAL Environments.** In: IEEE 29th International Symposium on ComputerBased Medical Systems (CBMS), 2016, p. 169, Belfast and Dublin, 2016c.

SILVA, Madalena Pereira da; NAZARIO, Debora Cabral; DANTAS, M.A.R.; GONCALVES, Alexandre Leopoldo; PINTO, A.R.;

MANERICHI, Guilherme; VANELLI, Bruno. **Context Management and Distribution Architecture Using Software-Defined Networking**. In: 2016 IEEE 25th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2016d, p, 185, Paris.

SHAIKH, J.; FIEDLER, M.; COLLANGE, D. **Quality of experience from user and network perspectives**, *Annales des Telecommunications/Annals of Telecommunications*, vol. 65, no. 1-2, pp. 47–57, 2010.

SIRIN, E., PARSIAA, B., GRAUA, B.C., KALYANPURA, A. e KATZA, Y. (2007). **Pellet: A Practical OWL-DL Reasoner, in Web Semantics: Science, Services and Agents on the World Wide Web**, vol. 5(2), pages 51-52.

SKORIN-KAPOV, L.; MATIJASEVIC, M. **Analysis of QoS Requirements for e-Health Services and Mapping to Evolved Packet System QoS Classes**. *Int J Telem Appl*. 2010; 2010: 628086. Published online 2010 Oct 7.

SKORIN-KAPOV, L.; VARELA, M. **A multi-dimensional view of QoE: the ARCU model**, in *Proceedings of the 35th Jubilee International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO '12)*, Opatija, Croatia, May 2012.

SLACK, Nigel, et al. **Administração da produção**. São Paulo: Atlas, 1997. 726p.

SOCEANU, A.; VASYLENKO, M.; EGNER, A.; MUNTEAN, T. **Managing the Privacy and Security of eHealth Data**, 2015 20th International Conference on Control Systems and Computer Science, Bucharest, 2015, pp. 439-446.

SOMMERMAN, A. **Inter ou transdisciplinaridade?** São Paulo: Ed. Paulos. 2006.

SOURCEFORGE.NET. **The OWL API**. Disponível em: <<http://owlapi.sourceforge.net/>>. Acesso em: 08 ago 2013.

STANKIEWICZ, R.; JAJSZCZYK, A. 2011. **A survey of QoE assurance in converged networks**. *Comput. Netw.* 55, 7 (May 2011), 1459-1473.

STUDER, Rudi; BENJAMINS, Richard; FENSEL, Dieter. **Knowledge engineering: principles and methods**. Data and Knowledge Engineering 25, 1998.

STUDER, R., S. DECKER, et al. **Situation and Perspective of Knowledge Engineering**. Knowledge Engineering and Agent Technology. IOS Press, 2000.

SURE, Y.; STUDER, R. **A Methodology for Ontology-based Knowledge Management**. In: DAVIES, John; et al (eds). *Towards The Semantic Web: Ontology-Driven Knowledge Management*. John Wiley & Sons, p. 33-46, 2003.

TAKAHASHI, A.; HANDS, D.; BARRIAC, V. **Standardization activities in the ITU for a QoE assessment of IPTV**, IEEE Communications Magazine, vol. 46, no. 2, pp. 78–84, 2008.

TAYLOR, S.; TODD P. A. **Understanding Information Technology Usage - a Test of Competing Models**, *Information Systems Research*, 6, 2 (1995), 144-176.

TEIXEIRA G., F.L., SPOHR Nedel, A. and CARDOSO Alves, M.R (2012). **An analysis of the air relative humidity in internal and external environment in the city of São Paulo**, Brazil, *Revista Brasileira de Medicina*, v. 69, n. 7.

TODESCO, J. L., et al. **Laboratório de Engenharia do Conhecimento: desenvolvendo ontologias para a Gestão do Conhecimento**. II SEMINÁRIO DE PESQUISA EM ONTOLOGIA NO BRASIL. Anais...Rio de Janeiro: 2009.

TRIVIÑOS, A. N. S. **Introdução à pesquisa em ciências sociais: a pesquisa qualitativa em educação**. São Paulo: Atlas, 1987.

TSARKOV, D.; HORROCKS, I. **FACT++**. 2009. Disponível em: <<http://owl.man.ac.uk/factplusplus/>>. Acesso em: 26 jul. 2010.

TSOLKAS, D.; LIOTOU, E.; PASSAS, N.; MERAKOS, L. **A survey on parametric QoE estimation for popular services**, Journal of Network and Computer Applications, Volume 77, 1 January 2017, Pages 1-17, ISSN 1084-8045.

ULLAH, M., FIEDLER, M. and WAC, K. **On the ambiguity of Quality of Service and Quality of Experience requirements for eHealth services**, 2012 6th International Symposium on Medical Information and Communication Technology (ISMICT), La Jolla, CA, 2012, pp. 1-4.

VANELLI, B.; Pinto, A.R.; SILVA, Madalena P.; DANTAS, M.A.R.; CELESTI, A.; VILLARI, M. **IoT Data Storage in the Cloud: A Case Study in Human Biometeorology**. In: 2nd EAI International Conference on Cloud, Networking for IoT Systems, v. 1. p. 1-8, 2017. Brindisi. Brindisi, 2017a.

VANELLI, B.; SILVA, M.; MANERICH, G.; Pinto, A. R.; DANTAS, M.A.R.; FERRANDIM, M.; BOAVA, A. **Internet of Things Data Storage Infrastructure in the Cloud Using NoSQL Databases**. IEEE Latin America Transactions, v. 15, p. 737-743, 2017b.

VANELLI, BRUNNO ; Rodrigues, Mariana ; da Silva, Madalena Pereira; Pinto, Alex ; DANTAS, M. A. R. **A Proposed Architecture for Robots as a Service**. Communications in Computer and Information Science. 1ed.: Springer International Publishing, 2017c, v. , p. 117-130.

VEJLØ, Mads Holdgaard; ANDERSEN, Niels Frstrup. **Flow-based Dynamic Queue Selector: Guaranteeing service level agreements in a congested IP-network**. 2014. 88 f. Tese (Doutorado) - Curso de Networks And Distributed Systems, Electronic Systems, Aalborg, Aalborg, 2014.

VOLPATO, F.; SILVA, Madalena Pereira da; GONCALVES, Alexandre Leopoldo; CASTRO, M.; DANTAS, M. A. R. **Provisioning and Delivering Sepsis Data Supported by an Enhanced SDN Environment**. In: IEEE 30th International Symposium on Computer Based Medical Systems (CBMS), Thessaloniki, Greece, 2017, pp. 492-497. 2017a.



VOLPATO, F.; SILVA, M. P. da; GONCALVES, A. L.; DANTAS, M. A. R. **An Autonomic QoS Management Architecture for Software-Defined Networking Environments.** In: 22th IEEE Symposium on Computers and Communications (ISCC), Heraklion, 2017, pp. 418-423. 2017b.

VOLPATO, F.; SILVA, Madalena Pereira da; GONCALVES, Alexandre Leopoldo; DANTAS, M. A. R. **An Autonomic QoE Aware Management Architecture for Software-Defined Networking.** In: IEEE 26th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2017, Poznan. 2017c.

VOUYIOUKAS, D.; MAGLOGIANNIS, I. and KOMNAKOS, D. **Emergency m-Health Services through High-Speed 3G Systems: Simulation and Performance Evaluation,** Simulation, vol. 83, no. 4, pp. 329-345, 2007.

YAKHLEF, A. **Immobility of tacit knowledge and the displacement of the locus of innovation.** *European Journal of Innovation Management*, [S.l.], v. 8, n. 2, p. 227-239, 2005.

YASAR, A.-U.-H., et al. **When efficiency matters: Towards quality of context-aware peers for adaptive communication in VANETs.** In: IEEE INTELLIGENT VEHICLES SYMPOSIUM (IV). Proceedings... 2011. p. 1006-1012.

WAC, K.; ICKIN, S.; HONG, J. H.; JANOWSKI, L.; FIEDLER, M.; DEY, A. K. **Studying the experience of mobile applications used in different contexts of daily life,** in Proceedings of the 1st ACM SIGCOMM Workshop on Measurements up the Stack (W-MUST'11), Toronto, Ontario, Canada, August 2011.

WAMSER, F.; DESCHNER, S.; ZINNER, T.; TRAN-GIA, P. 2013. **Investigation of different approaches for QoE oriented scheduling in OFDMA networks.** In: (EDS.) Mobile Networks and Management. Springer, vol. 125, pp. 172–187.

WAZLAWICK, Raul Sidnei. **Metodologia de Pesquisa para Ciência da Computação.** Rio de Janeiro : Elsevier, 2009.

WEBER, E. H. **Annotationes Anatomicae et Physiologicae: Programmata Collecta: Fasciculi Tres. de Pulsu, Resorptione, Audituet Tactu**, Koehler, 1834.

WECHSUNG I. et al. **Measuring the Quality of Service and Quality of Experience of multimodal human machine interaction**, J Multimodal User Interfaces, 2012.

WU W. et al. **Quality of Experience in Distributed Interactive Multimedia Environments: Toward a Theoretical Framework**, in Proceedings of the 17th ACM international conference on Multimedia, 2009.

W3C (RDF), **Resource Description Framework (RDF) Model and Syntax Specification**, Ora Lassila, et al., (eds.). W3C Recommendation 22 February 1999. Disponível em: <<https://www.w3.org/TR/1999/REC-rdf-syntax-19990222/#basic>>. Acesso em: 10 maio 2013.

W3C (SPARQL), **SPARQL Query Language for RDF**. W3C Recommendation 15 January 2008. Disponível em: <<https://www.w3.org/TR/rdf-sparql-query/>>. Acesso em: 20 setembro 2013.

3GPP TS 22.105. **Services and service capabilities**. V9.0.0, Release 9, 2008-12.

3GPP TS 26.944. **End-to-end multimedia services performance metrics**. V9.0.0, Release 9, 2009-12.

## APÊNDICES

<b>APÊNDICE A – PUBLICAÇÕES .....</b>	<b>297</b>
---------------------------------------	------------



## APÊNDICE A – PUBLICAÇÕES

Este apêndice lista as publicações realizadas ao longo da pesquisa deste trabalho entre os anos de 2013 a 2017. Optou-se em agrupá-las pelos tipos de publicações: Capítulos de livros, Periódicos Internacionais e Nacionais, Eventos Internacionais e Nacionais. Para cada publicação é feito um breve resumo relacionando o material publicado com a tese.

### MATERIAL PUBLICADO

#### CAPÍTULOS DE LIVROS PUBLICADOS

**SILVA, Madalena Pereira da; DANTAS, M. A. R.; GONCALVES, A. L. Uma Abordagem Interdisciplinar para a Proposição de um Modelo de Qualidade de Experiência em Serviços Multimídia.** In: Fernando Ostuni Gauthier; Selvino Assmann; Silvia Maria Puentes Bentancourt; Micheline Guerreiro Krause; Maricel Karina López Torres; Fernanda Martinhago; Jair Zandoná; Silvia Regina Pochmann de Quevedo. (Org.). *Interdisciplinaridade Teoria e Prática*. 1ª ed. Florianópolis: UFSC/EGC, 2014, v. 1, p. 401-430.

#### ➤ **Relação do trabalho com a tese**

No capítulo são apresentados os resultados da investigação dos fatores e parâmetros que tem sido usados para mensurar a QoE numa perspectiva interdisciplinar. Como resultado é apresentado um modelo da relação causa-efeito entre QoS e QoE. Um conjunto de dimensões, a partir do modelo proposto, é analisado considerando um cenário multimídia onde um serviço de vídeo sob demanda é fornecido com base na experiência do usuário, adotando a flexibilidade do paradigma SDN. O modelo interdisciplinar é amparado por teorias da Psicologia, Ciência da Cognição, Semiótica, Engenharia do Conhecimento, Computação Sensível ao Contexto e Tecnologia da Informação e da Comunicação.

#### PERIÓDICOS INTERNACIONAIS:

VANELLI, B.; **SILVA, M.**; MANERICH, G.; Pinto, A. R.; DANTAS, M.A.R.; FERRANDIM, M.; BOAVA, A. **Internet of Things Data Storage Infrastructure in the Cloud Using NoSQL Databases.** *IEEE Latin America Transactions*, v. 15, p. 737-743, 2017. **Qualis B1.**

➤ **Relação do trabalho com a tese**

Um dos experimentos da tese consistiu na monitoração de sinais vitais e ambientais num AAL com uso da IoT, possibilitando aumentar a potencialidade de dispositivos de monitoramento invasivos. No entanto, a implementação e integração de dispositivos IoT, armazenamento de dados e desenvolvimento de aplicativos ainda são considerados desafiantes, sobretudo se tratando do grande volume de dados a ser persistido. Partindo do pressuposto que os AAL já fazem parte da realidade de muitas pessoas (idosos e pacientes que necessitam de monitoramento remoto a partir de suas residências) e que num futuro próximo múltiplos AALs irão formar uma rede de assistência a saúde das pessoas, uma intenção de trabalho futuro da tese consiste em prover uma infraestrutura de agregação e armazenamento de dados em bases de dados noSQL na nuvem. Portanto, nesse artigo foi dado início nas pesquisas nessa direção, propondo uma infraestrutura para agregar e armazenar dados de diferentes fontes de dispositivos da IoT de um AAL para a nuvem.

**SILVA, Madalena Pereira da; DANTAS, M. A. R.; GONCALVES, Alexandre L. *Autonomic Quality of Experience Management of Software-Defined Networks*, *Revue des Nouvelles Technologies de l'information*, v. RNTI-L-8, p. 79-98, 2016.**

➤ **Relação do trabalho com a tese**

No artigo é proposto um motor semântico com características autonômicas para gerenciamento da QoE em SDN. O ciclo de gerenciamento do motor (MAPE-K), juntamente com a aplicação de controle SDN, contribuíram para a gestão da QoE com fina granularidade, uma vez que as configurações manuais são minimizadas e as violações da QoE são detectadas para que as ações de restauração sejam realizadas afim de prover o serviço ciente da experiência do usuário.

**PERIÓDICOS NACIONAIS:**

---

**MARTINS, D.; SILVA, Madalena Pereira da; DANTAS, M. A. R. *Um estudo de qualidade de serviço em ambientes SDN*. *Revista Junior de Iniciação Científica em Ciências Exatas e Engenharia*, v. 1, p. 23-28, 2016.**

➤ **Relação do trabalho com a tese**

No artigo foi realizado um estudo sobre o suporte à QoS em SDN, com uso do protocolo *OpenFlow*. O propósito maior consistiu em analisar o suporte à QoS no controlador *Floodlight* e as suas limitações para a aplicação de novos métodos de monitoração de parâmetros de QoS. Esse estudo foi primordial para explorar a QoS no *Floodlight*, o mesmo controlador usado nos experimentos da tese.

## **EVENTOS INTERNACIONAIS:**

---

VOLPATO, F.; SILVA, M. P. da; GONCALVES, A. L.; DANTAS, M. A. R. **An Autonomic QoS Management Architecture for Software-Defined Networking Environments.** In: *2017 IEEE Symposium on Computers and Communications (ISCC)*, Heraklion, 2017, pp. 418-423. **Qualis A2.**

### ➤ **Relação do trabalho com a tese**

No artigo é proposta uma aplicação de rede (*Autonomic QoS Broker*) e um módulo de controle que implementa o protocolo de gerenciamento de banco de dados *OpenVSwitch (OVSDB)* para o gerenciamento da QoS. Os componentes foram implementados e validados para fornecer gerenciamento de QoS com priorização de fila em um ambiente SDN. Através dos experimentos verificou-se que os recursos foram fornecidos com diferenciação, de acordo com o contexto disponível na KB. O *Broker* conseguiu gerenciar os fluxos, aumentando a largura de banda quando possível e alterando os caminhos dos mesmos quando os *links* estavam congestionados ou com latência (atraso) acima dos limites aceitáveis.

VOLPATO, F.; SILVA, M. P. da; GONCALVES, A. L.; DANTAS, M. A. R. **An Autonomic QoE Aware Management Architecture for Software-Defined Networking.** In: *2017 IEEE 26th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, Poznan, 2017, pp. 220-225. **Qualis B1.**

### ➤ **Relação do trabalho com a tese**

O artigo apresenta uma variação da arquitetura autônômica (DA SILVA et al. 2015), com duas principais contribuições: (1) o aplicativo QoE (App QoE), que é uma aplicação hipotética responsável por adquirir o *feedback* dos usuários; e (2) um módulo de controle (QoE-ACM), projetado com a funcionalidade do *loop* de controle MAPE-K

(IBM, 2005). Os componentes propostos foram projetados e validados em um ambiente SND. Os resultados demonstraram que a solução é capaz de coletar e fornecer informações sobre a experiência do usuário; mapear os parâmetros de QoS em QoE para prever o MOS e atuar de forma autônoma para ajustar os recursos na infraestrutura de rede subjacente, evitando degradação da QoE.

VOLPATO, F.; **SILVA, Madalena Pereira da**; GONCALVES, Alexandre Leopoldo; CASTRO, M.; DANTAS, M. A. R. **Provisioning and Delivering Sepsis Data Supported by an Enhanced SDN Environment**. In: *2017 IEEE 30th International Symposium on Computer-Based Medical Systems (CBMS)*, Thessaloniki, Greece, 2017, pp. 492-497. **Qualis B1**.

➤ **Relação do trabalho com a tese**

As aplicações médicas, juntamente com as TICs, contribuíram com muitas soluções para apoiar o tratamento da Sepsis, no entanto, existem poucas soluções para o transporte de dados de Sepsis com QoS. No artigo é proposta uma arquitetura SDN autogerenciável para a provisão e entrega de dados de Sepsis. Para avaliar a proposta, os experimentos foram realizados em laboratório e os resultados mostraram-se promissores, uma vez que a solução com SDN é capaz de aprimorar o desempenho, garantir QoS e priorizar os fluxos de Sepsis, quando comparada com a arquitetura TCP/IP.

VANELLI, B.; Pinto, A.R.; **SILVA, Madalena P.**; DANTAS, M.A.R.; CELESTI, A.; VILLARI, M. **IoT Data Storage in the Cloud: A Case Study in Human Biometeorology**. In: *2nd EAI International Conference on Cloud, Networking for IoT Systems*, 2017, Brindisi. Brindisi, 2017. v. 1. p. 1-8.

➤ **Relação do trabalho com a tese**

Uma das intenções de implementações futuras da tese consiste em armazenar os dados provenientes dos sensores em ambiente em nuvem, portanto, no artigo é apresentada uma infraestrutura para agregação e armazenamento de dados de diferentes fontes de dispositivos da IoT para a nuvem. Para avaliar a infraestrutura em relação à qualidade do armazenamento dos dados, a mesma foi desenvolvida e implantada num ambiente de computação em nuvem, sendo verificada num cenário de AAL, aplicada na Biometeorologia Humana.



**SILVA, Madalena Pereira da; NAZARIO, Debora Cabral; DANTAS, M. A. R.; GONCALVES, Alexandre Leopoldo; PINTO, A. R. MANERICHI, Guilhermer; VANELLI, Bruno. *An eHealth Context Management and Distribution Approach in AAL Environments*. In: *2016 IEEE 29th International Symposium on Computer Based Medical Systems (CBMS)*, 2016, p. 169, Belfast and Dublin. **Qualis B1**.**

➤ **Relação do trabalho com a tese**

O artigo propõe uma arquitetura para gerenciar e distribuir informações de contexto de um ambiente AAL para a nuvem. A arquitetura, com características semânticas, inclui o processamento dos dados detectados usando parâmetros QoC e transporte de dados usando SDN. Através dos experimentos verificou-se que a abordagem semântica ajuda a melhorar a precisão da qualidade da informação de contexto e, por conseguinte, o comportamento das aplicações sensíveis ao contexto que irão consumir os dados.

**SILVA, Madalena Pereira da; NAZARIO, Debora Cabral; DANTAS, M.A.R.; GONCALVES, Alexandre Leopoldo; PINTO, A.R.; MANERICHI, Guilherme; VANELLI, Bruno. *Context Management and Distribution Architecture Using Software-Defined Networking*. In: *2016 IEEE 25th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, 2016, p. 185, Paris. **Qualis B1**.**

➤ **Relação do trabalho com a tese**

No artigo é proposta uma arquitetura de gerenciamento e distribuição de contexto de *eHealth* com recursos semânticos, projetada para a Internet do Futuro. Uma base formal foi utilizada para gerenciar QoC e QoS. A abordagem proposta foi avaliada em uma configuração experimental utilizando sensores biomédicos e ambientais. Os experimentos mostraram que o processamento semântico do contexto é capaz de detectar anomalias ou inconsistências, gerando alertas consistentes. Foi evidenciado que a SDN permite que o contexto seja transportado e entregue de acordo com os requisitos de QoS. Portanto, a abordagem provê o conhecimento com fina granularidade para ajudar os cuidadores da saúde a tomarem decisões.

**ASSUITI, M.; VOLPATO, F.; SILVA, Madalena Pereira da; DANTAS, M. A. R. *A Software-Defined Network Configuration Providing***

**Differentiated QoS to an eHealth Environment.** *In: Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications*, PDPTA, 2016. WorldComp 2016. Las Vegas: CSREA Press, 2016. p. 237-243. **Qualis B2.**

➤ **Relação do trabalho com a tese**

O artigo apresenta uma abordagem para o gerenciamento da QoS, com o objetivo de prestar serviços cientes da urgência no diagnóstico de Sépsis. O modelo propõe um ambiente não concorrente para a prestação de serviços, onde a aplicação de Sépsis garante a QoS usando os recursos da SDN. Através dos experimentos foi possível evidenciar que a abordagem proposta mostrou-se viável e funcional, evitando dados concorrentes na prestação de serviços com a aplicação de Sépsis.

SILVA, Madalena Pereira da; DANTAS, M. A. R.; GONCALVES, A. L. **An Interdisciplinary Approach to Proposition of a Multidimensional Model of Quality of Experience.** *In: 12th International Conference on Management of Technology and Information Systems*, 2015, São Paulo. 12th CONTECSI. São Paulo: TECSI EAC FEA USP, 2015. p. 4869-4876. **Qualis B4.**

➤ **Relação do trabalho com a tese**

Como o foco da tese consiste no estudo da QoE como constructo multidimensional, pautado numa abordagem interdisciplinar, no artigo foi realizada uma revisão do tema Qualidade de Experiência para investigar se as pesquisas existentes são norteadas para as Ciências Humanas ou apenas para a Tecnologia da Informação e Comunicação. Com base nas pesquisas é proposto um modelo multidimensional da QoE e uma taxonomia das dimensões que causam impacto na qualidade percebida pelo usuário. A caracterização da QoE como um constructo multidimensional, fundamentada numa abordagem interdisciplinar, nos remete a uma reflexão do quanto as pesquisas precisam evoluir para oferecer serviços capazes de atender as necessidades hedônicas e pragmáticas do usuário.

NAZARIO, D. C.; SILVA, Madalena Pereira da; DANTAS, M. A. R.; TODESCO, J. L. **Quality of Context Knowledge Model Based on Ontology.** *In: 12th International Conference on Management of Technology and Information Systems*, 2015, São Paulo. 12th CONTECSI. São Paulo: TECSI EAC FEA USP, 2015. p. 2890-2913. **Qualis B4.**

➤ **Relação do trabalho com a tese**

O artigo aborda a representação de conhecimento, propondo um Modelo de Conhecimento de Qualidade de Contexto, baseado em Ontologia. O modelo de conhecimento estrutura, organiza e integra o conhecimento encontrado na literatura, através de ontologia, permitindo o compartilhamento e reutilização. O modelo serviu de subsídios para o gerenciamento e distribuição do contexto com qualidade nas novas pesquisas publicadas em Silva et al. (2016a) e Silva et al. (2016b).

SILVA, Madalena P. da; DANTAS, M. A. R.; GONCALVES, A. L. **An architecture with autonomic and semantic features for QoE management in SDN**. In: *MODA 2015: Management of Ontology-Driven Architectures*, 2015, Hammamet. MODA 2015, 2015.

➤ **Relação do trabalho com a tese**

No artigo é apresentada a primeira versão da arquitetura proposta nesta tese. O objetivo da pesquisa consistiu no gerenciamento da QoE para a prestação de serviços ciente da experiência do usuário, usando SDN. A arquitetura foi enriquecida com um motor semântico autônomo, capaz de detectar e executar ações de adaptação de políticas em caso de degradações da QoE.

DA SILVA, Madalena P.; DANTAS, Mario A. R.; GONÇALVES, Alexandre L.; PINTO, Alex R. **A Managing QoE Approach for Provisioning User Experience Aware Services Using SDN**. In: *the 11th ACM Symposium, 2015, Cancun. Proceedings of the 11th ACM Symposium on QoS and Security for Wireless and Mobile Networks - Q2SWinet '15*. v. 1. p. 51-58. **Qualis B2**.

➤ **Relação do trabalho com a tese**

No artigo a arquitetura original da tese é revisitada e novos componentes são propostos para aprimorar o gerenciamento da qualidade de experiência. São propostas a modelagem e a arquitetura de QoE, sendo a mesma composta de um mecanismo semântico capaz de aprender a experiência do usuário no uso de um serviço, detectar violações de métricas de QoS e fornecer informações para que o controlador execute ações nos elementos da SDN. Os resultados experimentais demonstraram que a proposta é viável e funcional e que o tempo entre a detecção de degradação na QoE e a adaptação de políticas nos recursos de rede não influencia na qualidade percebida pelo usuário.

SILVA, Madalena P. da; GONCALVES, A. L.; DANTAS, M. A. R.; Vanelli, B.; MANERICHI, G.; SANTOS, S. A. R. D.; FERRANDIM, M.; A.R. Pinto. **Implementation of IoT for Monitoring Ambient Air in Ubiquitous AAL Environments.** In: *V Brazilian Symposium on Computing Systems Engineering*, 2015, Foz do Iguacu. V Brazilian Symposium on Computing Systems Engineering. v. 1. **Qualis B2.**

➤ **Relação do trabalho com a tese**

Nos primeiros experimentos da tese foi usada a rede de sensores *ZigBee*, com IoT para capturar dados dos sensores de ambiente de AAL. Portanto, nesse artigo foi investigada a viabilidade do uso de WSN (*Wireless Sensor Networks*) em saúde e proposto a implementação do IoT (*Internet of Things*) com o *ZigBee* para monitoramento das condições ambientais em um ambiente AAL ubíquo. Os dados obtidos a partir dos sensores (temperatura e umidade) foram combinados de modo a obter a temperatura do ponto de orvalho, usada como indicador de desconforto térmico dos usuários com pré-disposição a problemas respiratórios. As leituras das variáveis monitoradas foram transmitidas através da Internet e armazenadas no sistema de supervisão ScadaBR. Os dados foram monitorados pela HMI (*Human-Machine Interface*) do ScadaBR e consumidos por uma aplicação externa para que os cuidadores da saúde assistissem os pacientes com antecedentes de crises asmáticas.

**EVENTOS NACIONAIS:**

---

SILVA, Madalena Pereira da; NAZARIO, D. C.; DANTAS, M. A. R.; GONCALVES, A. L.; A.R. Pinto; MANERICHI, G.; VANELLI, B. **Implementação da IoT para o Monitoramento das Variáveis Meteorológicas num AAL.** In: *XXXVI Congresso da Sociedade Brasileira da Computação*, 2016, Porto Alegre. CSBC 2016. Porto Alegre: Porto Alegre, 2016. p. 2577-2586. **Qualis B4.**

➤ **Relação do trabalho com a tese**

A maioria das pesquisas sobre a Biometeorologia Humana usa dados de provedores de dados da saúde e do clima para correlacionar e inferir os impactos das variáveis climáticas sobre as morbidades. Embora possam gerar indicadores para a gestão da saúde, as pesquisas precisam ser refinadas no sentido de considerar as leituras das variáveis meteorológicas no ambiente interno e, disponibilizar esses dados a

serem avaliados juntamente com os dados clínicos dos pacientes. Visando contribuir com a Biometeorologia Humana, no artigo é proposta a implementação da IoT com o uso da *ZigBee* para monitorar as condições ambientais de um AAL ubíquo, e armazenar os dados na nuvem para serem consumidos por aplicações de cuidadores da saúde.

**SILVA, Madalena P. da; DANTAS, M.A.R. Modelo de Qualidade de Experiência à Gestão do Conhecimento em Empresas Fornecedoras de Serviços Convergentes. In: Congresso Internacional Conhecimento e Inovação, 2013, Porto Alegre. CIKI, 2013, 2013.**

➤ **Relação do trabalho com a tese**

O artigo acima descrito foi a primeira publicação da tese. Assim que o tema foi definido iniciaram-se as pesquisas sobre os fatores de influência da QoE. Com base nos trabalhos existentes foi realizado um mapeamento das dimensões de impacto na QoE, incluindo os fatores humanos, conteúdo, contexto e tecnologia.