



**UNIFACS**  
UNIVERSIDADE SALVADOR  
LAUREATE INTERNATIONAL UNIVERSITIES\*

**UNIFACS UNIVERSIDADE SALVADOR  
PROGRAMA DE PÓS-GRADUAÇÃO  
MESTRADO ACADÊMICO EM SISTEMAS E COMPUTAÇÃO**

**CARLOS FREDERICO JANSEN MUAKAD**

**GERÊNCIA DE ENCAMINHAMENTO DINÂMICO E ADAPTATIVO  
BASEADO EM CONTEXTO**

Salvador  
2015

**CARLOS FREDERICO JANSEN MUAKAD**

**GERÊNCIA DE ENCAMINHAMENTO DINÂMICO E ADAPTATIVO  
BASEADO EM CONTEXTO**

Dissertação apresentada ao Mestrado em Sistemas e Computação da UNIFACS Universidade Salvador, Laureate International Universities como requisito parcial para a obtenção do título de Mestre.

Orientador: Prof. Dr. Paulo Sampaio.

Salvador  
2015

## FICHA CATALOGRÁFICA

(Elaborada pelo Sistema de Bibliotecas da UNIFACS Universidade  
Salvador, Laureate International Universities)

Muakad, Carlos Frederico Jansen

Gerência de encaminhamento dinâmico e adaptativo baseado  
em contexto/ Carlos Frederico Jansen Muakad. - 2015.

131 f. : il.

Dissertação Programa de Pós-Graduação em Sistemas e  
Computação de UNIFACS Universidade Salvador, Laureate  
International Universities como requisito parcial à obtenção do título  
de Mestre.

Orientador: Prof. Dr. Paulo Nazareno Maia Sampaio.

1. Encaminhamento adaptativo. 2. Redes sensíveis ao contexto.  
I. Sampaio, Paulo Nazareno Maia, orient. II. Título.

CDD:004. 6

## TERMO DE APROVAÇÃO

CARLOS FREDERICO JANSEN MUAKAD

### GERÊNCIA DE ENCAMINHAMENTO DINÂMICO E ADAPTATIVO BASEADO EM CONTEXTO

Dissertação aprovada como requisito parcial para obtenção do grau de Mestre, na UNIFACS Universidade Salvador, Laureate International Universities, pela seguinte banca examinadora:

Paulo Nazareno Maia Sampaio – Orientador \_\_\_\_\_  
Doutor em Informatique et Télécommunications Université Paul Sabatier - Toulouse III/LAAS-  
CNRS  
UNIFACS Universidade Salvador, Laureate International Universities

Jorge Alberto Prado de Campos \_\_\_\_\_  
Doutor em Spatial Information Science and Engineering.  
University of Maine at Orono, U.M.O., Estados Unidos  
UNIFACS Universidade Salvador, Laureate International Universities

Romildo Martins Bezerra \_\_\_\_\_  
Doutorado em Ciência da Computação - Ufba - Unifacs  
Instituto Federal da Bahia – IFBA

Salvador, 23 de outubro de 2015

## AGRADECIMENTOS

Após esse árduo período de concentração nos estudos e nas pesquisas, gostaria de registrar meus sinceros agradecimentos:

Primeiramente, agradecer a Deus, por me fazer acreditar que tudo é possível e me dar fé.

Ao meu orientador, professor Dr. Paulo Sampaio, pela amizade, pelo fácil convívio, pelas duras, com a sua educação de sempre, pela paciência e por acreditar em toda a equipe que ele lidera. Apesar da sua agenda sempre cheia de compromissos com os grupos de estudo, congressos internacionais, entre outras atividades acadêmicas, que ocupam muito seu tempo, ele sempre esteve focado em nos apoiar e orientar com a sua experiência e sabedoria. Serei sempre muito agradecido por tudo isso.

Ao nosso grupo de “Redes sensíveis ao contexto”, em especial a André, meu grande parceiro nas grandes discussões e definições. Agradeço, também, a Joatham e Sérgio, pela parceria, interação constante e compromisso que tiveram, ressaltando a experiência profissional de Sérgio e a experiência prática de Joatham. Nesse grupo cada um teve sua importância em todo o estudo.

Tenho a agradecer, também, aos meus colegas da UNIK, Francisco, Muller e Mateus, e ao meu coordenador da Estacio-Fib, Antonio Cordeiro. Aos meus antigos colegas da Embasa, Rubens e Rui Botelho, agradeço pelo grande apoio também.

E a minha família, em especial, a minha mãe, meu pai, minha esposa e meus filhos, agradeço pela força, pela compreensão nas minhas ausências e por acreditar em meu potencial. Vocês fizeram parte dessa minha conquista. Muito obrigado por tudo.

## RESUMO

A evolução tecnológica tem aumentado muito o número de usuários e dispositivos conectados, além de novas aplicações com exigências cada vez maiores de conteúdo com sensibilidade temporal. No entanto, as atuais soluções implementadas não têm sido eficientes para garantir a qualidade da entrega de pacotes das comunicações dentro das redes de computadores pelo mundo. Além disso, para as demandas emergentes, existe a necessidade da preocupação com aspectos como congestionamento, admissão e gerenciamento de recursos. Atualmente, os recursos que têm sido usados para manter toda essa demanda são as métricas de qualidade de serviço. Contudo, as novas características das novas aplicações como mobilidade, georeferenciamento, entre outros tipos de informação, tem exigido mais da infraestrutura e esta não tem sido eficiente para toda essa carga de dados. Essa situação tem impactado negativamente na experiência do usuário. A importância da avaliação do contexto, que engloba não somente a qualidade de serviço, mas também qualidade experiência e qualidade de dispositivo, têm se evidenciado como uma solução interessante para minimizar os problemas relacionados às comunicações dos usuários. O conceito de redes orientadas ao contexto é discutido nessa dissertação através da simulação de uma solução para encaminhamento adaptativo baseado em contexto, que contempla aspectos de qualidade de serviço, como atraso, largura de banda e variação do atraso, aspectos de qualidade de experiência, como o MOS (*Mean Opinion Score*) e aspectos de qualidade de dispositivo como capacidade de processamento e de memória de um computador. Nesse trabalho é possível demonstrar, através de simulação, que a implementação de redes sensíveis ao contexto permite otimizar o uso de recursos da rede e melhorar as condições de atendimento da demanda dos usuários, causando um impacto positivo na experiência.

**Palavras-chave:** Redes Sensíveis ao Contexto. Qualidade de Contexto. Qualidade de Serviço. Qualidade de Experiência. Qualidade de Dispositivos. Encaminhamento Adaptativo.

## ABSTRACT

The technological evolution has improved the number of interconnected users and devices, besides the number of applications with a growing demand for time-sensitive content. Nevertheless, the current implemented solutions have not been effective in order to ensure the packet delivery quality within communication networks throughout the world. Moreover, in order to support the emerging traffic demand related to applications such as video-on-demand and audio-conference, some issues should be revisited such as congestion, admission control and resources management. Currently, most of the existing solutions available for handling these issues are based on quality of service metrics. However, some features of these emerging applications such as mobility, georeference, among others require a larger amount of resources from the infrastructure which has become inefficient with this higher data load. This situation has caused a negative impact on users' experience. Therefore, context evaluation, which is based not only on quality of service, but also on experience quality and device's quality, has become an interesting solution in order to minimize problems related to user's communication. The concept of context-based networks is discussed in this dissertation through the simulation of a solution for the context-based adaptive routing, considering quality of service metrics such as delay, bandwidth and jitter; quality of experience metric such as MOS (Mean Opinion Score) and; quality of device metrics such as processing power and memory availability. In this work it is possible to demonstrate that, through simulation, that the implementation of context-sensitive networks allows the optimization of network resources utilization and improve the delivery of user demands causing a positive impact on users' experience.

**Keywords:** Context-aware Network. Quality of Context. Quality of Service. Quality of Experience. Quality of Device. Adaptive Routing.

## LISTA DE TABELAS

Tabela 1 - Lista de estações e seus roteadores.....	67
Tabela 2 - Lista de Encaminhadores .....	68
Tabela 3 - Dispositivos e Roteadores configurados no NS-2.....	69
Tabela 4 - Tabela de Conexões entre Encaminhadores.....	69
Tabela 5 - Definição dos canais no NS-2 .....	71
Tabela 6 - Aplicações usadas nos três cenários.....	71
Tabela 7 - Definição das Aplicações no NS-2.....	72
Tabela 8 - Fluxo e o intervalo de simulação em detalhe .....	72
Tabela 9 - Resultados do Cenário 1 –Antes da 1ª Execução.....	74
Tabela 10 - Resultados do Cenário 1 – Após a 2ª Execução.....	79
Tabela 11 - Cenário 1 - Evolução do Contexto .....	80
Tabela 12 - Cenário 1 - Contexto versus XML versus Banco versus TCL.....	81
Tabela 13 - Fluxo e o intervalo de simulação em detalhe .....	83
Tabela 14 - Resultados do Cenário 2–Antes da 1ª Execução .....	84
Tabela 15 - Resultados do Cenário 2–Antes da 3ª Execução .....	84
Tabela 16 - Resultados do Cenário 2 –Após a 3ª Execução.....	87
Tabela 17 - Cenário 2 - Evolução do Contexto .....	88
Tabela 18 - Cenário 2 - Contexto versus XML versus Banco versus TCL.....	89
Tabela 19 - Fluxo e o intervalo de simulação em detalhe .....	91
Tabela 20 - Resultados do Cenário 2–Antes da 3ª Execução .....	93
Tabela 21 - Resultados do Cenário 3 – Após a 3ª Execução.....	95
Tabela 22 - Cenário 3 - Evolução do Contexto .....	97
Tabela 23 - Cenário 3 - Contexto versus XML versus Banco versus TCL.....	98



## LISTA DE FIGURAS

Figura 1- Estrutura do CAARF (Ênfase na Abordagem de Encaminhamento) .....	17
Figura 2 - Características fundamentais de contexto (ZIMMERMANN et al., 2007). .....	27
Figura 3 - Caracterização do Modelo de Contexto.....	37
Figura 4 - Desenho da Arquitetura do CAARF.....	38
Figura 5 - Context-based Forwarding Management.....	41
Figura 6 - Flow Adaptation .....	42
Figura 7 - Grupos de Regras de Encaminhamento .....	44
Figura 8 - Fluxo Principal na análise das regras de encaminhamento.....	46
Figura 9 - Funcionamento da Abordagem de Encaminhamento .....	48
Figura 10 - Ilustração do Processo de Simulação.....	54
Figura 11 - Ilustração de cada rodada da Simulação .....	55
Figura 12 - Algumas tecnologias aderentes ao CAARF .....	56
Figura 13 - Modelo de dados do Context Controller.....	61
Figura 14 - Topologia dos cenários de simulação .....	66
Figura 15 - Informações das estações no Banco.....	68
Figura 16 - Informações das estações no Banco (Parâmetros).....	68
Figura 17 - Informações dos Encaminhadores no Banco .....	68
Figura 18 - Informações dos Encaminhadores no Banco (Parâmetros) .....	69
Figura 19 - Informações de Caminhos e Links no banco .....	70
Figura 20 - Informações das Aplicações no banco.....	72
Figura 21 - Evolução dos fluxos na simulação (em minutos) .....	73
Figura 22 - Execuções 1 e 3 da Simulação do Cenário 1 (Novas Relações).....	73
Figura 23 - Fluxos do Cenário 1 (Novas Relações) .....	74
Figura 24 - Visão ContextModelView .....	75
Figura 25 - Seleção do contexto no ContextDatabase na 1ª Execução .....	76
Figura 26 - Seleção dos nós do melhor caminho da 1ª Execução.....	76
Figura 27 - Seleção do contexto no ContextDatabase na 2ª Execução .....	77
Figura 28 - Seleção dos nós do melhor caminho da 2ª Execução.....	77
Figura 29 - Execuções 2 e 4 da Simulação do Cenário 1 (Novos Contextos).....	78
Figura 30 - Fluxos do Cenário 1 (Mudança de Contexto).....	78
Figura 31 - Evolução dos fluxos na simulação (em minutos) .....	83
Figura 32 - Fluxos do Cenário 2 (Novas Relações) .....	83

Figura 33 - Seleção do contexto no ContextDatabase na 3ª Execução .....	85
Figura 34 - Execução 3 da Simulação do Cenário 2 (Novos Contextos) .....	85
Figura 35 - Fluxos do Cenário 2 (Mudança de Contexto).....	86
Figura 36 - Seleção dos nós do melhor caminho da 3ª Execução para Tráfego CBR3 .....	86
Figura 37 - Seleção dos nós do melhor caminho da 3ª Execução para Tráfego CBR4 .....	87
Figura 38 - Evolução dos fluxos na simulação (em minutos) .....	92
Figura 39 - Fluxos do Cenário 3 (Novas Relações) .....	92
Figura 40 - Seleção dos nós do melhor caminho da 3ª Execução para Tráfego CBR3 .....	94
Figura 41 - Execução 3 da Simulação do Cenário 3 (Novos Contextos) .....	94
Figura 42 - Fluxos do Cenário 3 (Mudança de Contexto).....	95
Figura 43 - Gráficos Comparativos (Atraso e Vazão Total) .....	96
Figura 44 - Gráficos Comparativos (Bits Transmitidos totais) .....	96
Figura 45 - Tempo de Resposta para escolha do melhor caminho .....	100
Figura 46 - Seleção das Relações com mudança de contexto .....	100

## LISTA DE ABREVIATURAS

QoD	Quality of Device
CAARF	Context-Aware Adaptive Routing Framework
QoS	Quality of Service
QoE	Quality of Experience
QoC	Quality of Context
GMPLS	Generalized Multi-Protocol Label Switching
GPS	Global Positioning System
MPLS	Multi-Protocol Label Switching
XML	Extensible Markup Language
SQL	Structured Query Language
MOS	Mean Opinion Score
MPLS	Multi Protocol Label Switching
TCL	Tool Control Language
UDP	User Datagram Protocol
TCP	Transport Control Protocol
SLA	Service Level Agreement
IGP	Internet Gateway Protocol
VoIP	Voice over IP
VoD	Video on demand
IP	Internet Protocol
CANC	Context-aware Network Coding
INFORM	Interest Forwarding Mechanism
AMPLE	Adaptive Multi-Topology Traffic Engineering
OLWO	Offline Link Weight Optimization
REPLEX	Replication Exploration
MAC	Medium Access Control
MATE	MPLS Adaptive Traffic Engineering
TEXCP	Traffic Engineering Explicit Control Protocol
FTP	File Transfer Protocol
ATC	Adaptive Traffic Control

## SUMÁRIO

1 INTRODUÇÃO.....	14
1.1 PROBLEMÁTICA.....	15
1.2 MOTIVAÇÕES.....	16
1.3 OBJETIVOS.....	17
1.4 METODOLOGIA.....	18
1.5 PRINCIPAIS CONTRIBUIÇÕES.....	18
1.6 ORGANIZAÇÃO DA DISSERTAÇÃO.....	20
2 REVISÃO CONCEITUAL E DE LITERATURA.....	21
2.1 CONCEITOS E DEFINIÇÕES.....	21
2.1.1 Qualidade de Serviço.....	21
2.1.2 Qualidade de Experiência.....	22
2.1.3 Qualidade de Dispositivo.....	24
2.1.4 Encaminhamento Adaptativo.....	25
2.1.5 Engenharia de Tráfego.....	25
2.1.6 Contexto.....	26
2.2 ESTADO DA ARTE.....	28
2.2.1 Qualidade de Serviço.....	28
2.2.2 Qualidade de Dispositivo.....	30
2.3 ENCAMINHAMENTO ADAPTATIVO.....	30
2.4 CONCLUSÃO.....	34
3 ARCABOUÇO BASEADO EM CONTEXTO.....	35
3.2 CONTEXT-AWARE ADAPTIVE ROUTING FRAMEWORK (CAARF).....	37
3.3 GESTÃO DE ENCAMINHAMENTO DO CAARF: MODELAGEM E FORMALIZAÇÃO.....	39
3.3.1.1 Módulos de Encaminhamento.....	41
3.3.1.2 Regras de encaminhamento.....	42
3.3.1.1 Eventos.....	46
3.4 CAARF: Abordagem operacional.....	47
3.5 CONCLUSÃO.....	49
4 SIMULAÇÃO DA GESTÃO DE ENCAMINHAMENTO.....	51
<b>4.1 REQUISITOS FUNCIONAIS E NÃO FUNCIONAIS.....</b>	<b>51</b>
4.2 MODELAGEM DA SIMULAÇÃO.....	53
4.3 MODELAGEM DA PERSISTÊNCIA.....	55

4.4 LINGUAGEM DE IMPLEMENTAÇÃO .....	62
4.5 DISCUSSÃO E LIÇÕES APRENDIDAS.....	63
4.6 CONCLUSÃO .....	64
5 ESTUDOS DE CASO .....	65
5.1 CARACTERIZAÇÃO DO EXPERIMENTO.....	66
5.1.1 Topologia utilizada nos cenários .....	66
5.1.2 Definição dos Canais .....	69
5.2 PRIMEIRO CENÁRIO.....	72
5.3 SEGUNDO CENÁRIO.....	82
5.4 TERCEIRO CENÁRIO .....	91
5.5 DISCUSSÃO.....	99
5.6 CONCLUSÃO .....	101
6 CONCLUSÕES E PERSPECTIVAS FUTURAS.....	102
6.1 CONCLUSÕES.....	102
6.2 PERSPECTIVAS FUTURAS .....	104
REFERÊNCIAS .....	105
APÊNDICE A - PUBLICAÇÕES DO AUTOR .....	109
ANEXO A – REGRAS DE ENCAMINHAMENTO .....	110
ANEXO B – EVENTOS DO ARCABOUÇO .....	111
ANEXO C – EXEMPLO DE XML .....	113
ANEXO D – METADADOS DAS TABELAS DO CONTEXT DATABASE .....	117
ANEXO E – XMLs DOS CENÁRIOS DO ESTUDO DE CASO.....	125
ANEXO F – SQLs DOS CENÁRIOS DO ESTUDO DE CASO .....	126
ANEXO G – TCLs DOS CENÁRIOS DO ESTUDO DE CASO.....	127
ANEXO H – COMANDOS SQL EXECUTADOS NO CONTROLADOR .....	129

## 1 INTRODUÇÃO

Desde a criação das redes de computadores no final da década de 60 até os dias de hoje, esse instrumento de comunicação vem se tornando essencial para todos os segmentos da sociedade. A comunicação e a informação têm ficado cada dia mais rápidas e a necessidade de interação cada vez mais latentes em todos os cantos do globo. Com a chegada da internet cerca de 25 anos atrás à grande maioria dos usuários, a evolução dos serviços e das aplicações tem sido exponencial e tem causado a adesão de cada vez mais pessoas para a rede mundial.

As aplicações disponibilizadas na rede mundial de computadores têm trazido não somente a interação entre usuários, mas também proporcionaram a utilização de novos tipos de conteúdo e atendido a novos tipos de perfis de usuários, que têm demandado a utilização cada vez maior de recursos de infraestrutura para a comunicação e armazenamento.

Essas aplicações têm surgido em uma velocidade igual ou maior que a evolução da internet. Com isso, o grande desafio atual é permitir que a rede mundial esteja preparada para essa demanda crescente de comunicação de diversas formas, não somente em chats, que eram mais leves pois continham somente textos, mas também das aplicações que disponibilizam áudios e vídeos, tais como as aplicações de vídeo-chamadas em tempo real, entre outras, que exigem da rede uma série de parâmetros de qualidade para seu funcionamento.

A solução encontrada cerca de 15 anos atrás para permitir um melhor gerenciamento dos fluxos de dados na rede foi sobretudo baseada na qualidade de serviço (QoS). Dentre as soluções de QoS que tiveram maior destaque foram os Serviços Integrados (IntServ) (BRANDEN et al., 1994), que ganharam notoriedade através da utilização de reserva de recursos, e depois os Serviços Diferenciados (DiffServ) (ZHAO et al., 2000), que, utilizando o provisionamento de recursos, esta última foi uma melhor solução em muitos aspectos, mas que, no entanto, não conseguiu resolver efetivamente todos os problemas, para que a rede funcionasse de forma adequada para diferentes tipos de aplicações.

Atualmente, muitas empresas utilizam soluções híbridas, como a agregação de reservas, característica do Intserv, no Diffserv (RFC 3175 e RFC 4860) ou MPLS/GMPLS (ROSEN et al., 1999), que, também, não resolvem todos os problemas, mas que melhoram o desempenho da rede na maior parte do tempo. Algumas dessas soluções são baseadas na Engenharia de tráfego que permitem a definição das melhores rotas para a entrega de tráfego baseada em modelos matemáticos. No entanto, mesmo essas soluções ainda não têm sido eficazes o suficiente para algumas situações específicas, prejudicando o funcionamento da rede e

degradando o canal de comunicação para todo o tráfego na rede.

Portanto, nos últimos anos, as métricas de rede tais como largura de banda, atraso, variação do atraso, taxa de erros, entre outros têm sido aspectos preponderantes para a escolha de um canal de comunicação de qualidade de serviço. Além disso, e com proporcional importância, tem-se os aspectos relacionados aos dispositivos sendo usados na comunicação, sejam aspectos relacionados à sua capacidade ou mesmo às suas funções. Tal característica de determinado dispositivo pode ou não afetar a qualidade da comunicação.

Nesse cenário mais complexo é importante que a escolha do melhor caminho para a entrega dos dados possa ser feita usando diferentes informações e não somente as métricas de rede usadas na maioria das soluções que encaminhamento adaptativo. Sendo assim, além da rede e seus ativos, devem, também, ser analisadas as informações de usuários e dos dispositivos usados.

Dentro dessa evolução, surgem, cada vez mais importantes, os contextos aos quais essas aplicações precisam ter disponíveis, para que esses possam ser usados ou estejam disponíveis para uso por qualquer usuário na rede, seja em um desktop, seja um smartphone ou em um *tablet*, por exemplo.

Além do desafio de atender aos contextos das novas aplicações, existe ainda o desafio da uniformidade da qualidade dos parâmetros de QoS nas diversas redes existentes na internet, como redes de celulares com as suas tecnologias, redes sem fio corporativas, redes de provedores, redes cabeadas corporativas, etc. Esse universo heterogêneo de tecnologias precisam ser compatíveis com as necessidades mínimas de funcionamento de algumas aplicações na rede, por isso a descrição do seu contexto pode ser útil de forma a garantir que o seu funcionamento não seja prejudicado e que funcionem com uma qualidade satisfatória.

## 1.1 PROBLEMÁTICA

A partir do exposto na contextualização foi possível identificar alguns problemas relacionados à temática abordada neste trabalho de mestrado, que são:

- Dificuldade no gerenciamento de diferentes tipos de tráfego e o tratamento adequado dos mesmos;
- Inexistência de soluções integradas para o tratamento de aspectos de contexto como experiência do usuário e características dos dispositivos;

- Demorada adaptabilidade do tráfego da maioria das soluções atuais, onde a maioria das soluções não faz adaptação do tráfego *online*, ou seja, durante a ocorrência da comunicação;
- Inexistência de solução que suporte fim-a-fim todos os requisitos da comunicação e possa identificar problemas que estejam ocorrendo;
- Falta de identificação e notificação da indisponibilidade de recursos que impeçam ou mesmo interrompam a comunicação na rede, e;
- Soluções atuais desconhecem de forma ampla a capacidade dos recursos existentes para a comunicação na rede.

A partir dos problemas identificados, são apresentadas na próxima seção as motivações para o desenvolvimento desse trabalho.

## 1.2 MOTIVAÇÕES

De maneira a mitigar os problemas identificados anteriormente, é necessária uma solução para encaminhamento adaptativo, que aborde não somente as métricas de rede já utilizadas, como também contemple os requisitos e expectativas do usuário, como o seu comportamento e o comportamento do sistema computacional, como as características do dispositivo e do sistema operacional.

Dessa forma, o arcabouço *Context-Aware Adaptive Routing Framework* (CAARF) (OLIVEIRA et al., 2014), ilustrado na Figura 1, tem sido proposto de forma a viabilizar as tomadas de decisões de encaminhamento de dados baseadas em contexto, sem desprezar as informações comumente usadas para Qualidade de Serviço (QoS). Essa solução permite avaliar não somente os parâmetros usualmente conhecidos como largura de banda, latência, variação no atraso e taxa de erros, mas também parâmetros como o nível de satisfação do usuário em relação à entrega de conteúdo, a capacidade de processamento momentâneo de um dispositivo ou mesmo a capacidade de memória momentânea do mesmo (SILVA, 2015; OLIVEIRA, 2015; SPINOLA, 2015; MUAKAD, 2015).

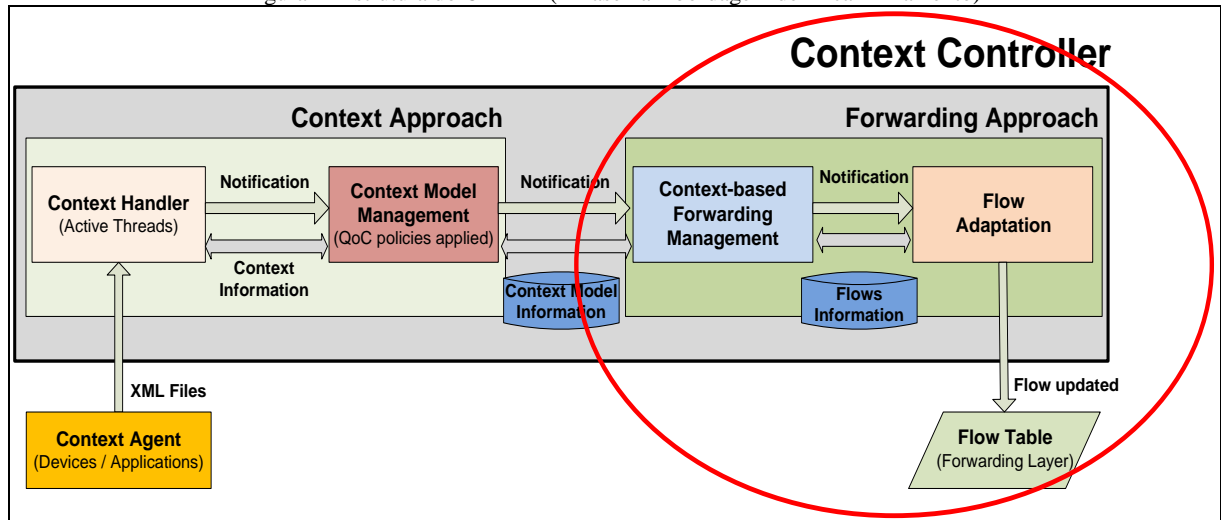
O projeto do arcabouço CAARF está sendo proposto e implementado como resultado da contribuição integrada de quatro alunos de uma equipe do Programa de Mestrado em Sistemas e Computação da Universidade Salvador (UNIFACS). Neste projeto, os diferentes alunos (Joatham Silva, André Oliveira, Sérgio Spínola e Carlos Frederico Muakad) estão realizando



contribuições distintas e complementares de forma a proporcionar a modelagem, implementação e validação do arcabouço CAARF.

Sendo assim, a principal motivação apresentada nesse trabalho foi contribuir com a simulação de uma arquitetura para o arcabouço proposto através de uma abordagem para o encaminhamento adaptativo baseado em contexto, conforme ênfase ilustrada na Figura 1.

Figura 1- Estrutura do CAARF (Ênfase na Abordagem de Encaminhamento)



Na próxima seção, serão apresentados os objetivos gerais e específicos desse trabalho de mestrado.

### 1.3 OBJETIVOS

O trabalho realizado nesta dissertação tem como objetivo geral apresentar uma metodologia para a implementação da gerência de encaminhamento adaptativo baseado em contexto, a partir do arcabouço CAARF.

A contribuição realizada visa provar a importância da utilização integrada de métricas orientadas ao usuário, aos seus dispositivos e à rede na avaliação do melhor caminho para adaptar os fluxos de tráfego das aplicações. Dessa forma, o objetivo geral pode ser dividido em quatro objetivos específicos:

- **Estudo e definição de um modelo e arcabouço para o tratamento do contexto** e a representação das informações necessárias para o encaminhamento de dados, mostrando a factibilidade do modelo;

- **Definição e modelagem de regras de encaminhamento** baseadas em informações contextuais dentro da arquitetura de contexto proposta;
- **Simulação da arquitetura proposta para a gerência de encaminhamento adaptativo** baseada no arcabouço *Context-Aware Adaptive Routing Framework (CAARF)*, e;
- **Validação da solução proposta através da realização de estudos de casos** para otimização do tráfego baseado em contexto utilizando um cenário real.

Na seção seguinte será apresentada a metodologia usada nesse trabalho.

#### 1.4 METODOLOGIA

Para a realização do trabalho proposto, a seguinte metodologia foi adotada:

- Realização de um estudo comparativo das soluções e modelos existentes para o encaminhamento baseado em contexto com Engenharia de Tráfego;
- Estudo de trabalhos relacionados para auxiliar na proposta e posterior definição de um modelo e arcabouço para o tratamento de contexto e a representação das informações necessárias para o encaminhamento de dados;
- Definição e modelagem do arcabouço CAARF e proposta dos mecanismos e regras necessários para a gerência de encaminhamento baseada em contexto;
- Simulação do módulo de gerência de encaminhamento baseado em contexto, e;
- Validação da solução implementada através de estudos de caso baseados em cenários fictícios.

Como resultado desse trabalho, na seção seguinte serão apresentadas as principais contribuições.

#### 1.5 PRINCIPAIS CONTRIBUIÇÕES

Nesse trabalho foram realizadas pesquisas sobre temas que dão base ao **roteamento adaptativo de tráfego** que é parte do tema abordado. Foram pesquisadas informações sobre engenharia de tráfego, qualidade de serviço (QoS) e roteamento adaptativo de tráfego propriamente dito. Além desses temas, foram feitas pesquisas sobre Qualidade de Dispositivo (QoD), Qualidade de Experiência (QoE) e Qualidade de Contexto (QoC).

Todos esses temas são abordados nesse trabalho nos seguintes aspectos: em relação ao roteamento adaptativo, ou seja, a **capacidade que o modelo e o arcabouço se apresentam para se adaptar, dinamicamente, durante o funcionamento da rede**, às mudanças que podem ocorrer no fluxo de suas aplicações, e; em relação às informações contextuais, que normalmente, de forma mais restrita, tem sido abordadas na maioria das soluções e pesquisas sempre com uma visão centrada na rede e no funcionamento da mesma somente.

Sendo assim, outra importante contribuição tratada nesse trabalho é a ampliação da **visão de atendimento para um foco maior no usuário** e nas necessidades do mesmo para entregar a ele uma rede com uma melhor qualidade na comunicação. Sendo assim, o usuário passa a ser não somente um agente passivo, que aceita as escolhas de encaminhamento feitas pela rede, mas também um agente ativo, que interage com a rede através das suas experiências que são captadas pelas aplicações e, assim, a infraestrutura pode entender melhor as necessidades para melhor atendê-los.

Na forma abordada nesse trabalho, a **interação entre rede e usuário** é, então, tratada com mais importância, pois a estrutura da rede deve ser construída e idealizada para atender a quem a usa. Sendo assim, no contexto desse trabalho, toda a análise do melhor caminho usa efetivamente as informações coletadas dos usuários para atender a ele com uma comunicação de melhor qualidade.

Além disso, outra relevante contribuição foi o **desenvolvimento de um arcabouço genérico que trata das questões de roteamento adaptativo de tráfego baseado em informações contextuais**. Para tanto, são apresentados os conceitos relacionados a esse arcabouço bem como suas estruturas, seu funcionamento, a interação entre seus módulos e as entradas e saídas que ocorrem em cada módulo e a interação entre os módulos descritos. Bem como, eventos que são tratados, regras utilizadas e aplicadas e a interação da estrutura com a camada de encaminhamento.

Com maior especificidade nesse trabalho, é apresentada a maneira com a qual este arcabouço trata o encaminhamento dos fluxos e como as informações coletadas são tratadas pelo controlador do arcabouço para estabelecer as escolhas de novos caminhos ou a manutenção dos caminhos existentes.

Por fim, também como contribuição, foram realizados alguns **estudos de caso** para ilustrar o funcionamento do arcabouço utilizando uma **simulação de uma arquitetura específica para**

**demonstrar a realização da gestão de encaminhamento de tráfego usando informações contextuais.**

Na seção seguinte, será apresentada a organização dessa dissertação.

## 1.6 ORGANIZAÇÃO DA DISSERTAÇÃO

Essa dissertação, além do capítulo introdutório que está sendo apresentado, foi desenvolvida e organizada da seguinte maneira:

- **Capítulo 2 (Revisão conceitual):** nesse capítulo é feita uma revisão conceitual e literária dos principais aspectos relacionados ao estudo dessa dissertação, além da identificação e estudo dos principais trabalhos existentes na literatura e relacionados com o tema dessa dissertação;
- **Capítulo 3 (Arcabouço):** nesse capítulo é apresentado o arcabouço de roteamento adaptativo baseado em contexto e suas questões relacionadas à gestão de encaminhamento;
- **Capítulo 4 (Simulação):** nesse capítulo é apresentado o ambiente da simulação e os detalhes da sua execução;
- **Capítulo 5 (Estudo de Caso):** nesse capítulo são apresentados alguns estudos de caso que visam validar a arquitetura apresentada no capítulo 4;
- **Capítulo 6 (Conclusões e perspectivas):** nesse capítulo são apresentadas as conclusões do trabalho e as perspectivas de trabalhos futuros.

## **2 REVISÃO CONCEITUAL E DE LITERATURA**

Nesse capítulo serão abordados os principais conceitos relacionados ao tema desse trabalho. Antes do proposto, foi realizado um estudo amplo sobre os seguintes conceitos: qualidade de serviço, qualidade de experiência, qualidade de contexto, qualidade de dispositivo, encaminhamento adaptativo e engenharia de tráfego.

Todas as pesquisas feitas para a realização desse trabalho foram balizadas pelos temas citados acima em alguns aspectos, seja o caráter adaptativo das soluções de encaminhamento, seja a otimização dos recursos proporcionado pelas soluções de engenharia de tráfego. O objetivo desse amplo estudo foi reunir o maior número de soluções estudadas e testadas a fim de auxiliar esse trabalho no direcionamento dado, visando alcançar uma solução de encaminhamento adaptativo.

Juntamente com essa ideia, convergiu a atual importância de outros aspectos, que não os parâmetros de rede, para a obtenção de uma melhor qualidade no uso de algumas aplicações na rede. Tanto quanto a variação do atraso ou mesmo a largura de banda, a capacidade de processamento de um dispositivo ou mesmo a falta de determinados recursos nesse ou até mesmo a percepção do usuário no uso de determinada aplicação, apresentaram-se como aspectos importantes para atingir uma melhor qualidade na comunicação dentro de uma rede.

Como é apresentado nas seções desse capítulo, em inúmeros estudos, as soluções abordadas tiveram como objetivo dar uma maior dinâmica no tráfego da rede, proporcionando uma qualidade aceitável, para que todas as aplicações pudessem utilizar a rede como se estivessem sozinhas ou como se houvesse uma rede para cada par de estações que estivessem se comunicando. Esses estudos buscaram proporcionar ao usuário final uma comunicação estável durante o seu uso, proporcionando a melhor qualidade possível no momento da comunicação.

### **2.1 CONCEITOS E DEFINIÇÕES**

Nessa seção são descritos alguns conceitos e definições usadas nesse trabalho.

#### **2.1.1 Qualidade de Serviço**

Com o advento do crescimento de aplicações de multimídia, dando ênfase em voz e vídeo além dos dados, fez-se necessário, além de técnicas de redução de congestionamento, o fornecimento, por parte das redes, de parâmetros de desempenho com uma determinada qualidade, que

permitiria assim uma comunicação fim-a-fim confiável, ou ainda a inviabilidade de sua utilização (MELLOUK, 2008).

A Qualidade de Serviço (QoS) busca atender às expectativas do usuário visando um melhor tempo de resposta e uma melhor qualidade na comunicação, que, em muitos momentos, é subjetiva.

A Qualidade de Serviço visa, principalmente, atender às necessidades da aplicação, ou seja, do que essa requisita da infraestrutura de rede, a fim de que tenha um bom funcionamento, atendendo assim às necessidades do usuário. Estes requisitos são traduzidos em parâmetros que indicam o desempenho da rede. Alguns desses requisitos são, por exemplo, o atraso máximo sofrido pelo tráfego da aplicação entre o computador origem e destino ou mesmo variação no atraso do caminho entre origem e destino.

Outra forma de definir QoS é como sendo um conjunto de características quantitativas ou qualitativas (LU, 1996). Estas características são chamadas de parâmetros de especificação da QoS. O uso da Qualidade de Serviço serve para mensurar a qualidade dos serviços oferecidos em uma rede de computadores, ou seja, este conjunto de parâmetros mostra o quanto essa rede é eficiente no atendimento das expectativas de seus usuários através dos serviços oferecidos.

Esse conceito, desde sua concepção, tem focado, principalmente, na rede, porém, ultimamente, tem evoluído para uma dimensão mais ampla, atendendo às inúmeras formas de interação entre usuário e sistema. Como foi citado, anteriormente, a QoS envolve, principalmente, aspectos de rede, porém tem sido importante e tem afetado aspectos relacionados à própria percepção do usuário, através das aplicações usadas por esses, e aos recursos disponíveis na infraestrutura, como equipamento do usuário ou aos equipamentos da rede.

Além dos aspectos de desempenho e percepção, o QoS tem relação ainda com requisitos como segurança, privacidade, contabilidade, política de preços dos serviços, estabelecimento e monitoração de contratos de serviços e grau de disponibilidade da rede, como tempo médio entre falhas ou mesmo tempo médio de reparo.

QoS é o termo utilizado para identificar as inúmeras técnicas que permitem qualificar as redes, garantindo que aplicações diferentes recebam tratamento diferente, ou seja, transmissões que exigem maior qualidade da rede, recebem maior prioridade.

### **2.1.2 Qualidade de Experiência**

Esta questão tem se tornado cada vez mais crítica na medida em que às aplicações tradicionais de rede, que compreendem transferência de arquivos, mensageria, comunicação através das redes sociais, vídeos recreativos, como o *Youtube*, e-mail e suas formas correlatas, acesso a sites, e-commerce, serviços financeiros e de consultas, somam-se novas aplicações multimídia e serviços que envolvem voz e vídeo interativo, requerendo um gerenciamento de expectativas de usuário muito mais complexo de administrar.

Ao universo de aplicações de voz e vídeo sensíveis ao QoS, anteriormente restritas ao VoIP e videoconferência, vêm sendo acrescentadas aplicações que envolvem um grau de interatividade crescente, tais como games interativos – muitos dos quais requerem sincronismo entre os jogadores, e outras formas de interação multimídia, notadamente videoconferência em escala mais disseminada, *e-learning* por meio de ferramentas de *webminar*, conferências online, vídeo-sob-demanda e a categoria denominada “telepresença”.

A mensuração da satisfação dos usuários destas aplicações sob a ótica do cliente representa um desafio ampliado em termos de medição, requerendo a definição de métricas e aferição que não podem ser obtidas diretamente a partir dos parâmetros clássicos de QoS somente. O que se tem observado é que a percepção do usuário cada vez mais contraria a convicção de que políticas e regras de QoS bem definidas são uma garantia absoluta de aprovação da qualidade de experiência do usuário.

Qualidade de Experiência (QoE) trata, portanto, da quantificação da percepção do subjetivo, mudando o ponto de vista da aferição de qualidade da rede para o usuário. Portanto, é “*user-centric*” em contraposição ao “*network-centric*”, envolvendo fatores cognitivos, comportamentais e psicológicos (MITRA et al., 2011), que passa a estudar a perspectiva e compreensão da percepção do usuário, contemplando expectativas e experiência com a aplicação.

QoE estabelece um conjunto de parâmetros que estão relacionados com a percepção do usuário. Esses parâmetros são vistos como a quantificação da percepção do subjetivo, mudando o ponto de vista da aferição de qualidade da rede para o usuário. A Qualidade de Experiência (QoE) é, portanto, um parâmetro que mede o sentimento do usuário quando do uso de determinadas aplicações, que, geralmente, exigem tal parâmetro.

O desafio que se apresenta na compreensão dos fatores de QoE é determinar que fatores se pretende medir objetivamente, mapeando, antecipando e prevendo a percepção subjetiva com base em métricas objetivas. Deseja-se, portanto, inferir o nível subjetivo de satisfação de usuário a partir de métricas objetivas. Esta é uma característica que embasa a ação proativa em

relação à percepção do usuário e não somente a percepção do bom funcionamento da rede, que não necessariamente traduz a “aferição subjetiva do usuário”.

Em outras palavras, o problema com o qual os provedores se defrontam é que não necessariamente boas práticas de QoS ou *Traffic Engineering* resultam em satisfação do usuário, isto é, o QoE, o que pode levar à perda inesperada de clientes, ainda que os parâmetros de QoS aparentemente estejam bem ajustados. De maneira geral, o QoS é uma solução estritamente orientada aos parâmetros de rede enquanto, o QoE é uma solução orientada ao usuário (ALRESHOODI ; WOODS, 2013). O QoE pode ser entendido também como uma cobertura de quatro domínios: contextual, tecnológico, negócios e humano (LAGHARI ; CONNELLY, 2012), entendido como uma pseudo-camada entre a aplicação e a rede (ALRESHOODI ; WOODS, 2013).

O conceito de QoE tem sido, ultimamente, associado às condições pelas quais os usuários percebem o ambiente de utilização de determinada aplicação. Tais parâmetros como o ruído do ambiente ou mesmo o *Mean Opinion Score (MOS)* que, dependendo da aplicação, podem interferir na qualidade da comunicação, independente da infraestrutura instalada para atender àquela aplicação em questão.

O MOS (*Mean Opinion Score*), cujos valores variam entre má experiência, pobre, aceitável, bom e excelente (SKORIN-KAPOV et al., 2012) tem sido um dos parâmetros de QoE dos mais usados. Esta é uma medida subjetiva que pode ser obtida por meio de uma pesquisa de satisfação ou por inferência recolhidas a partir da correlação com outros parâmetros.

Atualmente, algumas aplicações têm feito uso da experiência do usuário para efetuar ajustes internos para mudar a percepção do usuário durante a utilização.

### **2.1.3 Qualidade de Dispositivo**

Serviços são executados em componentes de hardware, estes dispositivos possuem também uma qualidade, chamada Qualidade de Dispositivos (QoD). A QoD é qualquer informação sobre as características técnicas e capacidades de um dispositivo (KIM ; LEE, 2006).

Os dispositivos que usam a rede possuem características técnicas que os diferenciam dos outros, sejam recursos de processamento, memória, armazenamento secundário ou mesmo recursos como GPS habilitado, sensores, entre outros recursos que podem viabilizar ou não o uso de determinadas funções em aplicações. Dessa forma, determinados dispositivos podem ser



viáveis serem usados em determinadas comunicações ou aplicações enquanto outros não possuem requisitos mínimos para usarem determinadas funções.

#### **2.1.4 Encaminhamento Adaptativo**

Em qualquer rede de computadores, uma das funções básicas da camada de rede é a de realizar o encaminhamento de pacotes entre uma origem e um destino. Esse encaminhamento marca o percurso entre origem e destino, podendo passar por muitos roteadores. Em muitas empresas, esses caminhos são marcados de forma manual, ou seja, os administradores de rede fixam o direcionamento de certo tráfego vindo de um determinado local e encaminham para outro de forma estática, sem necessidade de nenhuma inteligência nos roteadores.

Em outras organizações, é feito o uso de algoritmos de roteamento no qual realizam a função de escolha do caminho, conforme definido na sua estrutura de atendimento. Esse roteamento é considerado o roteamento dinâmico, sem que haja a interferência de quem quer que seja na mudança ou adaptação de rotas.

Além dessas técnicas, existe a adaptação do roteamento conforme avaliações feitas por determinada solução, como nos mecanismos de Engenharia de Tráfego, exemplo do MPLS ou GMPLS (ROSEN et al., 1999). Essa solução e outras utilizam dispositivos que realizam de forma dinâmica e, de acordo com critérios estabelecidos pela própria solução, uma adaptação do tráfego para melhor atender aos usuários ou mesmo para otimizar o acesso a recursos e atender a mais usuários sem tanta preocupação com a qualidade na comunicação.

Portanto, não somente os algoritmos de roteamento proporcionam uma adaptação do tráfego como, também, outras soluções utilizam formas distintas para atingir o mesmo objetivo.

#### **2.1.5 Engenharia de Tráfego**

Entende-se como Engenharia de Tráfego um conjunto de mecanismos capazes de organizar a fluência do tráfego através da rede para que sejam evitados os congestionamentos causados pela utilização da rede (FORTZ, 2002). A Engenharia de Tráfego tem foco na otimização do uso dos recursos da rede.

Geralmente, a Engenharia de Tráfegos e utiliza de princípios tecnológicos e científicos (através de modelos matemáticos) para medir, modelar, caracterizar e controlar o tráfego na Internet. Além disso, essas mesmas técnicas e conhecimentos são utilizadas para atingir determinados objetivos de desempenho (AWDUCHE et al., 1999).

O principal objetivo na Engenharia de Tráfego é tornar a operação da rede mais eficiente e confiável, enquanto que, ao mesmo tempo, otimiza o desempenho para uma melhor utilização da rede por parte dos usuários. Em determinadas organizações, o uso da Engenharia de Tráfego é extremamente importante, pois, se não fossem tais mecanismos, o custo de aquisição de equipamentos seria muito alto, para atendimento da demanda.

Além disso, é de extrema utilidade para os grandes provedores pelo mesmo motivo: otimização do uso dos recursos para não aumentar o custo de equipamentos e proporcionar uma melhor qualidade do canal para os usuários.

Outra grande utilidade da Engenharia de Tráfego é no atendimento de requisitos de QoS, pois possibilita a mudança do tráfego, conforme avaliação na rede.

Os objetivos de desempenho da Engenharia de Tráfego podem ser classificados como:

- **Baseados na otimização de recursos.** Estão relacionados com a otimização dos recursos de rede, impedindo congestionamentos e permitindo melhor utilização pelos usuários;
- **Baseados na melhoria do tráfego:** Estão relacionados com aspectos de manutenção das garantias de QoS dos fluxos de dados (ou agregações de fluxos).

Dentre as mais utilizadas soluções de Engenharia de Tráfego, tem-se o MPLS, que implementa um roteamento adaptativo usando rótulos de marcação e fixação do caminho mais adequado para o fluxo ou agregações. Existe a possibilidade, também, de serem implementadas soluções manuais de Engenharia de Tráfego mais orientadas à otimização dos recursos.

### 2.1.6 Contexto

A computação ubíqua e pervasiva (WEISER, 1991) é um paradigma caracterizado pela presença de dispositivos móveis, que cada vez mais fazem parte do cotidiano das pessoas. Estes dispositivos possuem uma considerável capacidade de processamento, recursos de comunicação sem fio e armazenamento de dados. Possuem ainda, funcionalidades diversificadas e interfaces como GPS, rádio e TV, reprodutores de áudio, câmeras digitais entre outros, sendo utilizados em aplicações de diversas áreas como: indústria, comércio, turismo, saúde, entretenimento, etc.

Este tipo de computação possui forte ligação com as características do mundo físico, bem como aquelas apresentadas pelos perfis de seus usuários. Tais informações são chamadas de contextos e representam o elemento de entrada para a computação ciente ou sensível ao contexto (LUO REN et al., 2009).

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

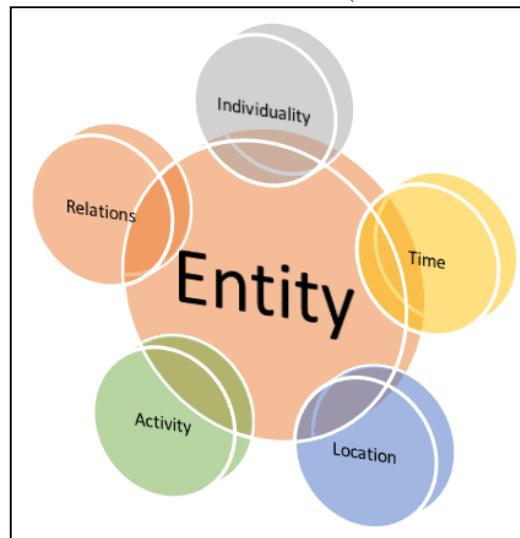
Contexto pode ser definido como

qualquer informação que possa ser usada para caracterizar a situação das entidades (ou seja, se uma pessoa, objeto ou local) que são considerados relevantes para a interação entre um usuário e uma aplicação, incluindo o próprio utilizador e a aplicação. Contexto geralmente está relacionado à localização, identidade e estado de pessoas, grupos e objetos computacionais e físicos. (DEY et al., 2001).

Diversas são as classificações de contexto encontradas na literatura (CHEN; KOTZ, 2000), (VIANA et al., 2007), (EMMANOUILIDIS et al., 2012), (SCHUSTER et al., 2012). Por exemplo, para (CHEN; KOTZ, 2000) o contexto apresenta quatro dimensões composto por: contexto computacional, contexto físico, contexto de tempo e contexto do usuário.

O uso de contexto pode ser aplicado a entidades, pessoas, lugares, ou mesmo a um objeto relevante para a aplicação, através da definição das características como individualidade, atividade, localização e de tempo e até mesmo relações com outras entidades (ZIMMERMANN et al., 2007), tal como representado na Figura 2.

Figura 2 - Características fundamentais de contexto (ZIMMERMANN et al., 2007).



Baseado nestas informações contextuais, os serviços podem ser adequar às características do ambiente atual ou ainda do perfil do usuário, promovendo assim, mais otimização e personalização e automaticamente a satisfação do usuário.

A Qualidade de Contexto (QoC) é qualquer informação que descreve a qualidade da informação que é usada como informação de contexto (BUCHHOLZ et al., 2003).

A definição de QoC em (KRAUSE, 2005) está relacionada a qualquer informação inerente que descreve o valor da informação de contexto para uma aplicação específica. Isso inclui informações sobre o processo de provisionamento que a informação foi submetida (“histórico”, “idade”), mas não tratam de estimativas sobre os passos de provisionamentos futuros.

Após a apresentação dos principais conceitos utilizados nesta dissertação de mestrado, a próxima seção discute algumas das contribuições existentes na literatura relacionadas aos conceitos apresentados anteriormente.

## 2.2 ESTADO DA ARTE

### 2.2.1 Qualidade de Serviço

Qualidade de serviço é a capacidade de fornecer prioridades distintas para diferentes aplicações, usuários ou fluxos de dados.

São várias as soluções de Qualidade de serviço e dentre eles muitas são baseadas em *frameworks* para gerenciamentos de QoS, que muitas vezes são usados para resolver problemas específicos, um exemplo disso é o que relata os autores (KUSMIEREK et al., 2002) em seu artigo “*An Integrated Network Resource and QoS Management Framework*” onde é proposto uma dissociação do plano de controle de rede do plano de dados.

Da mesma forma, como no artigo “*A QoS Oriented Framework for Adaptive Management of Web Service based Workflows*” escrito por (PATEL et al, 2002), onde é desenvolvido um modelo de QoS como uma estrutura para fluxos de trabalhos baseados em serviços web.

Na grande maioria dos artigos pesquisados, relacionados a QoS, os autores seguem um padrão para a definição de modelo para seus *frameworks* (HONG; HONG, 2003), (KUSMIEREK et al., 2003), (DEORA et al., 2011), (AGBOMA et al., 2008), (AL-ALI et al., 2002), (YERIMA, 2011):

- **Parâmetros gerais:**

- Desempenho (Latência), ou seja, o tempo necessário para prestar serviços entre solicitantes de serviços e provedores;
- Desempenho (*throughput*), o número de pedidos solicitados em um determinado período;
- Falhas, este parâmetro está relacionado com o número de falhas de um serviço, em um intervalo de tempo;
- Custo, o custo da execução do serviço.

- **Parâmetros específicos:**
  - Disponibilidade, a probabilidade de que o serviço estará disponível em algum período de tempo;
  - Segurança, ou seja, a confidencialidade;
  - Acessibilidade, está associada ao fato do serviço estar disponível ou não;
  - Regularidade, um aspecto de qualidade, que lida com questões de conformidade do serviço com as regras, a lei, a conformidade com a norma, bem como o acordo de nível de serviço estabelecido (SLA's).
- **Tarefas Específicas**
  - Tarefas específicas relacionadas com a qualidade da saída ou o tipo de serviço oferecido. Exemplo, para modelos baseados na camada aplicação, podemos destacar videoconferência, telemedicina, educação remota ou até mesmo navegadores web como as especificações QoS, que por sua vez terá como principais funções do seu *framework*, controles de prioridade, gerenciamento de *buffer*, UPC e etc. Já para aqueles que desejam direcionar seus modelos para nível de usuário, tem que ser levado em consideração aspectos como perda, atraso e orientar a solução a engenharia de redes de longo prazo.

A grande maioria dos artigos pesquisados apresentam seus *frameworks* e muitos deles seguem um padrão para criação, que são definidos por 4 parâmetros principais:

- **Gerenciador de Recursos da rede/aplicação** (HONG; HONG, 2003), (KUSMIEREK et al., 2003), (AL-ALI et al., 2002), (YERIMA, 2011), (DEORA et al., 2011), (AGBOMA et al., 2008) e (CHOWDHURY et al., 2002).
  - Negocia SLAs (*Service Level Agreement*) com clientes e comunica os parâmetros associados a um SLA com o gerente de recurso correspondente, além de garantir SLAs conforme os recursos alocados;
- **Gestão de Contexto e adaptação de cada subsistema** (HONG; HONG, 2003), (AL-ALI et al., 2002), (YERIMA, 2011), (DEORA et al., 2011), (AGBOMA et al., 2008) e (CHOWDHURY et al., 2002)
  - É inserida no desenho da estrutura para permitir a gestão consciente de recursos do contexto;
- **Medição e monitoramento** (HONG; HONG, 2003), (AL-ALI et al., 2002), (YERIMA, 2011) (AGBOMA et al., 2008) e (CHOWDHURY et al., 2002)
  - É necessário pois alimenta a base de dados dos gerenciadores de Recursos da rede,

monitorando os dados obtidos dentro da rede;

- **Gerenciador de QoS de cada aplicação** (HONG et al., 2003), (KUSMIEREK et al., 2003), (AL-ALI et al., 2002), (YERIMA. 2011) e (CHOWDHURY et al., 2002)
  - Usado quando uma aplicação tem requisitos de QoS específicos.

Partindo desses 4 pilares, a maioria dos artigos que possuem *framework*, para gerenciar QoS em suas aplicações específicas, conseguem estruturar seu modelo e conseqüentemente obter sucesso na implementação do mesmo.

### 2.2.2 Qualidade de Dispositivo

Dentro das informações contextuais obtidas, estão a de QoD, onde, por exemplo, o sistema de posicionamento global (*Global Positioning System – GPS*) de cada dispositivo podem ter níveis de precisão diferentes, ou mesmo um dispositivo pode não fornecer alguns parâmetros em comparação a outro pelo simples fato de não ter tal hardware ou a capacidade de coletar tais informações (BUCHHOLZ et al., 2003).

Sendo assim, a QoD vai fornecer informações referentes as características técnicas de cada dispositivo, e suas capacidades (NAZARIO et al., 2012), (VIEIRA et al., 2009).

Alguns estudos comparativos de contribuições relacionadas com roteamento que tenha conhecimento do contexto foram realizados em (PETZ et al., 2012), (YASAR et al., 2010), (WENNING et al., 2009) e (MASCOLO et al., 2006). A maioria dessas contribuições estão relacionadas com a utilização do contexto aplicado, principalmente, às redes sem fio.

Nestes estudos, o uso de contexto permitiu melhoramentos, principalmente em: estabilidade da ligação de comunicação, o aumento da largura de banda (por diminuir em cima), maior autonomia baterias, atraso mais curto e escalabilidade. Esses ambientes diferem muito de redes cabeadas, principalmente devido à capacidade de armazenamento e processamento de restrições, a limitação a vida da bateria e, em alguns casos, a largura de banda limitada.

### 2.3 ENCAMINHAMENTO ADAPTATIVO

Com a dinâmica de utilização das redes IP atuais, o encaminhamento adaptativo tem sido um meio relativamente eficiente, estudado pelos pesquisadores, para permitir que vários tipos de tráfegos na rede convivam o mais harmoniosamente possível com a infraestrutura existente.

Dessa forma, muitos estudos sendo feitos estão relacionados ao encaminhamento dinâmico ou adaptativo, que procura capacitar a rede na adaptação à dinâmica de funcionamento com aumento de tráfego inesperado, por exemplo. A engenharia de tráfego tem sido um item bastante utilizado como um mecanismo importante na resolução dessa equação complicada entre qualidade da aplicação e a qualidade da banda utilizada para ela. A otimização do uso da banda também é um ponto importante sendo estudado, que é um dos focos principais da engenharia de tráfego com a otimização do uso dos recursos na rede. O estudo dos múltiplos caminhos, possibilitando múltiplas alternativas para os nós, tem sido, também, um importante ponto. Outra abordagem sendo feita é o controle do congestionamento, que viabiliza uma melhor qualidade do canal para todas as aplicações que nele precisam trafegar.

Todas essas abordagens têm sido bastante utilizadas nos estudos sobre encaminhamento adaptativo com algumas pequenas variações a depender do algoritmo ou do modelo que venham a utilizar. Portanto, podem ser classificados de forma genérica os roteamentos adaptativos da seguinte forma:

- Baseados em contexto (MUSOLESI et al., 2005), (WENNING et al, 2009), (PETZ et al, 2012);
- Baseados em engenharia de tráfego (KANDULA, 2005), (FISCHER, 2006), (KARTHIGA, 2012); e
- Baseados em dados probabilísticos (XIE, 2005).

A abordagem do roteamento adaptativo baseado em dados de contextos baseou-se em métricas distintas das usadas em qualidade de serviço, como atraso, latência entre outras. As métricas usadas pelos algoritmos ou pelas soluções usadas pelos protocolos baseados em contexto foram as seguintes: Taxa de mudança de conectividade de um host, Nível de energia, Probabilidade de entrega, Potência do Sinal, Saúde do Nó, Contador de Saltos e Taxa de aprendizado (MUSOLESI et al, 2005), (WENNING et al., 2009), (PETZ et al., 2012).

O algoritmo CAR (MUSOLESI et al., 2005), por exemplo, usa informações de contexto tendo ação antecipada, evitando, com isso, a inundação de pacotes e o tráfego epidêmico. Portanto, tem um caráter proativo no roteamento, não sendo reativo. Além disso, a informação de melhor probabilidade de entrega é um parâmetro importante, podendo, por exemplo, mudar o roteamento dos pacotes se existir host com melhor probabilidade naquele momento. Este algoritmo foi implementado para redes sem fio *Ad-hoc*.

Na solução apresentada em (WENNING et al., 2009), é discutida um arcabouço para roteamento baseado em contexto, incluindo fluxo de mensagem bem como método de decisão de rota. Apesar de apresentar o arcabouço para uma rede de sensores sem fio, pode ser utilizado em qualquer outro tipo de rede. Nessa solução, as informações de contexto usadas são: critério de seleção do nó, regra de combinação de critério parametrizado, regra de restrição de encaminhamento e os valores dos critérios. Além dessas informações, outras informações como potência do sinal, energia no nó e o contador de saltos são usadas na solução.

Em (PETZ et al., 2012), é apresentado um arcabouço baseado em contexto para redes tolerantes a atraso. Nele é descrito sobre um portal de adaptação onde o contexto externo obtido por agentes pode modificar o roteamento interno da rede. Foi implementado o CANC que é um agente codificado e embutido ao roteador para obter informações do mesmo e, também, para obter informações de movimentação, as quais permitem reconfigurar o roteador com a análise destas.

Na solução apresentada em (CHIOCCHETTI et al., 2013), é discutido um novo modelo de comunicação em rede onde as primitivas são baseadas em nome e não em identificadores de nome. Nele é apresentada uma técnica inspirada no arcabouço *Q-Routing*, chamada INFORM, que trata da dinâmica do encaminhamento de requisições de usuários baseadas em ICN (*information Centric networking*).

Outra abordagem utilizada é a do roteamento adaptativo com soluções baseadas em engenharia de tráfego. Nas soluções baseadas nessa abordagem, a engenharia de tráfego é utilizada, principalmente, na otimização do uso dos recursos de banda.

Na solução apresentada em Kandula et al. (, 2005), é apresentado o protocolo TeXcp, que é um protocolo de engenharia de tráfego distribuído online, ou seja, que reage a acontecimentos em tempo real diferentemente de outras soluções que são offline, as quais usam informações pré-computadas para determinadas situações. Em Kandula et al. (, 2005), o foco é dado na máxima utilização do link com a menor banda possível com manutenção do desempenho e do atendimento da demanda de tráfego. Dessa forma, prorroga-se o uso da mesma largura de banda por um longo período de tempo.

Nessa solução, os seguintes parâmetros são utilizados: utilização do canal, utilização do caminho, capacidade do canal, caminhos disponíveis, caminhos ativos. Seu funcionamento ocorre da seguinte forma: O agente TeXcp que reside no roteador de Ingresso e que libera o



tráfego para o egresso por túneis ou múltiplos caminhos. Através de pacotes de feedback descobre quais caminhos estão com baixa ou alta utilização ou mesmo os caminhos com falha. Dessa forma, o balanceamento de carga dos canais é realizado com eficiência.

Em Fischer et al. (, 2006), é apresentado um algoritmo de engenharia de tráfego dinâmico distribuído, chamado REPLEX, que se baseia na política de re-roteamento de jogos que possibilita evitar oscilação e permite rápida adaptação em caso de convergência. Este algoritmo baseou-se em outros algoritmos como o utilizado no TeXcp (KANDULA et al., 2005) e o MATE (ELWALID et al., 2001) ambos, também, utilizando engenharia de tráfego.

Diferentemente das outras soluções que utilizam engenharia de tráfego e que são somente baseados na infraestrutura MPLS, o REPLEX não se restringe somente ao MPLS, podendo ser integrado a outras soluções de roteamento e proporciona baixa sobrecarga de processamento. Além disso, este algoritmo não se baseia simplesmente na distribuição do fluxo para caminhos com mesmo custo, mas especifica como o roteador deve reagir às mudanças de carga do tráfego, seja por efeitos externos ou pelas decisões de outros roteadores. Esta solução procura dividir o tráfego base em cada fluxo utilizando uma técnica de *hashing* (CAO, 2000). Nas simulações se mostrou escalável para grandes redes sendo, com isso, mais adequado que protocolos de engenharia de tráfego entre domínios. Esse algoritmo utiliza-se de parâmetros como: quantidade de saltos, múltiplos caminhos alternativos e, principalmente, latência e utilização do canal.

Outra solução de roteamento baseada em engenharia de tráfego, o AMPLE (WANG et al. 2012) que é um eficiente sistema de gerenciamento e é baseado em engenharia de tráfego que executa controle de tráfego adaptativo usando múltiplas topologias de roteamento virtualizado. Possui dois (2) componentes: o OLWO que faz o provisionamento *offline* da máxima diversidade de caminhos no plano de roteamento, enquanto o ATC ajusta a pequena escala de tempo de atribuição do tráfego no plano de encaminhamento. Esta solução tem como foco tratar a dinâmica do tráfego de forma eficiente. Faz uso de uma abordagem híbrida entre engenharia de tráfego online e engenharia de tráfego *offline*.

Outra solução vista em (XIE et al., 2005), que propõe um esquema de roteamento baseado em informações probabilísticas distribuídas para roteamento auto adaptativo em ambientes dinâmicos e simulados. Em (XIE et al., 2005), o roteamento foca no equilíbrio do *link*, verificando a latência e a utilização do *link* para evitar possíveis sobrecargas. Nessa solução

alguns parâmetros são utilizados como: latência, atraso, contenção na camada MAC e utilização do canal.

Em um estudo mais recente (KARTHIGA; BALAMURUGAN, 2013), é abordado o roteamento adaptativo baseado no provisionamento *offline* dos múltiplos caminhos no plano de roteamento e o online espalhamento da carga do tráfego para um balanceamento de carga dinâmico no plano de encaminhamento.

O algoritmo AMPLE (WANG et al. 2012) é utilizado sobre o protocolo de roteamento IGP virtualizado. Esse algoritmo faz uso de agentes de monitoramento que coletam as condições de tráfego e existe também o gerenciador de engenharia de tráfego que tem uma visão geral do tráfego na rede e pode fazer as escolhas de divisão do tráfego sem causar congestionamento.

## 2.4 CONCLUSÃO

Conforme relatado nesse capítulo, foram realizados vários estudos, foram desenvolvidas várias soluções, porém a maioria delas não baseou sua análise em outras informações, que não fossem as clássicas, como largura de banda, atraso, taxa de erros, entre outras.

Portanto, o arcabouço apresentado nesse trabalho considera não somente os aspectos relacionados diretamente com a rede, mas também os aspectos relacionados ao usuário e ao sistema computacional. Dessa forma, dá-se importância efetiva, durante a escolha do melhor caminho, em aspectos mais ligados ao funcionamento da aplicação em determinado computador com menos recursos ou mesmo com uma pior qualidade dos recursos para determinada comunicação. Além disso, a experiência do usuário é efetivamente avaliada e medida para que se façam escolhas condizentes com a sua experiência durante a comunicação.

### 3 ARCABOUÇO BASEADO EM CONTEXTO

Nesse capítulo é apresentado o arcabouço para o encaminhamento adaptativo de dados baseado em contexto chamado *Context-Aware Adaptive Routing Framework (CAARF)*. CAARF apresenta um novo paradigma no tratamento das questões de tráfego dentro de uma rede, pois permite avaliar não somente os parâmetros usualmente conhecidos, mas também parâmetros como capacidade de processamento momentâneo de um dispositivo ou mesmo a capacidade de memória momentânea de um dispositivo. Além disso, também são levadas em consideração informações extraídas do usuário, que podem mostrar a sua visão em relação a determinada conexão ou mesmo algo no seu ambiente, que afete a sua experiência no uso de uma determinada aplicação, como exemplo o ruído para uma comunicação de um aplicativo de voz sobre IP. Esse novo formato de interpretação de informações de tráfego, que a rede permite captar, faz com que a realidade da rede seja conhecida com um pouco mais de precisão e, assim, permita uma resposta mais precisa em relação a todo esse contexto de avaliação.

Portanto, nesse arcabouço, o contexto, que envolve questões de rede, dispositivo e experiência do usuário, é o foco principal e permite uma avaliação completa de uma série de requisitos que, aparentemente, não fariam diferença no todo da comunicação, mas que refletem com maior realidade o ambiente em que as informações trafegam. Essa avaliação, além de proporcionar uma melhor qualidade do tráfego, pode otimizar, também, o uso de recursos.

O CAARF será detalhado apresentando suas características, a interação entre os módulos e submódulos internos, os papéis dos mesmos, as regras estabelecidas, os eventos que são tratados, as entradas e saídas de cada submódulo e a interação da arquitetura com o domínio de rede onde este arcabouço está inserido.

#### 3.1 MODELO DE CONTEXTO

Antes de ser descrito o arcabouço baseado em contexto para roteamento adaptativo, faz-se necessária uma descrição do modelo de contexto que baseia tal estrutura e que permite compreender as funcionalidades, características, interações, abrangência e funcionamento do CAARF.

O modelo de contexto proposto nesse trabalho é aplicado a uma solução de roteamento adaptativo baseado em informações contextuais. Esse modelo é genérico permitindo vários cenários de rede e, além disso, introduz a adaptação baseada na experiência do usuário,

juntamente com as informações de rede, representadas pela Qualidade de Serviço (QoS) e, também, informações dos dispositivos, representadas pela Qualidade de Dispositivo (QoD).

O modelo de contexto adotado descreve o estado de uma entidade particular, que pode ser um dispositivo, um roteador, um comutador, um usuário, uma aplicação, etc. Seguem algumas informações relacionadas a entidade descrita no modelo e ilustradas na Figura 4. Essas informações são características inerentes à entidade descrita no modelo. Essas características são agrupadas da seguinte forma:

- **Individualidade**—esse item descreve informações particulares sobre a entidade, tais como identificação, endereçamento, protocolos, entre outras informações;
- **Tempo** – esse item descreve informações relacionadas ao tempo tais como última atualização relacionada à entidade;
- **Localização** – esse item descreve aspectos relacionados à localização real ou virtual da entidade tais como localização baseada no GPS, localização de sua casa, edifício, escritório, endereço da rede, entre outros;
- **Atividade** – esse item descreve metas, tarefas, ações executadas pela entidade;
- **Relações** – esse item descreve o relacionamento dessa entidade com outras entidades, dentro e fora do domínio de rede, suas dependências, conexões com objetos, pessoas, serviços, entre outros.

Além dessas características descritas, o modelo de contexto (ilustrado na Figura 3) é enriquecido com outros aspectos importantes, tais como:

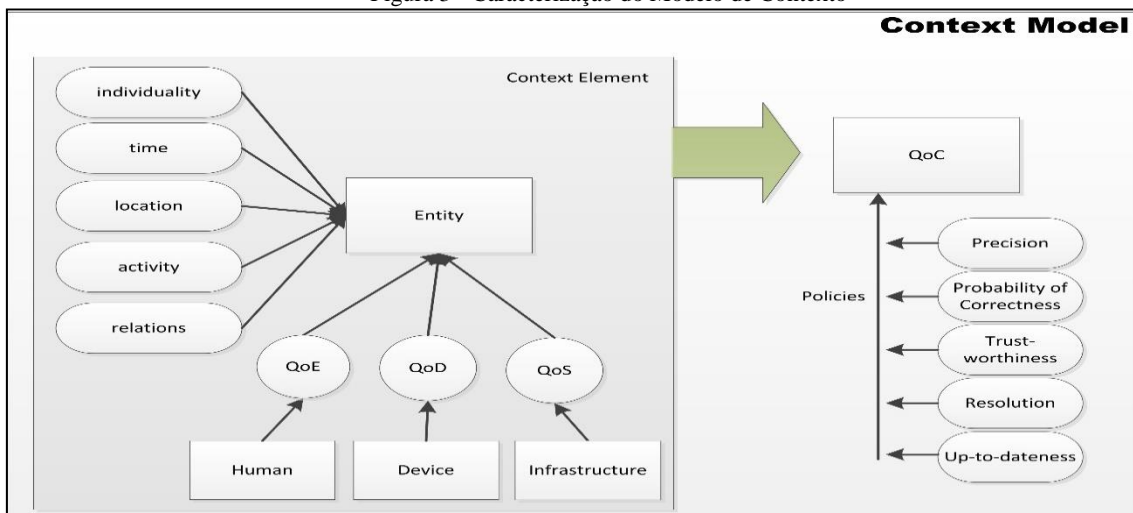
- **Qualidade de Experiência (QoE)** – esse aspecto descreve um grupo de parâmetros relacionados às experiências e perspectivas do usuário, como *Mean Opinion Score (MOS)*;
- **Qualidade de Dispositivo (QoD)** – esse aspecto descreve um grupo de parâmetros relacionados às características dos dispositivos tais como capacidade de armazenamento, poder computacional, nível de precisão dos dados coletados, entre outros;
- **Qualidade de Serviço (QoS)** – esse aspecto descreve um grupo de parâmetros relacionados às métricas, tanto qualitativas quanto quantitativas, considerando o acordo de nível de serviço (*Service Level Agreement - SLA*) entre o usuário e a plataforma, que contempla métricas tais como largura de banda, latência e variação do atraso.

As informações relacionadas às características e aspectos anteriormente citados são validadas baseadas em algumas métricas, as quais determinam a qualidade de contexto (QoC). Essas métricas são:

- **Precisão** – está relacionado ao nível de precisão das informações que permitem avaliar sua relevância;
- **Probabilidade de correção** – está relacionada à probabilidade de a informação estar correta;
- **Confiabilidade** – está relacionado ao nível de confiabilidade sobre a fonte da informação em questão;
- **Resolução** – está relacionado ao nível de granularidade de uma determinada informação;
- **Capacidade de Atualização** – está relacionado à forma pela qual aquela informação fornecida é atualizada.

É importante salientar que a qualidade de contexto não é uma característica ou mesmo um aspecto de informação relacionado à entidade e sim um conjunto de restrições que devem ser aplicadas às informações das entidades no que diz respeito às métricas descritas acima.

Figura 3 - Caracterização do Modelo de Contexto



### 3.2 CONTEXT-AWARE ADAPTIVE ROUTING FRAMEWORK (CAARF)

Após introduzir o modelo de contexto juntamente com seus aspectos e características, é possível apresentar um arcabouço que descreve tais funcionalidades de forma explícita mostrando a relação das entidades dentro de um domínio de rede com o seu contexto. Esse arcabouço, como

já citado, é o *Context-aware adaptive routing framework (CAARF)* que foi concebido com base na descrição de dois grupos funcionais principais:

- **Context management modules** – tem a responsabilidade de analisar e filtrar as informações contextuais;
- **Forwarding management modules** – tem a responsabilidade de processar as informações contextuais e aplicar as regras de encaminhamento na camada de encaminhamento da rede.

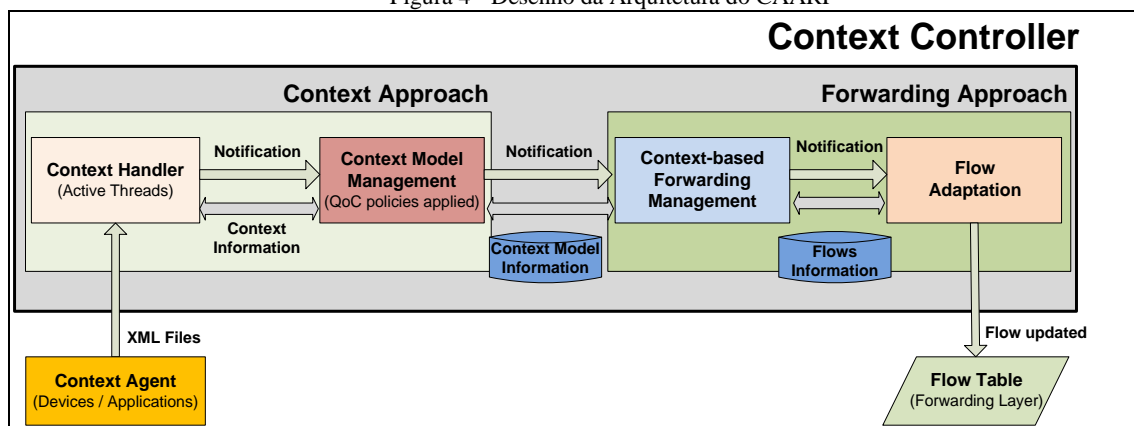
A nomenclatura dos módulos de gestão de contexto (*Context management module*) e de gestão de encaminhamento (*Forwarding management module*), assim como dos seus respectivos submódulos, foi definida na língua inglesa por questões de generalização e utilização nas publicações científicas. Por essa razão, esses termos são utilizados em inglês nesta dissertação.

Seguem algumas funcionalidades descritas por esse arcabouço:

- Coletar e compartilhar informações contextuais;
- Centralizar o armazenamento de informações contextuais;
- Realizar validação e avaliação da informação contextual baseada nas políticas definidas pela QoC (Qualidade de Contexto);
- Suportar consultas e utilização da informação contextual a fim de atualizar o encaminhamento para serviços sensíveis ao contexto

A partir das funcionalidades descritas anteriormente, uma arquitetura para a implementação do arcabouço CAARF foi concebida através da integração de diferentes funcionalidades e dividido em módulos, conforme descrito na Figura 4.

Figura 4 - Desenho da Arquitetura do CAARF



O primeiro grupo funcional apresentado na Figura 4 está relacionado aos módulos que coletam, armazenam e tratam as informações contextuais. Esses submódulos são os seguintes:

- ***Context Agent/Context Agent Middleware*** – responsável por receber as informações contextuais dos dispositivos de comunicação ativos, aplicações, usuários e suas relações. Essas informações são repassadas para o *Context Handler*;
- ***Context Handler*** – responsável por receber as informações contextuais do *Context Agent* e armazená-las no *Context Database*;
- ***Context Model Management*** – responsável por processar as regras de qualidade de contexto (QoC), definir o novo contexto de rede com as informações tratadas, e tornar essas informações disponíveis para serem acessadas pelo *Context-based Forwarding Management*.

O segundo grupo funcional apresentado na Figura 5 está relacionado aos módulos que analisam o contexto definido nos módulos de contexto e estabelecem ou não novos caminhos para os fluxos utilizarem em suas relações ativas na rede. Além de definir caminhos para os fluxos, estes podem ser aplicados às tabelas de fluxos da camada *Forwarding Layer*. Esses submódulos são os seguintes:

- ***Context-based Forwarding Management*** – responsável por processar as regras de encaminhamento baseadas em informações contextuais;
- ***Flow Adaptation*** – responsável pela notificação de informação de encaminhamento atualizada nos respectivos dispositivos de comunicação ativos como comutadores ou roteadores.
- ***Flow Table***—são as tabelas de fluxo da camada de encaminhamento (comutadores / roteadores).

Após uma visão geral do modelo de contexto e da arquitetura funcional do CAARF terem sido apresentados, neste capítulo é dada ênfase ao módulo de gestão de encaminhamento do CAARF.

### 3.3 GESTÃO DE ENCAMINHAMENTO DO CAARF: MODELAGEM E FORMALIZAÇÃO

No CAARF todos os seus componentes têm funções bem definidas, específicas e complementares de acordo com o funcionamento geral de todo o controlador de contexto. De

acordo com o que o arcabouço CAARF propõe, os módulos descritos, anteriormente desempenham tarefas bem distribuídas entre eles onde estes cooperam de forma efetiva para definir o novo contexto e fazer com que este reflita positivamente no tráfego de todos dispositivos, aplicações e usuários que usam determinada rede.

Conforme definido no CAARF, os módulos de funcionamento realizam as seguintes atividades:

- **Context Agent** – é o módulo no arcabouço que pode coletar dos dispositivos, dentro do domínio de funcionamento do controlador de contexto, informações de contexto como usuário, nome do host, domínio, se o dispositivo possui GPS, qual a precisão desse GPS, se tem o relógio de rede ativo (*Network timer protocol – NTP*) ativo, qual o fuso-horário do dispositivo, MOS da aplicação, ruído do ambiente, largura de banda, atraso, variação do atraso, taxa de erros, entre outras informações que o administrador de rede considere relevante para mudar o contexto de rede. Cada dispositivo da rede, seja dispositivo de encaminhamento ou não, possui um *context agent* fazendo tal coleta de informações e fornecendo dados para serem avaliados no controlador de contexto da rede;
- **Context Handler** – é o módulo no arcabouço que faz o processamento de várias requisições, ao mesmo tempo, de todos os XMLs de dispositivos, suas relações com outros dispositivos, bem como informações contextuais que serão inseridas em um repositório e estas informações serão tratadas em um módulo que analisa o contexto envolvendo dados de vários dispositivos ao mesmo tempo;
- **Context Model Management** – é o módulo no arcabouço que recupera dados registrados no repositório de contexto e os filtra com base em regras de contexto (QoC), definidas pelo administrador da rede que estabelecem a qualidade requerida pelo perfil de tráfego da rede específica;
- **Context-based Forwarding Management** – é o módulo no arcabouço que, baseado em um novo contexto definido no módulo *Context Model Management*, processa essas informações, filtra dados baseados em regras de encaminhamento e define novos fluxos de encaminhamento que serão aplicados posteriormente;
- **Flow Adaptation** – é o módulo no arcabouço que identifica os novos caminhos para os fluxos e determina as novas portas para comutação para eles e os aplica nos comutadores/roteadores da rede;
- **Flow Table** – é o módulo no arcabouço que recebe os novos fluxos e implementa novos caminhos para o tráfego.



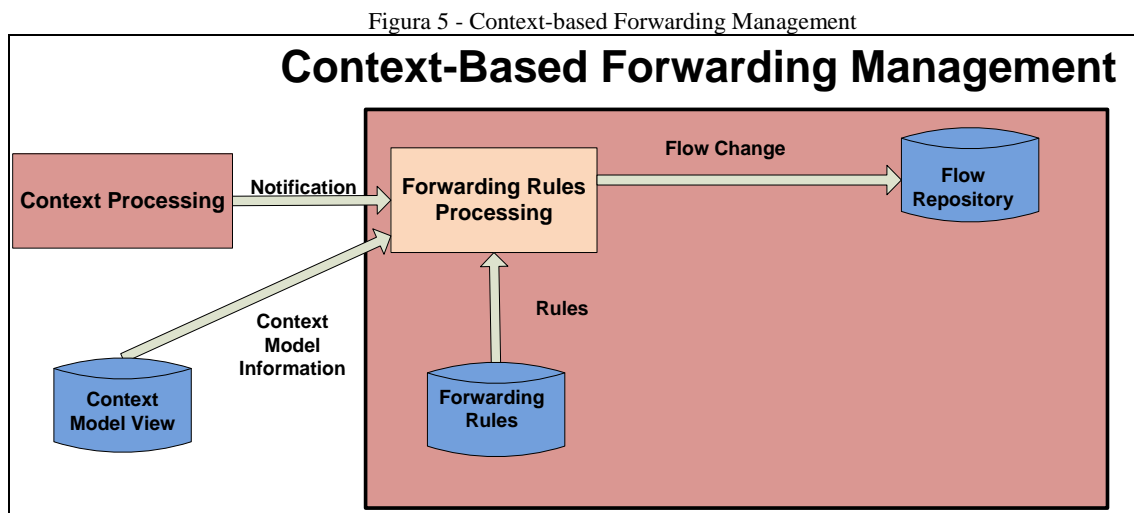
De forma a esclarecer os principais mecanismos de funcionamento dos módulos funcionais do CAARF, passamos a apresentar o seu módulo de encaminhamento, as regras de encaminhamento e os eventos relacionados ao encaminhamento.

### 3.3.1.1 Módulos de Encaminhamento

Os módulos de encaminhamento dentro do arcabouço CAARF estão estruturados da seguinte maneira:

#### Context-based Forwarding Management

É dividido em quatro submódulos, conforme Figura 5:



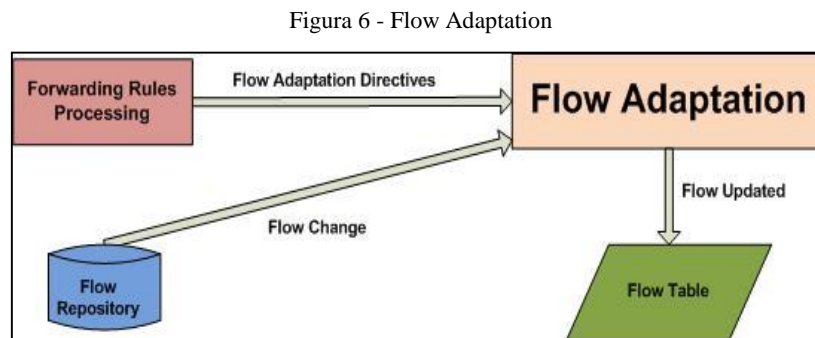
São eles:

- **Forwarding Rules Processing:** responsável por processar as regras de encaminhamento, pré-definidas pelo administrador de rede, no tratamento dos fluxos. Recebe a notificação do submódulo *Context processing* e consulta as informações no *Context Model View Database*;
  - ✓ Após criação das novas regras, compara com as regras atualmente em uso nos dispositivos e avalia se o índice de modificação foi maior que o mínimo pré-determinado para atualização dos dispositivos, se sim, autoriza transmissão;
- **Forwarding rules:** responsável por armazenar as regras de encaminhamento (ANEXO A) que serão analisadas para a criação ou não de novos fluxos nos comutadores;

- **Flow repository:** responsável por armazenar os fluxos que serão encaminhados pelo sub-módulo *Flow Adaptation* para atualização das tabelas de fluxos dos comutadores. Armazena não só regras em uso nos dispositivos da rede, como também as novas regras criadas, mesmo que não sejam consideradas aptas para aplicação no novo cenário da rede.

### **Flow Adaptation**

É responsável por receber as notificações de novos caminhos a serem aplicados para os fluxos ou mesmo atualizações de caminhos já existentes feitas pelo *Context-based Forwarding Manager* e, baseado no *Flow Repository*, atualizar as tabelas de fluxos dos comutadores. O *Flow Adaptation* é ilustrado na Figura 6:



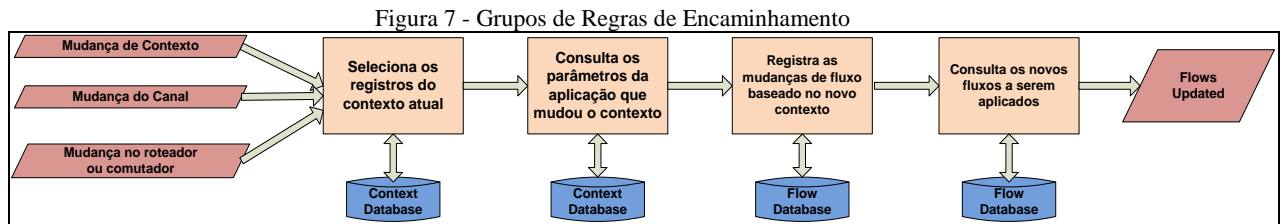
#### **3.3.1.2 Regras de encaminhamento**

No CAARF, dentro da abordagem de encaminhamento, existem algumas regras, definidas previamente pelo administrador de rede, que são usadas para orientar as escolhas dos novos caminhos, visando a melhor qualidade possível para uma dada comunicação. Essas regras estão associadas a várias situações que devem ser analisadas e que podem ser relacionadas aos próprios caminhos registrados dentro do repositório do controlador de contexto, os roteadores previamente cadastrados que são utilizados na rede ou ao contexto previamente tratado na abordagem de contexto. O foco principal na aplicação dessas regras é possibilitar uma coexistência de qualidade entre as aplicações na rede sem prejudicar umas às outras sem afetar sua qualidade e permitindo uma melhor experiência para o usuário final da rede.

Basicamente, as regras de encaminhamento podem ser agrupadas em (Uma relação mais detalhada das regras de encaminhamento definidas nesta dissertação, podem ser encontradas no ANEXO A):

- **Regras baseadas em mudança de contexto:** essas regras estão relacionadas às mudanças de contexto já processadas e disponíveis para serem analisadas para a mudança do encaminhamento. Com base no novo contexto, são avaliados todos os caminhos e definidos quais os caminhos principais ou alternativos que ofereçam uma melhor qualidade para uma dada relação que foi tratada na base de contexto. Seguem alguns exemplos que estão relacionados com esse grupo de regras:
  - Mudança de contexto por mudança do MOS de uma relação usando uma aplicação VoIP, e;
  - Mudança de contexto por conta de um alto ruído na comunicação de uma relação usando uma aplicação VoIP.
- **Regras baseadas na mudança de conexão:** essas regras estão relacionadas às condições de tráfego nas conexões nas quais estão sendo avaliados os caminhos. Por exemplo, é possível que em um dado momento uma determinada conexão esteja sem conectividade, oferecendo um atraso elevado ou então com uma intermitência que prejudique o uso de determinadas aplicações. Seguem alguns exemplos que estão relacionados com esse grupo de regras:
  - Indisponibilidade da conexão;
  - Degradação da conexão relacionada à: diminuição da largura de banda, aumento do atraso, aumento na taxa de erros ou aumento do *jitter*;
  - Identificação de uma sobrecarga de tráfegos na conexão.
- **Regras baseadas na mudança da camada encaminhamento:** essas regras estão relacionadas às condições nas quais os equipamentos da camada de encaminhamento estão no momento em que estão sendo avaliados os caminhos. Por exemplo, é possível que em um dado momento um determinado roteador esteja sem conectividade ou mesmo que os parâmetros coletados referentes ao atraso, largura de banda ou mesmo variação do atraso estejam aquém da necessidade de determinada aplicação. Além disso, pode ser que, por análise do módulo *Context-based Forwarding Management*, este roteador esteja sobrecarregado e que isso possa comprometer não somente aos fluxos que nele estejam trafegando, mas também os novos fluxos que venham a ser escolhidos para trafegarem nele. Seguem alguns exemplos relacionados com esse grupo de regras:
  - Indisponibilidade do roteador;
  - Sobrecarga de processamento no roteador;
  - Falha de hardware no roteador, e;
  - *Timeout* na comunicação com o roteador.

A Figura 7 ilustra o tratamento que é dado à informação de contexto quando as informações de encaminhamento estão sendo analisadas para a escolha de novos caminhos.



A Figura 8 apresenta como são analisadas as regras de encaminhamento no controlador de contexto. No começo da ilustração são indicadas por quais circunstâncias ocorrem a validação das regras de encaminhamento, seja por mudança de contexto, seja por mudança do canal ou mesmo por mudança no roteador ou no comutador. De acordo com o fluxo acima, a etapa inicial da validação das que é realizada é a seleção dos registros no *Context Model View*, que representam o atual contexto da rede. Com essas informações recuperadas da base de dados, para cada contexto de cada relação é identificada a aplicação que está sendo utilizada e, a partir dela, são recuperados os parâmetros padrão destas aplicações.

Com os parâmetros padrão recuperados, são obtidas as opções de caminho para esse contexto e, assim, dentre várias opções, é escolhida a que melhor adapte o tráfego do novo contexto a um caminho com melhor qualidade, em comparação com o caminho anteriormente usado por essa relação.

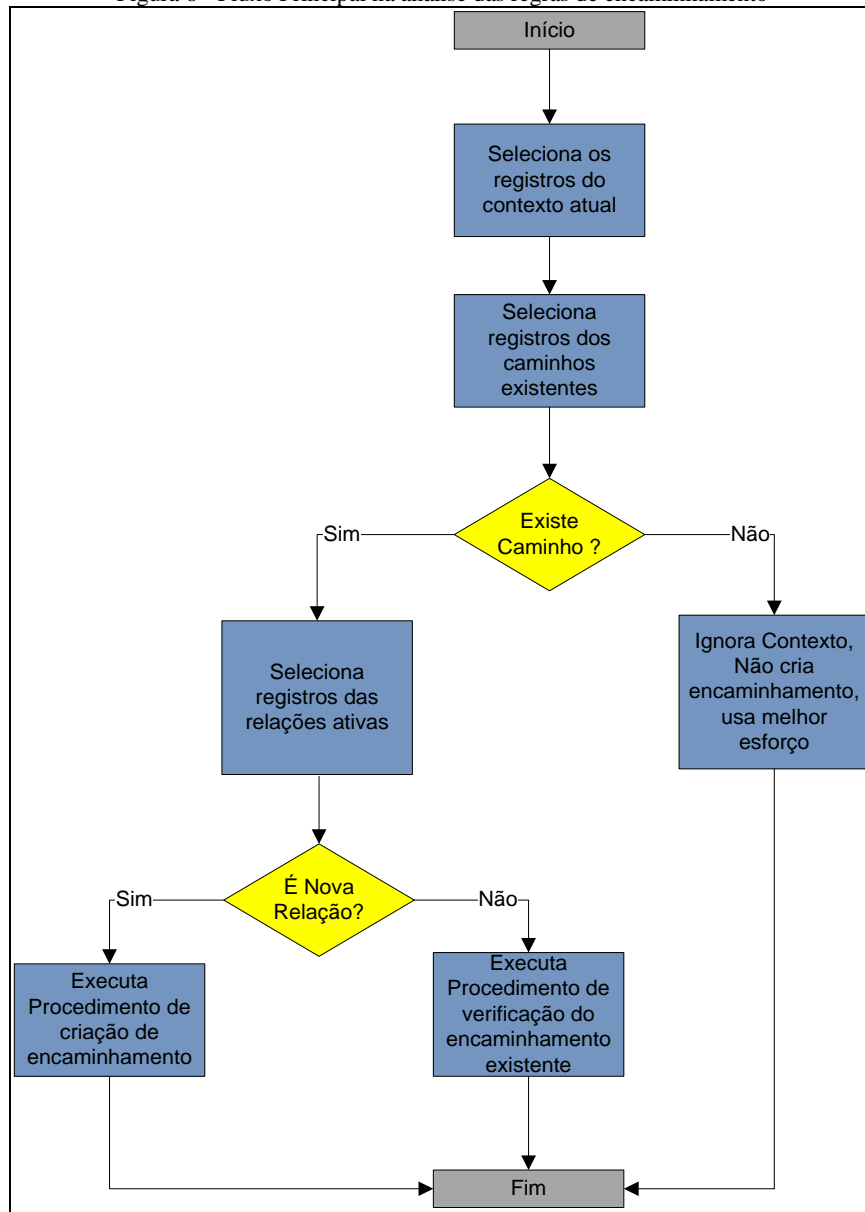
Com as escolhas de caminho realizadas para todos os novos contextos da rede, pode ser realizada a etapa final que é a identificação dos novos caminhos de fluxo a serem aplicados e assim aplicá-los na camada de encaminhamento a fim de melhorar o tráfego das relações afetadas pela mudança de contexto naquele instante.

A Figura 8 apresenta um fluxograma com o fluxo principal de verificação das regras de encaminhamento quando executado no submódulo *Forwarding Processing Rules*. Importante salientar que, durante a análise e verificação das regras de encaminhamento, são feitas checagens iniciais como: se existe caminho ativo, disponível e sem intermitência na comunicação; se o caminho não está sobrecarregado com muitos tráfegos que consomem boa parte da largura de banda, e; por fim são analisadas as questões relacionadas aos contextos dos usuários e das aplicações que estão sendo usadas e seus requisitos mínimos para um bom funcionamento, sem prejudicar os outros contextos.

Conforme o fluxo apresentado na Figura 8, independente, do tipo de mudança no contexto que tenha ocorrido, a etapa inicial é a seleção do contexto atual conforme análise feita na abordagem de contexto. Após selecionar o contexto atual, é feita a seleção dos caminhos possíveis para cada um dos tráfegos que serão afetados. Caso não haja caminho selecionado para determinado contexto, esse tráfego será ignorado e assim será utilizado para este o melhor esforço (LEINER et al., 1997).

Caso haja algum caminho para tal tráfego, é feita uma outra verificação das relações ativas. Nessa fase, é visto se a relação é nova dentro do controlador do contexto. Caso seja uma nova relação, é registrada a criação de um novo caminho, conforme análise que é feita pelo *Context-based Forwarding Management*, no banco do controlador de contexto e a posterior adaptação do novo caminho no roteador ou comutador determinado depois da análise. Caso não seja uma relação nova, é feita uma verificação se o caminho existente está adequado ao novo contexto ou, caso contrário, é feita uma verificação de outros caminhos e assim é feita a escolha mais adequada do novo contexto em questão.

Figura 8 - Fluxo Principal na análise das regras de encaminhamento



### 3.3.1.1 Eventos

Na abordagem de encaminhamento, podem ser identificados alguns eventos que ocorrem quando da realização das atividades dos módulos operacionais do controlador de contexto, associados à essa abordagem.

Seguem alguns eventos identificados relacionados aos módulos da abordagem de encaminhamento (Uma relação mais detalhada dos eventos definidos nesta dissertação, podem ser encontradas no ANEXO B). Podemos dividi-los em eventos de:

- **Notificação:** Esses eventos estão relacionados com as notificações recebidas e/ou enviadas de/por módulos do arcabouço, por exemplo:
  - Novo contexto disponível para ser analisado;
  - Novos fluxos a serem selecionados para aplicação pelo *Flow Adaptation*;
- **Persistência:** Esses eventos estão relacionados com a interação dos módulos com a camada de persistência do controlador de contexto, como por exemplo:
  - Consulta ao novo contexto dentro do *Context Database*;
  - Salva de novas regras de encaminhamento no *Context Database*;
  - Consulta às regras de encaminhamento dentro do *Context Database*;
  - Consulta às opções de caminho registradas dentro do *Context Database*;
- **Regra:** Esses eventos estão relacionados com as regras de encaminhamento registradas no *Context Database* e utilizadas pelo módulo *Context Forwarding-based Management* para tomada de decisão dos novos caminhos dos novos contextos identificados na rede. Temos como exemplo desses eventos:
  - Processamento da análise das regras de encaminhamento pré-definidas;
  - Registro de novas regras de fluxo por parte do administrador de rede.
- **Fluxo:** Esses eventos estão relacionados aos novos fluxos que são identificados após a aplicação das regras de encaminhamento e análise do contexto de rede. Como exemplo tem-se:
  - Aplicação dos novos fluxos na camada de encaminhamento;
  - Verificação dos fluxos existentes para identificar os fluxos que podem ser invalidados de acordo com o atual contexto.

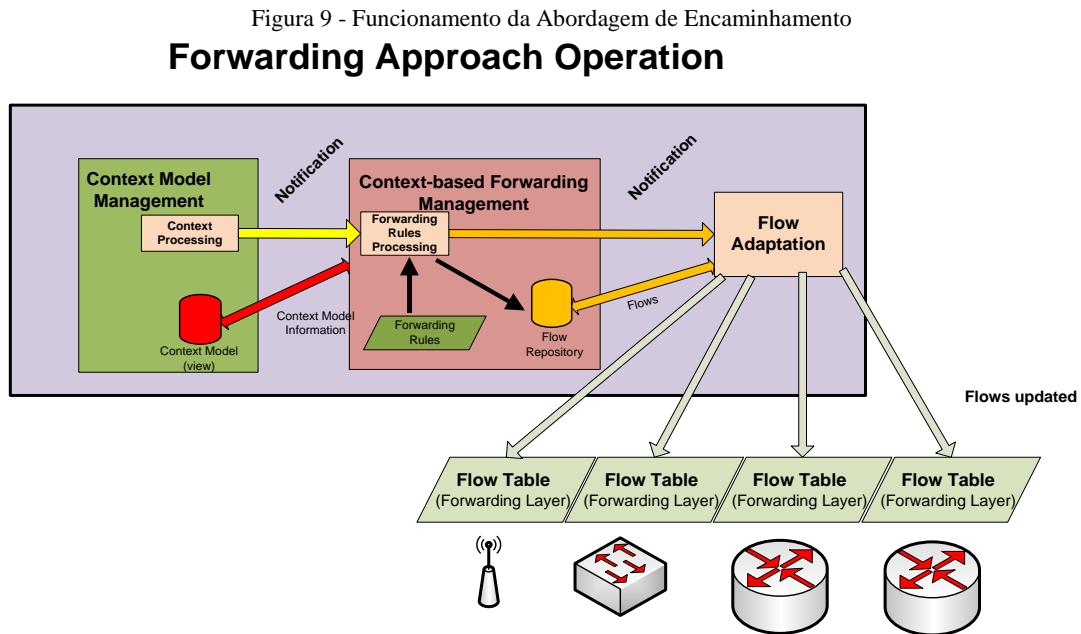
Os aspectos operacionais do arcabouço CAARF são apresentados na próxima seção.

### 3.4 CAARF: ABORDAGEM OPERACIONAL

O arcabouço de encaminhamento adaptativo de tráfego possui duas abordagens bem definidas: a abordagem de contexto a qual filtra, processa e define o novo contexto e a abordagem de encaminhamento a qual filtra, define e aplica novos fluxos baseados em contexto. Ambas possuem características bem definidas e complementares visando a identificação do novo contexto com base nas informações contextuais coletadas do ambiente em que o controlador está em execução.

Dentro da abordagem de encaminhamento ocorre a definição da gestão de caminhos, ou seja, a gestão da mudança dos fluxos dos tráfegos da rede, baseados nos novos contextos identificados pelo controlador de contexto, que podem ou não, conforme o contexto, serem modificadas visando a melhoria da utilização de determinada aplicação.

A Figura 9 ilustra como ocorre o funcionamento dentro da abordagem de encaminhamento.



Conforme ilustrado na Figura 9, a abordagem de encaminhamento envolve a interação entre 3 módulos dentro do *Context Controller*: o *Context Model Management*, o *Context-based Forwarding Management* e o *Flow Adaptation*.

Podemos definir em diferentes fases como ocorre a gestão de encaminhamento dentro do *Context Controller*:

- **Fase 1** – É recebida uma notificação, por parte do *Context Model Management*, mais especificamente, executado pelo submódulo *Context Processing*, sobre a ocorrência de novos contextos para o *Context-based Forwarding Management* submeter à sua análise;
- **Fase 2** - Em seguida, dentro do módulo *Context-based Forwarding Management*, o submódulo *Forwarding Processing Rules* consulta dados do novo contexto dentro do *Context Database*;
- **Fase 3** – Com as informações do novo contexto das relações já selecionadas, é feita uma consulta na base de encaminhamento para obter todas as opções de caminho possíveis a serem analisadas;



- **Fase 4** – Com os caminhos selecionados é feita uma verificação geral destes, baseado nas regras de encaminhamento definidas pelo administrador de rede. A partir disso e após todas as informações selecionadas, é feita uma análise dos caminhos que já passaram pela filtragem das regras, e é verificada a melhor opção de caminho dentre as várias opções definidas na base de encaminhamento;
- **Fase 5** – Com a escolha definida, o submódulo *Forwarding Processing Rules* notifica o módulo *Flow Adaptation* que existem novos fluxos a serem aplicados na camada de encaminhamento;
- **Fase 6** – Nessa fase, o *Flow Adaptation* consulta os novos fluxos a serem aplicados na camada de encaminhamento e os seleciona para executar a última fase do processo de escolha de novos caminhos para o novo contexto de rede, e;
- **Fase 7** – A última fase é quando ocorre a efetiva aplicação dos novos caminhos na camada de encaminhamento, conforme escolha feita no módulo *Context Forwarding-based Management*.

Com esse funcionamento é possível que determinado fluxo degradado, que esteja sendo usado por determinada comunicação origem e destino, seja modificado dinamicamente sem haver prejuízo do tráfego, tendo sim melhoria dessa comunicação a partir de determinado momento que o novo contexto for analisado e aplicado na camada de encaminhamento.

### 3.5 CONCLUSÃO

O arcabouço definido pelo CAARF define, portanto, uma forma de realizar o roteamento adaptativo de tráfego com base em informações contextuais, dando importância a dados relacionados à experiência do usuário. Dessa forma, é possível que uma conexão que o usuário indique que esteja com baixa qualidade, possa ser analisada pelo controlador de contexto baseado no CAARF e processe a mudança de caminho e que possibilite uma melhoria dessa conexão no momento posterior, permitindo ganho de qualidade na comunicação para o usuário final.

Com a abordagem trazida pelo CAARF, além de permitir uma melhor qualidade nas comunicações na rede possibilitará um uso mais otimizado do canal (ou conexão) dentro da rede. Será possível, por exemplo, minimizar o congestionamento das rotas e, também, impossibilitar que dentro do controlador de contexto novas rotas sejam direcionadas por determinado canal somente por este possuir uma grande capacidade de vazão ou mesmo que

apresente outros dados com melhor qualidade de serviço. A princípio, em outras abordagens, esse canal seria mais utilizado, pelos seus requisitos, possibilitando em pouco tempo o congestionamento de um canal de qualidade, sem distribuir tal carga para outras alternativas na rede.

## **4 SIMULAÇÃO DA GESTÃO DE ENCAMINHAMENTO**

Nesse capítulo são apresentados os detalhes que envolvem as estratégias utilizadas para a simulação dos aspectos relacionados à gestão de encaminhamento do arcabouço CAARF.

São apresentados os requisitos funcionais e não funcionais que envolvem a simulação da gestão de encaminhamento, bem como apresentar detalhes do modelo arquitetural utilizado nesse trabalho.

A linguagem de programação utilizada na simulação é discutida, bem como a modelagem de persistência realizada, seja em linguagem de marcação ou utilizando banco de dados. Com relação aos dados armazenados em banco de dados, a definição das tabelas que armazenam os dados, também, são descritas nesse capítulo, bem como os relacionamentos entre as tabelas utilizadas para expor os dados dentro do controlador de contexto e, mais especificamente, dentro da abordagem de encaminhamento.

### **4.1 REQUISITOS FUNCIONAIS E NÃO FUNCIONAIS**

Os requisitos de software, conforme definido por (SOMMERVILLE, 2008), permitem descrever os serviços que devem ser fornecidos pelo sistema e as suas restrições operacionais que influenciam o seu funcionamento.

Todo requisito de um sistema é descrito, conforme (PFLEEGER, 2004), como uma característica do sistema ou a descrição de algo que a aplicação é capaz de realizar para atingir os seus objetivos. Já em (ROBERTSON; ROBERTSON, 2006) os requisitos são definidos como algo que o produto tem de fazer ou uma qualidade que ele precisa apresentar.

Baseado nas definições apresentadas, pode-se dizer que os requisitos de um sistema incluem as especificações dos serviços que o sistema tem de oferecer, restrições de operação e até restrições inerentes ao seu processo de desenvolvimento. Dentre os diferentes tipos de requisitos dentro da engenharia de software, os requisitos de um sistema, normalmente, são caracterizados como funcionais, que seriam associados às funções do sistema, ou não funcionais, que seriam associados a aspectos não relacionados às funções do sistema, como por exemplo, desempenho, escalabilidade, segurança, portabilidade, entre outros (SOMMERVILLE, 2008).

Como requisitos funcionais relacionados à gestão de encaminhamento podem ser listados os seguintes tópicos:

- O módulo de gestão de encaminhamento deve receber as notificações dos dados de contexto recuperados, os quais já devem ter sido filtrados pelas regras de QoC e assim não devem possuir quaisquer inconsistências ou incompletudes para o processamento das escolhas dos novos caminhos;
- Para todo novo contexto recebido, é necessário realizar a verificação das regras de encaminhamento, seja para relações novas ou relações existentes;
- Deve permitir o cadastramento das informações de encaminhamento pelo administrador de rede, bem como as regras de encaminhamento que irão ser utilizadas durante a análise dos caminhos para a escolha do caminho mais adequado para o fluxo, que teve mudança de contexto;
- Deve realizar a verificação das conexões utilizadas, antes de qualquer análise de caminho. Em caso de alguma conexão estar inativa ou mesmo foi detectada com problema, todos os caminhos que utilizam esta conexão não devem ser selecionados para a escolha de um novo caminho, e;
- Deve permitir a aplicação (configuração) dos novos caminhos selecionados para determinadas relações, através do submódulo *Flow Adaptation*, aplicando os novos caminhos na camada de encaminhamento sejam em roteadores ou mesmo em comutadores.

Em relação aos requisitos não funcionais podem ser listados:

- Os módulos de encaminhamento devem ser construídos independente da tecnologia de rede que está funcionando;
- O submódulo *Flow Adaptation* deve estar capacitado a interagir funcionalmente, aplicando os novos caminhos, através de múltiplas tecnologias de encaminhamento;
- O desempenho das funções realizadas pelo módulo de encaminhamento deve ser otimizado, visando evitar contenções dentro do controlador e procurar refletir com mais rapidez a realidade do novo contexto de rede na camada de encaminhamento;
- O controlador deve ter uma estrutura escalável que permita o atendimento, minimizando ao máximo os prejuízos ao tempo de resposta do controlador de contexto, de um grande número de requisições de análise de contexto em uma rede, sem que haja necessidade de ampliações constantes da infraestrutura que suporta o controlador de contexto, e;

- A estrutura desenhada do controlador de contexto deve ser flexível no atendimento a novas implementações sem que haja a necessidade de mudanças radicais na estrutura de processamento do controlador e sem prejudicar o desempenho do controlador após as mudanças.

Após apresentar os requisitos funcionais e não funcionais associados à simulação em questão, neste capítulo é apresentada a forma que foi modelada a simulação.

#### 4.2 MODELAGEM DA SIMULAÇÃO

Para apresentar o funcionamento da gestão de encaminhamento do controlador de contexto foi realizada uma simulação com dados obtidos através do simulador Network Simulator Version 2 (NS-2) (NS-2, 2015), usando scripts de simulação TCL (NS-2, 2015). Além disso, são utilizadas consultas utilizando linguagem SQL fazendo acesso a um banco de dados relacional para obtenção dos novos caminhos, conforme análise feita das informações de encaminhamento dentro do *context database*, cujo modelo de dados está descrito na seção 4.4.

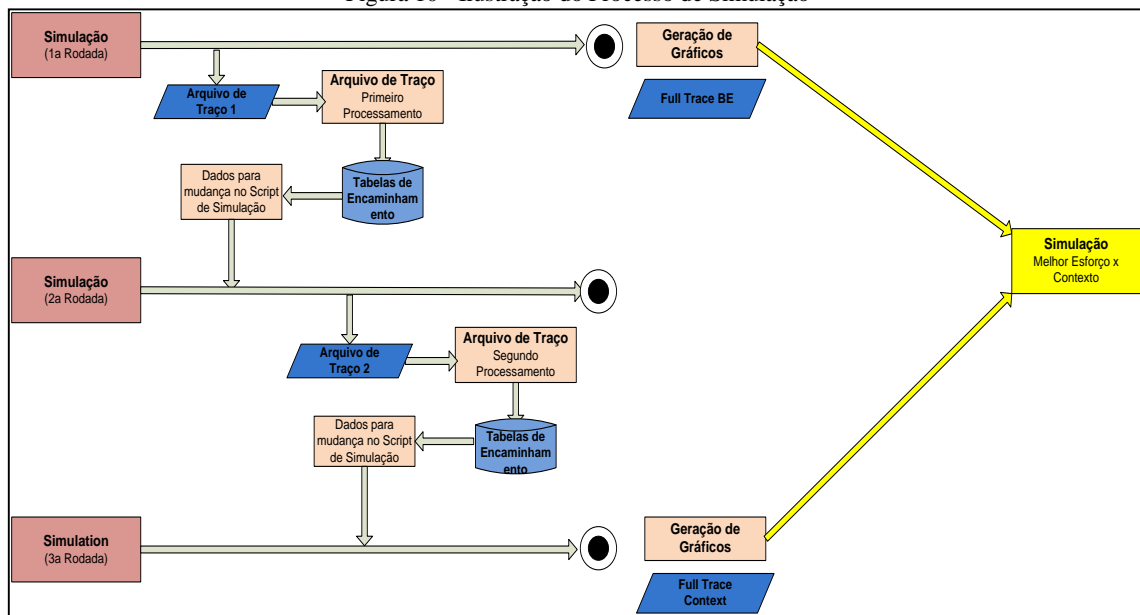
O uso da linguagem SQL juntamente com o banco de dados relacional proporcionou, no processo de simulação, um padrão de entrada de dados, com o cadastramento de todas as opções de encaminhamento possíveis já cadastradas na base de dados, bem como uma sistematização na saída de dados, através do registro dos novos caminhos associados aos novos contextos processados dentro do CAARF.

Outro aspecto positivo do uso do banco de dados para a simulação foi a rápida adequação das mudanças de contexto, obtidas com as consultas no banco de dados, para adaptação das rodadas da simulação. Além disso, os dados cadastrados em banco de dados podem ser formatados através de comandos SQL e podem ser apresentados de formas diferentes de acordo com a necessidade de quem consulta os dados.

Portanto, o uso do banco de dados proporcionou uma flexibilidade na apresentação dos dados para análise, diferentemente da apresentação dos dados em arquivos texto.

A Figura 10 ilustra de forma completa a simulação e as suas etapas.

Figura 10 - Ilustração do Processo de Simulação

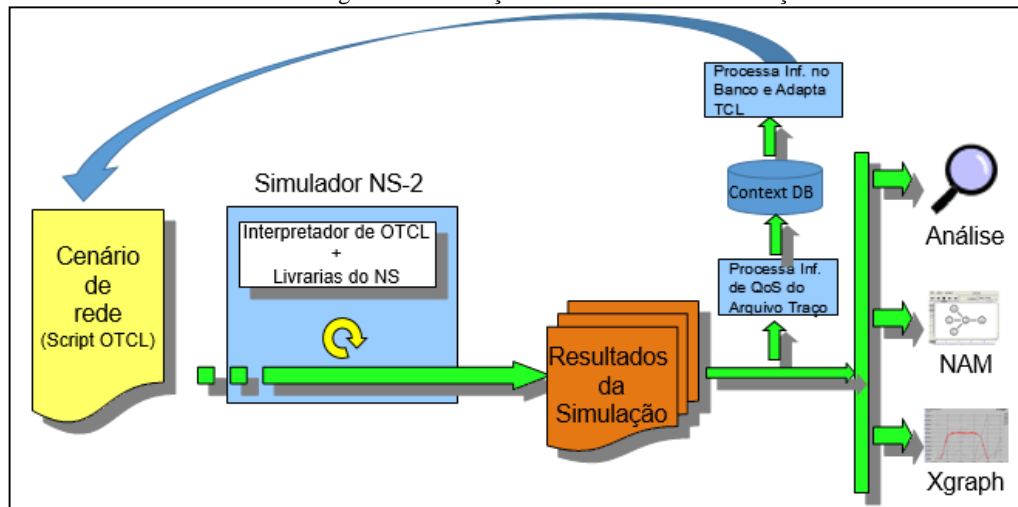


O processo de simulação utilizado é realizado em etapas de execução do script de simulação e obtenção dos arquivos de traço para processamento das informações, entrada de dados em banco e, por fim, em cada rodada são informadas as mudanças necessárias no script de simulação para serem processadas novamente. Portanto, essas etapas são processadas em 3 ciclos até a obtenção do resultado final com a mudança de contexto obtida para determinadas relações.

As etapas do processo de simulação são descritas com mais detalhes:

- **Simulação e geração de arquivo de traço:** Execução da simulação em um intervalo de tempo e geração do arquivo de traço para ter os dados inseridos no banco de dados;
- **Processamento do arquivo de traço e inclusão no banco de dados:** Processamento dos arquivos de traço, com informações do contexto da simulação em determinado intervalo, e inclusão dos dados de parâmetros de rede a serem analisados pela próxima etapa;
- **Análise dos dados incluídos e geração de novas mudanças no script TCL:** A partir dos dados de simulação obtidos e inseridos no banco de dados, é feita a análise do contexto e, caso seja adequada, é processada a mudança de caminho. Com os novos caminhos obtidos, que refletem a mudança de contexto, é feita a adaptação do script TCL para que seja usado na próxima rodada da simulação.

Figura 11 - Ilustração de cada rodada da Simulação



Durante a apresentação dos cenários do estudo de caso, essas etapas serão executadas em algumas rodadas para apresentarem as opções de mudança de contexto que podem ocorrer em uma rede, conforme representado na Figura 11.

Com os resultados iniciais e os finais, é feita uma comparação de gráficos de diferentes parâmetros demonstrando as melhorias obtidas nas etapas do mesmo cenário de simulação, apresentando assim as mudanças de contexto do início ao fim do processo de simulação.

Após a apresentação do modelo de simulação, será apresentada na próxima seção a modelagem de persistência utilizada pela simulação e suas características.

#### 4.3 MODELAGEM DA PERSISTÊNCIA

Para o CAARF, a modelagem de persistência foi dividida em duas partes: a persistência em relação à coleta dos dados para serem analisados pelo controlador de contexto (CAARF) e a persistência relacionada aos dados armazenados pelo controlador de contexto para o seu funcionamento.

Para a coleta de dados feito pelos *Context Agent* distribuídos na rede nos dispositivos que irão se comunicar e nos dispositivos de encaminhamento, que tem sua coleta feita pelo *Context Agent Middleware*, foi definida uma estrutura de marcação ilustrada no capítulo anterior de forma conceitual pelo modelo de contexto, através do elemento de contexto.

As informações definidas nesta dissertação são: individualidade, localização, relação, atividade, tempo e dados relacionados a Qualidade de Serviço (como atraso, *jitter* e largura de banda), Qualidade de Dispositivo (como capacidade de processamento de um roteador ou comutador

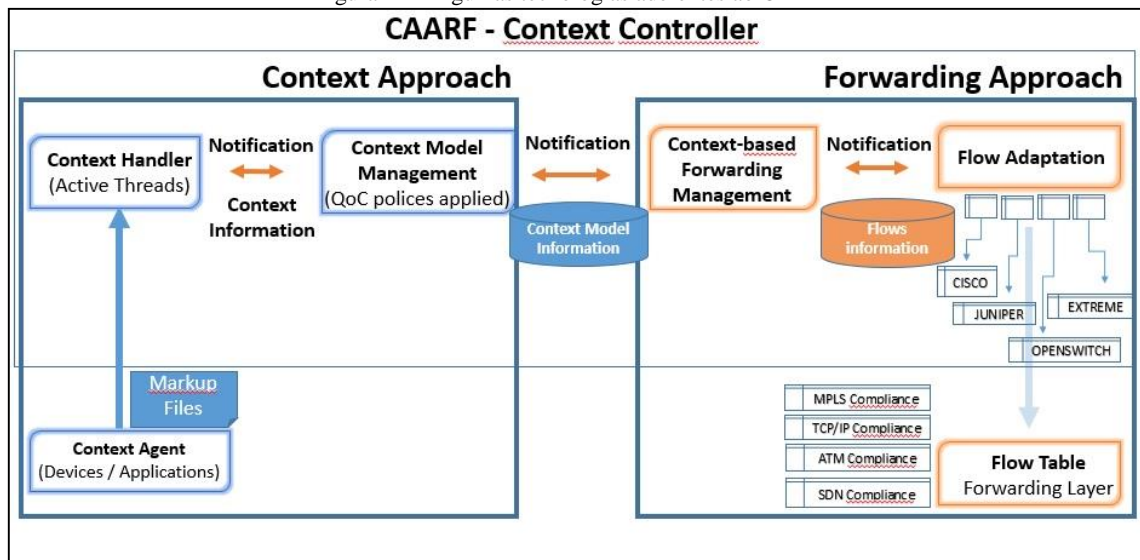
ou mesmo a identificação da existência de GPS em um determinado dispositivo que irá se comunicar na rede) e, Qualidade de Experiência, que seria a impressão do usuário sobre o uso de determinada aplicação, como o MOS (*Mean Opinion Score*).

A estrutura de marcação usada é o XML (*Extensible Markup Language*) (XML, 2015), cuja manipulação é muito utilizada no universo da programação e muito utilizada, também, como linguagem de integração entre softwares e ferramentas. Pela sua fácil manipulação e o extenso número de ferramentas que entendem seu formato, o XML foi uma escolha adequada para nosso objetivo de apresentar as características dos aspectos da gestão de encaminhamento do CAARF.

Portanto, conforme descrito, segue no ANEXO B alguns exemplos de XML de coleta de dados que são gerados pelos *Context Agent* distribuídos na rede. Esses XML irão refletir o contexto da rede naquele momento em que os dados estão sendo coletados.

Dentro do controlador de contexto, foi definida uma segunda estrutura de persistência já sido citada acima. Para todas as informações tratadas dentro do controlador, existe uma estrutura de tabelas normalizadas e estruturadas com uma certa flexibilidade, que permitem a construção de um controlador de contexto adaptável a uma série de tecnologias de rede, sem especificamente ficar aderida a uma arquitetura, conforme ilustra a Figura 12.

Figura 12 - Algumas tecnologias aderentes ao CAARF



As tabelas definidas no *Context Database*, apresentadas em detalhe no ANEXO D, estão definidas da seguinte maneira:

**Relation:** Essa tabela armazena todos os dados coletados pelo *Context Agent* / *Context Agent Middleware* e que são enviados, posteriormente, para o *Context Handler* no qual os insere,



inicialmente, e cujos dados são tratados pelo *Context Model Management* para verificação das regras de contexto e depois para verificação do contexto.

Cada linha dessa tabela contém somente informações de uma única relação de um dispositivo que teve seus dados coletados. Sendo assim, apesar de os dados coletados e inseridos no XML padrão do contexto poderem conter mais de uma relação, cuja representação é a comunicação do dispositivo da coleta com outro dispositivo através de determinada aplicação, no processo de leitura e tratamento do XML, o *Context Handler* irá separar essas informações na tabela como informações distintas, pois o são, já que cada comunicação dentro do *Context Controller* deve ser tratada de forma única e exclusiva.

Sendo assim, todo contexto identificado, para determinada relação, dentro da arquitetura de contexto deve ser tratado de forma exclusiva sem deixar, porém, de se preocupar com todas as outras comunicações que ocorrem na rede;

**Device**: Esta tabela armazena dados sobre todos os dispositivos da rede que irão estabelecer comunicação ou mesmo que irão encaminhar os pacotes de tráfego na rede. Sendo assim, todo roteador ou comutador na rede terão seus dados cadastrados nessa tabela, bem como os dados dos dispositivos como *Smartphones, tablets, notebooks, desktops*, entre outros dispositivos que desejem se comunicar.

Para diferenciar dispositivos que somente se comunicam com os que fazem o roteamento ou comutação dos pacotes, existe um campo de status dentro da tabela que os diferencia e permite que no *Context-based Forwarding Management* possa identificar os elementos da camada de encaminhamento que serão analisados para a escolha dos novos caminhos em caso de novos contextos identificados;

**Parameter**: Essa tabela armazena dados de parâmetros que serão utilizados pela tabela *Individuality* na estrutura do *Context Database*. Pela estrutura do modelo de dados do *Context Database*, essa tabela é extremamente importante, pois é ela que permite dar uma maior flexibilidade ao Arcabouço para adequação das várias tecnologias de rede que existem na rede.

Por exemplo, se por conta do arcabouço tiver que suportar uma nova tecnologia de rede, haverá a necessidade de que a tabela de dispositivo aceite uma nova informação, o trabalho do administrador de rede em adequar o controlador a essa mudança seria somente de inserir mais um parâmetro nessa tabela e associar esse parâmetro a tabela *Individuality* de forma dinâmica e sem a necessidade de grandes mudanças dentro do controlador.

Em uma estrutura que precise adaptar o modelo de dados a novas tecnologias com a adição de novos campos em uma tabela existente, isso poderia causar uma indisponibilidade e não proporcionaria tal flexibilidade ao arcabouço em mudanças e adaptações que sejam necessárias.

**Parameter dev**: Essa tabela armazena dados de parâmetros que serão utilizados pela tabela *Device* na estrutura do *Context Database*. Da mesma maneira que a *Parameter* permite uma maior flexibilidade para a tabela *Individuality*, esta tabela tem a mesma função.

Por exemplo, se houver a necessidade de que haja uma mudança no XML acerca da inclusão de mais informações de coleta para os dispositivos, o novo parâmetro para o dispositivo pode ser inserido nessa tabela e, posteriormente, associado a todos os dispositivos essa nova informação sem necessidade de parada do controlador para criação de um novo campo na tabela, a fim de atender essa nova necessidade.

**Parameter link**: Essa tabela armazena dados de parâmetros que serão utilizados pela tabela *Link* na estrutura do *Context Database*. Da mesma maneira que a *Parameter* permite uma maior flexibilidade para a tabela *Individuality*, esta tabela tem a mesma função.

Por exemplo, se houver a necessidade de uma mudança no controlador de contexto para a análise de mais parâmetros de rede para uma conexão entre roteadores ou comutadores, é necessário cadastrar o novo parâmetro nessa tabela e associar à tabela *link*.

**Parameter Appl**: Essa tabela armazena dados de parâmetros que serão utilizados pela tabela *Application* na estrutura do *Context Database*. Da mesma maneira que a *Parameter* permite uma maior flexibilidade para a tabela *Individuality*, esta tabela tem a mesma função.

Por exemplo, se houver a necessidade de uma mudança no controlador de contexto para a análise de mais parâmetros de para as aplicações cadastradas no controlador e que estejam aptas para terem analisadas o seu contexto dentro da rede, é necessário cadastrar o novo parâmetro nessa tabela e associar à tabela *Application*.

**Application**: Essa tabela armazena os dados sobre as aplicações que estão aptas a serem analisadas dentro do controlador de contexto. Mantendo sua característica de flexibilidade, o modelo de dados do controlador de contexto permite que, a qualquer momento, o administrador de rede cadastre uma nova aplicação e registre as informações relacionadas a esta, para que o controlador de contexto esteja apto a analisar seus dados no processamento dos novos contextos em cada rodada de execução do controlador.

**ApplicationParameter**: Essa tabela armazena os dados sobre os parâmetros cadastrados no *Context Database* para cada aplicação cadastrada pelo administrador. Cada aplicação pode considerar diferentes parâmetros importantes para a análise do contexto e, portanto, de acordo com cada uma aplicação cadastrada podem haver parâmetros distintos associados. Essa tabela é baseada em um relacionamento muitos-para-muitos entre as tabelas *Application* e *Parameter\_Appl*.

Por exemplo, uma aplicação de VoIP (*Voice Over IP*) pode considerar importante analisar os parâmetros variação no atraso e largura de banda e, por esse motivo, esses parâmetros devem estar associados a essa aplicação dentro da tabela *ApplicationParameter*. De forma distinta, em um outro exemplo, uma aplicação de FTP (*File Transfer Protocol*) precisa ter associado somente o parâmetro largura de banda, de acordo com a necessidade do serviço para um funcionamento mais adequado.

Dessa forma, essa tabela refletiria as características das aplicações cadastradas no controlador em relação às suas necessidades de qualidade para um funcionamento mais adequado. É desta maneira, que o controlador de contexto analisará cada aplicação cadastrada para obter a melhor condição de tráfego para o contexto.

**Activity**: Essa tabela armazena os dados relacionados ao relacionamento entre a tabela *Relation* e a tabela *ApplicationParameter*. Dessa forma, para toda a relação, os parâmetros cadastrados para cada aplicação são analisados de forma separada. Assim, podem existir dados de parâmetros que mudam o contexto de rede e outros dados que não mudam. Essa tabela reflete de forma geral todas as relações e todos os seus parâmetros de aplicações.

**Individuality**: Essa tabela armazena os dados relacionados à individualidade, que seria a combinação única de dispositivo (endereço IP e *mac address*), usuário e domínio. Essa individualidade caracteriza determinada coleta de dados em um ponto da rede. Dessa forma, é possível, por exemplo, que o controlador tenha o cadastro de determinado dispositivo em localidades diferentes da rede, mas que são analisadas como elementos diferentes na rede.

**IndividualityParameter**: Essa tabela armazena os dados relacionados aos parâmetros associados às individualidades na rede. Essa tabela permite associar de forma dinâmica novos parâmetros para as individualidades, conforme necessidade operacional do controlador de contexto.

**ForwardingEvent**: Essa tabela armazena os dados relacionados aos novos eventos associados aos caminhos de encaminhamento escolhidos. Essa tabela será populada logo após o módulo *Context-based Forwarding Management* receber a notificação do módulo *Context Model Management* da existência de novo contexto. Após inserir o dado nessa tabela, o *Context-based Forwarding Management* fará a escolha, dentre os caminhos ativos existentes, através do processamento das regras de encaminhamento pré-cadastradas e de acordo com os dados de contexto das aplicações associadas às relações que possuem novo contexto.

**Tables**: Essa tabela armazena os dados relacionados às tabelas do controlador de contexto que terão seus parâmetros analisados pelas regras de QoC, regras de encaminhamento ou validação do contexto.

**Rules**: Essa tabela armazena os dados relacionados às regras de encaminhamento, às regras de contexto e às validações de contexto. Essa estrutura foi projetada para proporcionar agilidade no controlador de contexto nas validações das regras e do contexto. Além disso e focando na flexibilidade, essa tabela proporciona a criação, em tempo de execução no controlador de contexto, de novas validações que podem ser utilizadas tão logo são criadas e sem prejudicar o funcionamento do controlador. Essa estrutura permite um desempenho favorável já que suas regras são feitas com base em comparações de dados entre os dados dessa tabela com as tabelas de *Individuality*, *Device*, *Application* e *Link*.

**Path**: Essa tabela armazena os dados dos caminhos cadastrados pelo administrador de rede. Esses registros descrevem um cadastro básico descritivo do caminho sem detalhar origem-destino e informa a quantidade de saltos que existe naquele caminho.

**Path fwr**: Essa tabela armazena os dados de caminhos e os roteadores ou comutadores origem-destino associados.

**Path link**: Essa tabela armazena os dados de caminhos e as conexões associadas. Essa tabela é um relacionamento muitos-para-muitos entre as tabelas *Path* e *Link*.

**Link**: Essa tabela armazena os dados de conexões entre os roteadores ou comutadores. As conexões são parte do caminho completo e para estes são registradas, em caso de problema, através de mudança de estado, as conexões inativas.

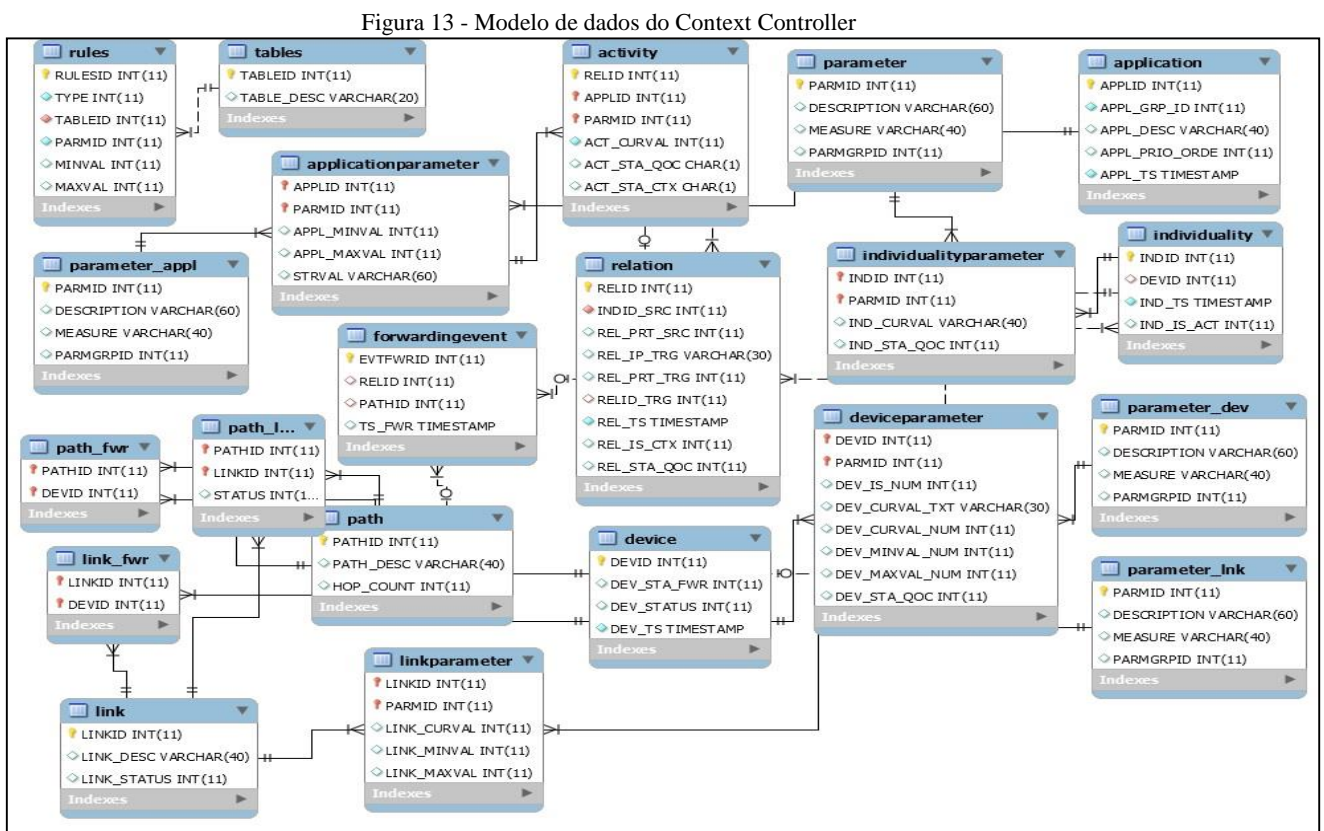
**Link fwr**: Essa tabela armazena os dados de conexões e seus respectivos roteadores ou comutadores origem-destino.

**LinkParameter:** Essa tabela armazena os dados sobre os parâmetros cadastrados no *Context Database* para cada conexão cadastrada pelo administrador de rede. Cada conexão pode considerar diferentes parâmetros importantes para a análise das regras de encaminhamento e, portanto, de acordo com cada conexão cadastrada podem haver parâmetros distintos associados. Essa tabela é baseada em um relacionamento muitos-para-muitos entre as tabelas *Link* e *Parameter\_Ink*.

Essa tabela reflete, em cada rodada de escolha de caminhos para os novos contextos, a situação que os links estão, se ativos ou não, e, também, o desempenho das conexões em relação a largura de banda livre, variação do atraso e atraso, além de taxas de erros, entre outros parâmetros.

Essa estrutura de tabela, como em outras estruturas do modelo de dados do controlador de contexto, permite cadastrar, em tempo real, novos parâmetros sem a necessidade de parar o controlador ou mesmo sem necessitar realizar mudanças importantes no controlador de contexto.

A Figura 13 ilustra o relacionamento das tabelas do controlador de contexto.



#### 4.4 LINGUAGEM DE IMPLEMENTAÇÃO

Para a solução que foi utilizada nesse trabalho para simular os aspectos relacionados à gestão de encaminhamento do CAARF, foi utilizado a linguagem SQL, pois todo processamento e análise das regras de encaminhamento bem como a escolha dos novos caminhos para os fluxos, com novos contextos, é feita em um banco de dados relacional, especificamente o MySQL (MySQL, 2015) nesse trabalho, e, portanto, a linguagem SQL é a linguagem adequada para a manipulação dos dados na base do controlador.

Conforme já descrito nas seções anteriores, após a coleta de dados feita pelo *Context Agent*, através de uma linguagem de marcação como o XML, todo o processamento dos dados desde o registro inicial das informações feito no *Context Handler*; processamento das regras de QoC e a análise do novo contexto feito pelo *Context Model Management*; recuperação do novo contexto feito pelo *Context-based Forwarding Management* e análise das regras de encaminhamento; até a identificação dos novos caminhos a serem aplicados pelo *Flow Adaptation*, na camada de encaminhamento, são feitas através de comandos SQL.

Na diretiva SQL apresentada abaixo, conforme descrito no capítulo 3, é feita a recuperação dos dados de novos contextos do *Context Database* para iniciar a escolha dos novos caminhos pelo *Context-based Forwarding Management*.

A consulta, conforme ANEXO H – letra a, recupera os registros do contexto atual definido pelo *Context Model Management* e a partir desses dados, são inseridos dados na tabela *ForwardingEvent*, conforme ANEXO H – letra b, com informações iniciais informando a qual relação aquele evento de encaminhamento está relacionado.

Após essa etapa, é feita a análise das regras de encaminhamento através da consulta no ANEXO H – letra c, que irá recuperar somente as conexões que tiveram seus dados considerados aptos para serem utilizados, baseados nas regras.

A consulta acima obtém somente as conexões ativas dentro da tabela *link* e os roteadores ou comutadores ativos, usando a tabela *device*. Esses estados tanto da conexão quanto do dispositivo de encaminhamento são modificados por um processo do *Context Handler* quando da leitura do XML com os dados de coleta dos roteadores ou comutadores que estão no domínio do controlador de contexto. Portanto, essa verificação se torna rápida causando o menor impacto possível nessa etapa.

A partir dos caminhos validados na etapa anterior, é feita a análise do caminho mais adequado para aquela relação. Sendo assim, conforme o caminho, origem e destino da comunicação da aplicação em determinada relação, são escolhidos as opções de caminhos e, dentro desse universo, é escolhido um único caminho para ser associado a determinada relação e assim ser adaptado à camada de encaminhamento, posteriormente.

A partir das informações de conexões e seus valores correntes selecionados, é feita uma verificação dentre os maiores valores do que foi selecionado, em caso de largura de banda livre por exemplo, ou dos menores valores do que foi selecionado, em caso de atraso ou variação do atraso. Em caso de conexões com os mesmos valores, é escolhida uma delas aleatoriamente, conforme consulta ANEXO H – letra d.

Depois da escolha das conexões que compõem o caminho, é selecionado o caminho e este é registrado na tabela *ForwardingEvent* através do comando SQL em ANEXO H – letra e.

Por fim, depois de todo o processamento feito das regras de encaminhamento e da escolha de um novo caminho a ser adaptado, o módulo *Flow Adaptation* consulta os novos caminhos a serem aplicados, conforme em ANEXO H – letra f, e faz acesso à camada de encaminhamento e adapta esses novos caminhos para permitir uma melhor qualidade no tráfego para as relações que mudaram o contexto. Dando assim dinamicidade no roteamento do tráfego da rede para determinadas aplicações.

Na próxima seção, serão descritos alguns aspectos observados na simulação bem como serão exploradas algumas lições aprendidas nesse processo.

#### 4.5 DISCUSSÃO E LIÇÕES APRENDIDAS

A utilização do banco de dados relacional juntamente com a sua linguagem de manipulação, o SQL, proporcionou uma padronização dos dados de coleta. Dessa forma, essa característica do arcabouço CAARF propicia a obtenção de informações mais precisas sobre as condições da rede e sobre o seu perfil de tráfego.

A padronização, também, propicia uma documentação da estrutura usada no controlador de contexto (CAARF) através da descrição dos campos. A estrutura definida no banco prevalece o entendimento do funcionamento do arcabouço apenas com o conhecimento do modelo de dados, entendendo o relacionamento que existe entre as tabelas.

Apesar dessas restrições, o uso de banco de dados permitiu padronizar a análise durante as validações e assim facilitar o registro de novas validações a serem feitas e sem proporcionar grandes modificações na simulação. Foi possível, por exemplo, ativar uma regra de QoC durante a simulação, fazendo com que nas rodadas posteriores à mudança, a regra já estivesse valendo sem haver a menor intervenção no processo de simulação já construído e modelado.

Essas foram três importantes contribuições, destas tecnologias, para a manipulação de dados: a modelagem, a flexibilidade e a facilidade de manipulação dos dados durante a simulação.

#### 4.6 CONCLUSÃO

A simulação da gestão de encaminhamento permite avaliar não somente a funcionalidade da operação, como um todo, de um novo arcabouço como, também, permite avaliar, por exemplo, o uso de processamento de ações usando um banco de dados relacional, juntamente com a sua linguagem de manipulação, o SQL.

De certa maneira, a utilização desses recursos proporcionou certa dificuldade, inicialmente, no tratamento das regras ou validações, pois o uso de banco de dados relacional e suas características segue uma estruturação que, se for bem utilizada, pode trazer benefícios como, por exemplo o desempenho.



## 5 ESTUDOS DE CASO

Nesse capítulo são descritos, apresentados e ilustrados os estudos de casos realizados utilizando a simulação do módulo de gestão de encaminhamento descrita no capítulo anterior.

Nas seções posteriores são apresentados os detalhes que envolveram cada uma das simulações e suas rodadas (cada etapa de execução de um experimento), descrevendo os aspectos e as nuances de cada cenário para demonstrar as adaptações dos fluxos baseados nos contextos que se apresentaram em cada simulação. Sendo assim, no fim de cada rodada de cada cenário serão apresentados os resultados da mudança ou não dos caminhos baseada em contexto.

Em cada um dos cenários, são detalhados os fluxos, a evolução do contexto e os gráficos de antes e depois dos novos contextos. A partir dos gráficos, é possível estabelecer um comparativo entre as rodadas de simulação antes e depois do contexto para demonstrar qual foi a influência, em cada cenário, do contexto da rede que se apresentou em cada demonstração.

O intuito de realizar as simulações foi provar que, baseado em informações contextuais, a abordagem de encaminhamento mostrou-se efetiva e gerou mudanças nos caminhos de forma a colocar os fluxos em canais de melhor qualidade de forma adequada para determinada aplicação.

Os cenários de simulação foram estruturados da seguinte forma:

- **Cenário 1:** Carga e complexidade Baixas
  - Objetivo: Apresentar o funcionamento básico do arcabouço, no que se refere à gestão de encaminhamento baseado em contexto;
  - Execuções:
    - Execução 1: Baixa - Melhor Esforço;
    - Execução 2: Baixa – Contexto;
- **Cenário 2:** Carga e complexidade Média
  - Objetivo: Apresentar o funcionamento do arcabouço com um volume maior de fluxos e avaliação de QoE feita na gerência de encaminhamento, além da validação das regras de encaminhamento;
  - Execuções:
    - Execução 1: Média - Melhor Esforço;
    - Execução 2: Média – Contexto;
- **Cenário 3:** Carga e complexidade Alta

- Objetivos: Apresentar o funcionamento do arcabouço com um volume maior de fluxos e avaliações de QoE e QoD feita na gerência de encaminhamento, além da validação das regras de encaminhamento;
- Execuções:
  - Execução 1: Alta - Melhor Esforço;
  - Execução 2: Alta – Contexto.

Portanto, as seções posteriores estão organizadas da seguinte maneira: Ambiente de simulação que está dividido em subseções com a topologia usada nos cenários, caracterização dos canais e aplicações utilizadas; primeiro cenário de simulação, segundo cenário de simulação, terceiro cenário de simulação e as conclusões sobre os resultados obtidos nas simulações.

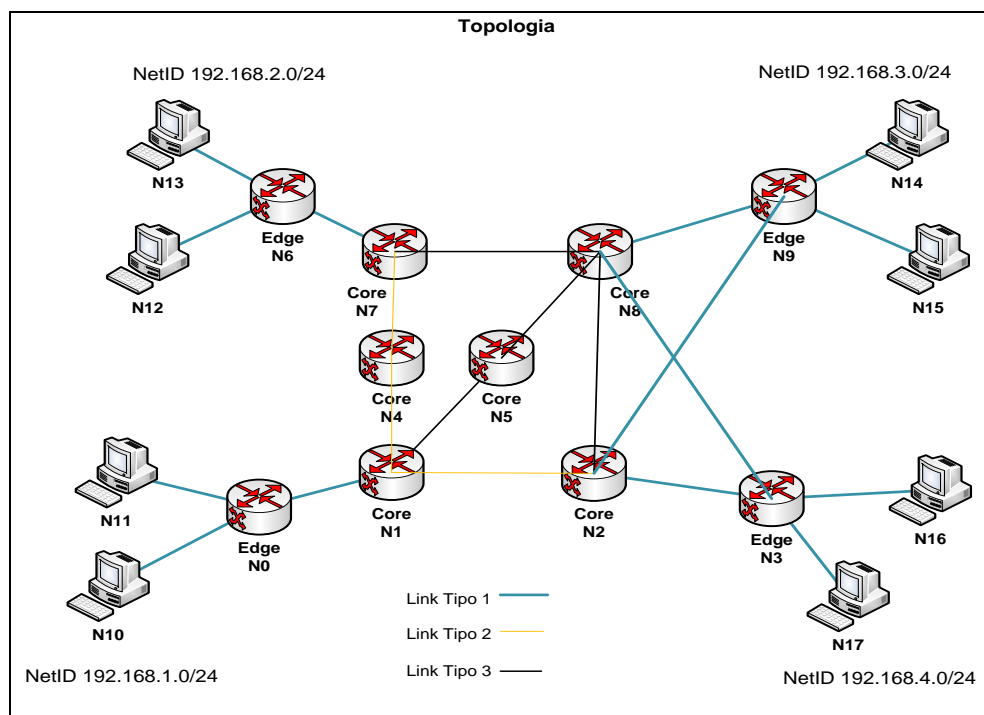
## 5.1 CARACTERIZAÇÃO DO EXPERIMENTO

### 5.1.1 Topologia utilizada nos cenários

Nessa seção é apresentada a topologia usada nos três cenários que são apresentados nesse capítulo.

A Figura 14 apresenta a ilustração da topologia.

Figura 14 - Topologia dos cenários de simulação



A seguir descreveremos a topologia da Figura 14:

- Quatro (4) roteadores de borda (*edge*) onde cada um dos quatro (4) conectados às suas redes internas e estes conectados aos roteadores do núcleo (*core*) da rede;
- Seis (6) roteadores de núcleo (*core*) que formam o núcleo da rede;
- Os canais de ligação entre os roteadores são de três (3) tipos:
  - Em cor azul, é um canal com largura de banda de 8MB e atraso de 1ms;
  - Em cor laranja, é um canal com largura de banda de 1,5MB e atraso de 5 ms;
  - Em cor preta, é um canal com largura de banda de 3MB e atraso de 50 ms.
- Além do núcleo da rede e das bordas, são apresentados os nós, que representam as estações de trabalho, que foram definidas em número de oito (8), com duas (2) estações ligadas a cada roteador edge. Além da caracterização mostrada na Figura 14, segue na Tabela 1 a lista das estações usadas nesse cenário e quais roteadores estão associados à rede. A descrição dos nós de estação segue na Tabela 1. Tabela 1 Nas Figuras 15, 16, 17 e 18 e seguem os comandos em linguagem SQL que servem para incluir os registros de estações e encaminhadores.

Na Tabela 2 segue a lista de encaminhadores e suas interconexões.

Tabela 1 - Lista de estações e seus roteadores

<b>Nó</b>	<b>IP</b>	<b>Processador</b>	<b>Memória</b>	<b>Mascara</b>	<b>Gateway</b>	<b>Conexão</b>
N10	192.168.1.10	1,2 GHz	2GB	255.255.255.0	192.168.1.1	N0
N11	192.168.1.11	1,2 GHz	2GB	255.255.255.0	192.168.1.1	N0
N12	192.168.2.12	1,6 GHz	4GB	255.255.255.0	192.168.2.1	N6
N13	192.168.2.13	1,6 GHz	4GB	255.255.255.0	192.168.2.1	N6
N14	192.168.3.14	1,2 GHz	2GB	255.255.255.0	192.168.3.1	N9
N15	192.168.3.15	1,2 GHz	2GB	255.255.255.0	192.168.3.1	N9
N16	192.168.4.16	1,6 GHz	4GB	255.255.255.0	192.168.4.1	N3
N17	192.168.4.17	1,6 GHz	4GB	255.255.255.0	192.168.4.1	N3

Figura 15 - Informações das estações no Banco

```

INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (10, 0, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (11, 0, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (12, 0, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (13, 0, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (14, 0, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (15, 0, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (16, 0, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (17, 0, 1);

```

Figura 16 - Informações das estações no Banco (Parâmetros)

```

INSERT INTO DEVICEPARAMETER (DEVID, PARMID, DEV_IS_NUM, DEV_CURVAL_TXT, DEV_CURVAL_NUM) VALUES (10, 8, 1, NULL, 1200);
INSERT INTO DEVICEPARAMETER (DEVID, PARMID, DEV_IS_NUM, DEV_CURVAL_TXT, DEV_CURVAL_NUM) VALUES (10, 9, 1, NULL, 2000);
INSERT INTO DEVICEPARAMETER (DEVID, PARMID, DEV_IS_NUM, DEV_CURVAL_TXT, DEV_CURVAL_NUM) VALUES (10, 12, 0, 'N10', NULL);
INSERT INTO DEVICEPARAMETER (DEVID, PARMID, DEV_IS_NUM, DEV_CURVAL_TXT, DEV_CURVAL_NUM) VALUES (10, 22, 0, '192.168.1.0', NULL);
INSERT INTO DEVICEPARAMETER (DEVID, PARMID, DEV_IS_NUM, DEV_CURVAL_TXT, DEV_CURVAL_NUM) VALUES (10, 26, 0, '192.168.1.10', NULL);
INSERT INTO DEVICEPARAMETER (DEVID, PARMID, DEV_IS_NUM, DEV_CURVAL_TXT, DEV_CURVAL_NUM) VALUES (10, 47, 0, '255.255.255.0', NULL);
INSERT INTO DEVICEPARAMETER (DEVID, PARMID, DEV_IS_NUM, DEV_CURVAL_TXT, DEV_CURVAL_NUM) VALUES (10, 49, 1, NULL, 10);

```

Tabela 2 - Lista de Encaminhadores

<i>Nó</i>	<i>Tipo Encaminhador</i>	<i>Interconexão</i>
<i>N0</i>	<i>Edge</i>	<i>N1</i>
<i>N1</i>	<i>Core</i>	<i>N0, N2, N4, N5</i>
<i>N2</i>	<i>Core</i>	<i>N1, N3, N8, N9</i>
<i>N3</i>	<i>Edge</i>	<i>N2</i>
<i>N4</i>	<i>Core</i>	<i>N1, N7</i>
<i>N5</i>	<i>Core</i>	<i>N1, N8</i>
<i>N6</i>	<i>Edge</i>	<i>N7</i>
<i>N7</i>	<i>Core</i>	<i>N4, N6, N8</i>
<i>N8</i>	<i>Core</i>	<i>N2, N3, N5, N7, N9</i>
<i>N9</i>	<i>Edge</i>	<i>N2, N8</i>

Figura 17 - Informações dos Encaminhadores no Banco

```

INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (0, 1, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (1, 1, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (2, 1, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (3, 1, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (4, 1, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (5, 1, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (6, 1, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (7, 1, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (8, 1, 1);
INSERT INTO DEVICE (DEVID, DEV_STA_FWR, DEV_STATUS) VALUES (9, 1, 1);

```

Figura 18 - Informações dos Encaminhadores no Banco (Parâmetros)

```

INSERT INTO DEVICEPARAMETER (DEVID, PARMID, DEV_IS_NUM, DEV_CURVAL_TXT) VALUES (0, 12, 0, 'N0');
INSERT INTO DEVICEPARAMETER (DEVID, PARMID, DEV_IS_NUM, DEV_CURVAL_TXT) VALUES (0, 22, 0, '192.168.1.0');
INSERT INTO DEVICEPARAMETER (DEVID, PARMID, DEV_IS_NUM, DEV_CURVAL_TXT) VALUES (0, 26, 0, '192.168.1.1');
INSERT INTO DEVICEPARAMETER (DEVID, PARMID, DEV_IS_NUM, DEV_CURVAL_TXT) VALUES (0, 47, 0, '255.255.255.0');
INSERT INTO DEVICEPARAMETER (DEVID, PARMID, DEV_IS_NUM, DEV_CURVAL_TXT) VALUES (0, 49, 1, 10);

```

Tabela 3 - Dispositivos e Roteadores configurados no NS-2

```

# Routers                                     # Hosts
set n0 [$ns node]                             set n10 [$ns node]
set n1 [$ns node]                             set n11 [$ns node]
set n2 [$ns node]                             set n12 [$ns node]
set n3 [$ns node]                             set n13 [$ns node]
set n4 [$ns node]                             set n14 [$ns node]
set n5 [$ns node]                             set n15 [$ns node]
set n6 [$ns node]                             set n16 [$ns node]
set n7 [$ns node]                             set n17 [$ns node]
set n8 [$ns node]                             # Hosts
set n9 [$ns node]
# Routers

```

Conforme apresentado nas Figuras 15, 16, 17 e 18, as informações dos dispositivos de clientes e dos dispositivos de encaminhamento são, previamente, cadastradas nas tabelas do ContextDatabase, a fim de que sejam identificadas quando na execução das consultas para análise do melhor caminho pela gestão de encaminhamento.

### 5.1.2 Definição dos Canais

Para todos os cenários, as conexões entre os encaminhadores têm a seguinte configuração, conforme Tabela 4:

Tabela 4 - Tabela de Conexões entre Encaminhadores

Conexão	Origem	Destino	Banda	Atraso
1	N0	N1	8MB	1ms

2	N1	N2	1,5MB	5ms
3	N1	N4	1,5MB	5ms
4	N1	N5	3MB	50ms
5	N2	N3	8MB	1ms
6	N2	N9	8MB	1ms
7	N2	N8	3MB	50ms
8	N3	N8	8MB	1ms
9	N4	N7	1,5MB	5ms
10	N5	N8	3MB	50ms
11	N6	N7	8MB	1ms
12	N7	N8	3MB	50ms
13	N8	N9	8MB	1ms

Essas informações dos encaminhadores e estações ativas na rede são armazenadas no *ContextDatabase*, conforme ilustrado nas Figuras 15, 16, 17 e 18. Além da configuração dos encaminhadores, é realizada a configuração das conexões entre os encaminhadores que apresenta a largura de banda e o atraso entre eles.

Dentro do *ContextDatabase* também é realizada a associação de todas as conexões entre encaminhadores com os caminhos possíveis entre uma origem e um destino, de forma que seja possível identificar todas as alternativas existentes de caminhos para uma determinada comunicação na rede.

A Figura 19 ilustra as informações que são armazenadas em banco de dados para serem utilizadas na análise da gestão de encaminhamento:

Figura 19 - Informações de Caminhos e Links no banco

```

INSERT INTO LINK (LINKID, LINK_DESC, LINK_STATUS) VALUES (1, 'LINK DE NO 0 A NO 1', 1);
INSERT INTO LINK (LINKID, LINK_DESC, LINK_STATUS) VALUES (2, 'LINK DE NO 1 A NO 2', 1);
INSERT INTO LINK (LINKID, LINK_DESC, LINK_STATUS) VALUES (3, 'LINK DE NO 1 A NO 4', 1);

INSERT INTO LINKPARAMETER (LINKID, PARMID, LINK_CURVAL, LINK_MINVAL, LINK_MAXVAL) VALUES (1, 1, 0, 0, 8000);
INSERT INTO LINKPARAMETER (LINKID, PARMID, LINK_CURVAL, LINK_MINVAL, LINK_MAXVAL) VALUES (1, 2, 1, 0, 1);
INSERT INTO LINKPARAMETER (LINKID, PARMID, LINK_CURVAL, LINK_MINVAL, LINK_MAXVAL) VALUES (1, 5, 8000, 0, 8000);

INSERT INTO LINK_FWR (LINKID, DEVID) VALUES (1,0);
INSERT INTO LINK_FWR (LINKID, DEVID) VALUES (1,1);
INSERT INTO LINK_FWR (LINKID, DEVID) VALUES (2,1);

INSERT INTO PATH (PATHID, PATH_DESC, HOP_COUNT) VALUES (1, 'CAMINHO DE NO 0 A NO 3', 3);
INSERT INTO PATH_FWR (PATHID, DEVID) VALUES (1,0);
INSERT INTO PATH_FWR (PATHID, DEVID) VALUES (1,1);
INSERT INTO PATH_FWR (PATHID, DEVID) VALUES (1,2);

INSERT INTO PATH_LINK (PATHID, LINKID, STATUS ) VALUES (1,1,1);
INSERT INTO PATH_LINK (PATHID, LINKID, STATUS ) VALUES (1,2,1);
INSERT INTO PATH_LINK (PATHID, LINKID, STATUS ) VALUES (1,5,1);

```

Tabela 5 - Definição dos canais no NS-2

# Local Links	# Core Links	#Give node position (for NAM)
\$ns duplex-link \$n11 \$n0 8.0Mb 1ms DropTail	\$ns duplex-link \$n7 \$n4 1.5Mb 5ms DropTail	\$ns duplex-link-op \$n11 \$n0 orient right-down
\$ns queue-limit \$n11 \$n0 50	\$ns queue-limit \$n7 \$n4 50	\$ns duplex-link-op \$n10 \$n0 orient right-up
\$ns duplex-link \$n10 \$n0 8.0Mb 1ms DropTail	\$ns duplex-link \$n4 \$n1 1.5Mb 5ms DropTail	\$ns duplex-link-op \$n0 \$n1 orient right
\$ns queue-limit \$n10 \$n0 50	\$ns queue-limit \$n4 \$n1 50	\$ns duplex-link-op \$n12 \$n6 orient right-up
\$ns duplex-link \$n0 \$n1 8.0Mb 1ms DropTail	\$ns duplex-link \$n7 \$n8 3.0Mb 50ms DropTail	\$ns duplex-link-op \$n13 \$n6 orient right-down
\$ns queue-limit \$n0 \$n1 50	\$ns queue-limit \$n7 \$n8 50	\$ns duplex-link-op \$n6 \$n7 orient right
\$ns duplex-link \$n12 \$n6 8.0Mb 1ms DropTail	\$ns duplex-link \$n1 \$n2 1.5Mb 5ms DropTail	\$ns duplex-link-op \$n14 \$n9 orient left-down
\$ns queue-limit \$n12 \$n6 50	\$ns queue-limit \$n1 \$n2 50	\$ns duplex-link-op \$n15 \$n9 orient left-up
\$ns duplex-link \$n13 \$n6 8.0Mb 1ms DropTail	# Core Links	\$ns duplex-link-op \$n9 \$n8 orient left-down
\$ns queue-limit \$n13 \$n6 50	# Local Links	\$ns duplex-link-op \$n16 \$n3 orient left-down
\$ns duplex-link \$n6 \$n7 8.0Mb 1ms DropTail	\$ns duplex-link \$n9 \$n2 8.0Mb 1ms DropTail	\$ns duplex-link-op \$n17 \$n3 orient left-up
\$ns queue-limit \$n6 \$n7 50	\$ns queue-limit \$n9 \$n2 50	\$ns duplex-link-op \$n3 \$n2 orient left-up
\$ns duplex-link \$n14 \$n9 8.0Mb 1ms DropTail	\$ns duplex-link \$n3 \$n8 8.0Mb 1ms DropTail	\$ns duplex-link-op \$n7 \$n4 orient left-down
\$ns queue-limit \$n14 \$n9 50	\$ns queue-limit \$n3 \$n8 50	\$ns duplex-link-op \$n4 \$n1 orient left-down
\$ns duplex-link \$n15 \$n9 8.0Mb 1ms DropTail	\$ns duplex-link \$n5 \$n8 3.0Mb 50ms DropTail	\$ns duplex-link-op \$n7 \$n8 orient right
\$ns queue-limit \$n15 \$n9 50	\$ns queue-limit \$n5 \$n8 50	\$ns duplex-link-op \$n1 \$n2 orient right
\$ns duplex-link \$n9 \$n8 8.0Mb 1ms DropTail	\$ns duplex-link \$n5 \$n1 3.0Mb 50ms DropTail	\$ns duplex-link-op \$n9 \$n2 orient left-down
\$ns queue-limit \$n9 \$n8 50	\$ns queue-limit \$n5 \$n1 50	\$ns duplex-link-op \$n3 \$n8 orient left-up
\$ns duplex-link \$n16 \$n3 8.0Mb 1ms DropTail	\$ns duplex-link \$n8 \$n2 3.0Mb 50ms DropTail	\$ns duplex-link-op \$n5 \$n8 orient right-up
\$ns queue-limit \$n16 \$n3 50	\$ns queue-limit \$n8 \$n2 50	\$ns duplex-link-op \$n5 \$n1 orient left-down
\$ns duplex-link \$n17 \$n3 8.0Mb 1ms DropTail	# Local Links	\$ns duplex-link-op \$n8 \$n2 orient left-down
\$ns queue-limit \$n17 \$n3 50		
\$ns duplex-link \$n3 \$n2 8.0Mb 1ms DropTail		
\$ns queue-limit \$n3 \$n2 50		
# Local Links		

### 5.1.3 Definição das aplicações usadas

Para todos os cenários utilizamos as aplicações a seguir, conforme Tabela 6:

Tabela 6 - Aplicações usadas nos três cenários

Aplicação	Protocolo	Codificação	Vazão
	<b>Transporte</b>		
VoIP	UDP	H911	64 Kbps
VoD	UDP	Não se aplica	Disponível
VoD HD	UDP	Não se aplica	Disponível

Figura 20 - Informações das Aplicações no banco

```
INSERT INTO APPLICATION (APPLID, APPL_GRP_ID, APPL_DESC, APPL_TS) VALUES (13, 5, 'WINSXP', '2015-07-19 20:36:56');
INSERT INTO APPLICATIONPARAMETER (APPLID, PARMID, APPL_MINVAL, APPL_MAXVAL) VALUES (13, 1, 1, 50);
INSERT INTO APPLICATIONPARAMETER (APPLID, PARMID, APPL_MINVAL, APPL_MAXVAL) VALUES (13, 2, 1, 100);
```

Tabela 7 - Definição das Aplicações no NS-2

<pre>set cbr0 [new Application/Traffic/CBR] \$cbr0 attach-agent \$udp2 \$cbr0 set packetSize_ 160 \$cbr0 set interval_ 0.005 \$cbr0 set codePt_ 0 \$cbr0 set rate_ 80Kb</pre>	<pre>set cbr1 [new Application/Traffic/CBR] \$cbr1 attach-agent \$udp0 \$cbr1 set packetSize_ 1000 \$cbr1 set interval_ 0.005 \$cbr1 set codePt_ 0 \$cbr1 set rate_ 2Mb</pre>
---	---

## 5.2 PRIMEIRO CENÁRIO

Nesse primeiro cenário, foi utilizada uma configuração de simulação de menor carga para demonstrar que o contexto pode influenciar na melhoria da comunicação de determinada aplicação entre duas (2) estações. Para isso, o controlador faz a mudança de caminho, com a opções apresentadas pela gestão de encaminhamento e as adapta na camada de encaminhamento.

Portanto, o cenário apresentado nessa seção é mais simples, porém com a mesma importância para o controlador no que diz respeito à análise das informações contextuais para efetivar ou não a mudança de caminho dos fluxos.

Na Tabela 8 são apresentados os tipos de fluxos que serão tratados no cenário dessa simulação e em que intervalo irão executar:

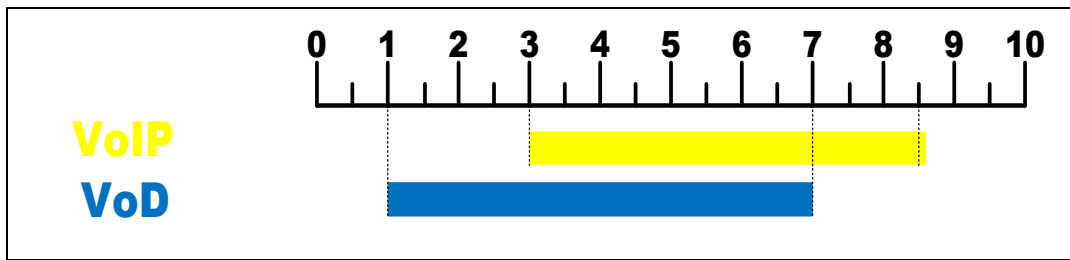
Tabela 8 - Fluxo e o intervalo de simulação em detalhe

<b>TIPOS DE TRÁFEGO:</b>	<b>CAMINHO INICIAL:</b>	<b>DURAÇÃO</b>
<b>CBR1- UDP (VOIP)</b>	{N6 ↔ N7 ↔ N8 ↔ N9}	3 a 8.5 min.
<b>CBR2 - UDP (VoD)</b>	{N0 ↔ N1 ↔ N2 ↔ N3}	1 a 7 min.

Na Figura 21, é apresentada a evolução de cada fluxo durante a simulação do cenário 1. A intersecção dos fluxos ocorre a partir do minuto 3 quando é iniciada a comunicação VoIP.

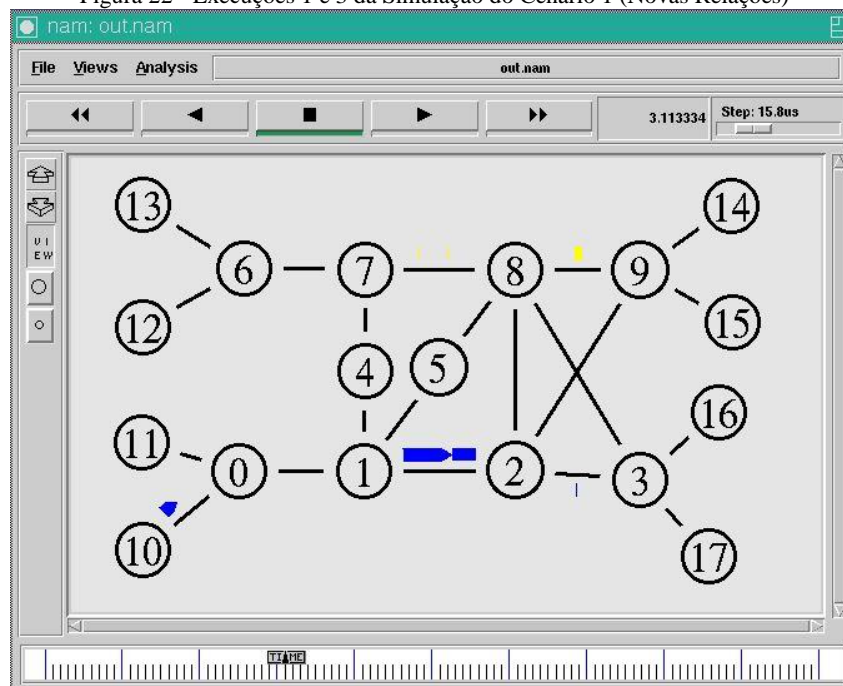


Figura 21 - Evolução dos fluxos na simulação (em minutos)



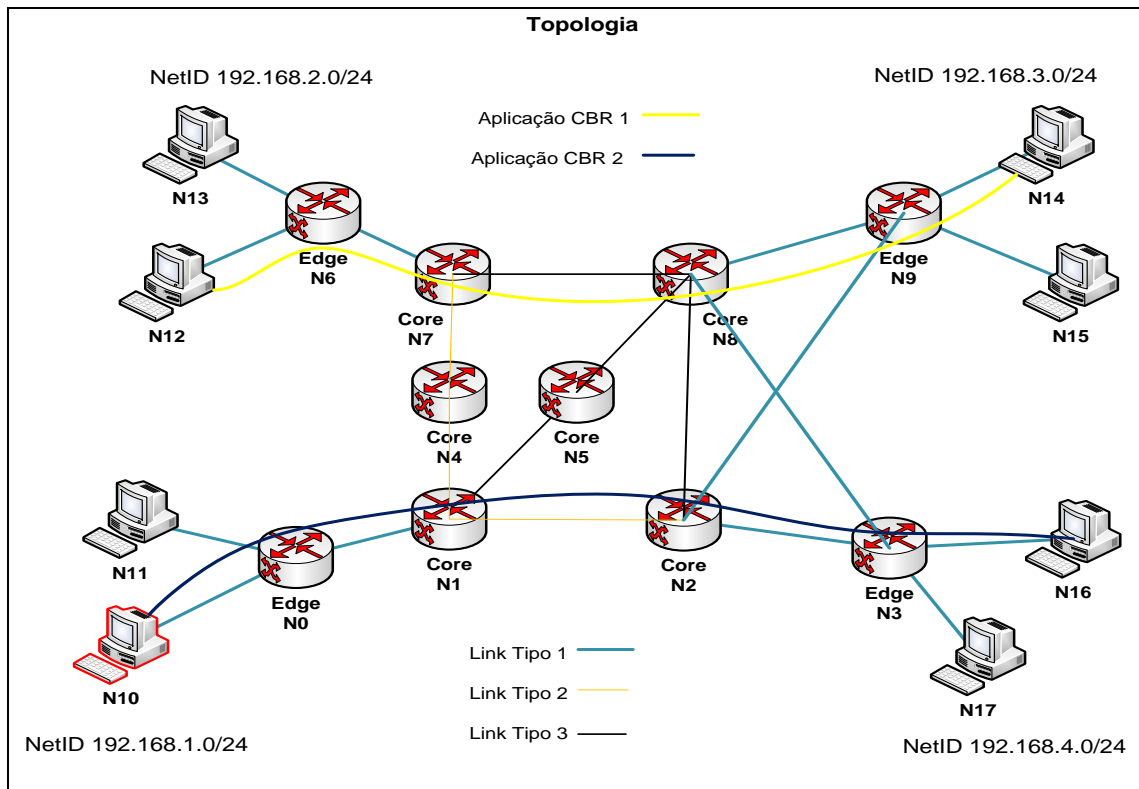
Na Figura 21 é apresentada uma captura de tela da execução dos fluxos iniciais das aplicações VoIP e VoD utilizando a ferramenta NAM (NAM, 2015). No exemplo mostrado nessa figura, são mostrados os dois (2) fluxos usando o caminho inicial não escolhido pelo controlador de contexto.

Figura 22 - Execuções 1 e 3 da Simulação do Cenário 1 (Novas Relações)



Todas as relações iniciais de dois (2) nós origem-destino específicos, optam inicialmente, como mostra a Figura 22, por um caminho de melhor esforço sem que o controlador de contexto influencie na escolha. A partir das próprias interações entre origem-destino, os agentes de contexto (*Context Agent*) coletam dados, conforme mostrado no documento xml apresentado na seção 4.5, e os enviam para o controlador de contexto. A partir desse momento, as informações de suas comunicações avaliadas pelo controlador, podem ou não ter os caminhos de seus fluxos, eventualmente, mudados.

Figura 23 - Fluxos do Cenário 1 (Novas Relações)



Na Figura 23, é mostrado o desenho dos fluxos do Cenário 1 quando ocorrem as novas relações usando as aplicações VoIP e VoD.

Depois da nova relação ser iniciada, os novos contextos passam a ser analisados pelo controlador como será mostrado nas execuções em seguida.

Na Tabela 9, são mostrados alguns parâmetros que foram medidos durante o início das relações. Esses mesmos parâmetros são comparados depois do final da simulação para serem avaliados os ganhos implementados com uso da gestão de encaminhamento baseada em contexto.

Tabela 9 - Resultados do Cenário 1 –Antes da 1ª Execução

Aplicação	Transmitidos	Atraso Fim a Fim	Largura de Banda
VoIP	480000 bits	0.011 ms	80.21 Kbps
VoD	8648000 bits	0.046 ms	1501.39Kbps

### 1ª Execução

Após consulta na visão de contexto *ContextModelView* dentro do *ContextDatabase*, são os obtidos os novos contextos para terem seus caminhos analisados e, caso seja conveniente para

uma melhor comunicação, poderem ter seus caminhos alterados de forma dinâmica durante uma relação ativa entre duas (2) máquinas na rede.

Conforme mostrado na seção 4.5 e citado anteriormente, é executada a consulta *select \* from relation where rel\_is\_ctx = 1*, a qual representa a visão *ContextModelView*, e que retorna todas as relações que na gestão de contexto foram analisadas e marcadas como novo contexto de rede em um dado momento específico.

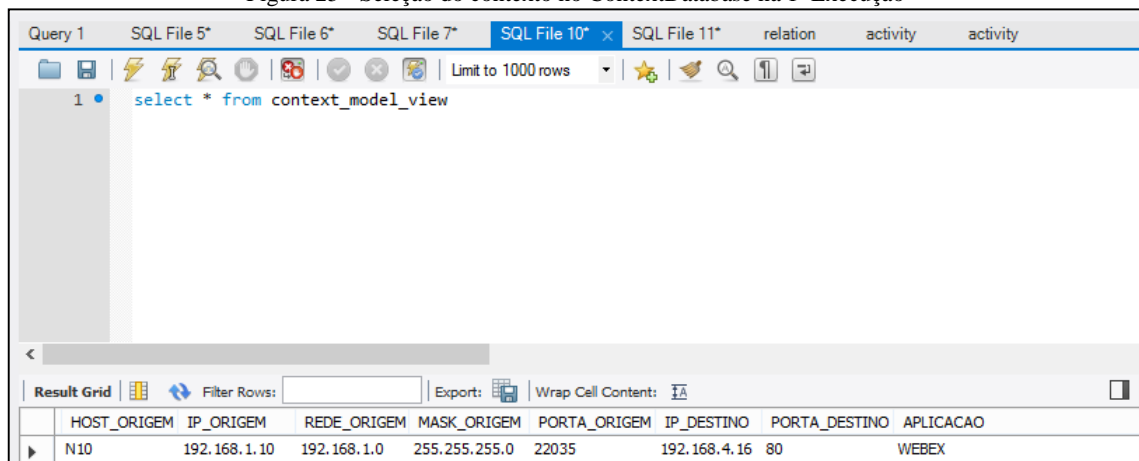
Mais, especificamente, a consulta a seguir (Figura 24) seleciona as principais informações para avaliar o contexto para a gestão de encaminhamento definida pelo arcabouço CAARF.

Figura 24 - Visão ContextModelView

```
select pai.relid pai, i1.dev_curval_txt as HOST_ORIGEM, i3.ind_curval as IP_ORIGEM, i1.ind_curval as REDE_ORIGEM,
       i2.ind_curval as MASK_ORIGEM, REL_PRT_SRC as PORTA_ORIGEM, pai.REL_IP_TRG as IP_DESTINO, pai.REL_PRT_TRG as PORTA_DESTINO,
       a3.strval as APLICACAO, a1.appl_minval as APPL_VAZAO, a1.order as ORDEM_VAZAO, a2.appl_maxval as APPL_ATRASO,
       a2.order as ORDEM_ATRASO
from (select * from relation where rel_is_ctx = 1) pai
     inner join ( select b.*,c.devid,c.dev_curval_txt from individuality a inner join individualityparameter b on
                 (a.indid = b.indid and b.parmid = 22)
                 inner join deviceparameter c on
                 (a.devid = c.devid and c.parmid = 12)) i1 on (pai.indid_src = i1.indid)
     inner join ( select b.*
                 from individuality a inner join individualityparameter b on
                 (a.indid = b.indid and b.parmid = 47)) i2 on (pai.indid_src = i2.indid)
     inner join ( select b.*
                 from individuality a inner join individualityparameter b on
                 (a.indid = b.indid and b.parmid = 26)) i3 on (pai.indid_src = i3.indid)
     inner join ( select b.*, a.relid, a.act_curval from activity a inner join applicationparameter b on
                 (a.applid = b.applid and a.parmid = b.parmid and b.parmid = 1)) a1 on (a1.relid = pai.relid)
     inner join ( select b.*, a.relid, a.act_curval from activity a inner join applicationparameter b on
                 (a.applid = b.applid and a.parmid = b.parmid and b.parmid = 2)) a2 on (a2.relid = pai.relid)
     inner join ( select b.*, a.relid
                 from activity a inner join applicationparameter b on
                 (a.applid = b.applid and a.parmid = b.parmid and b.parmid = 31)) a3 on (a3.relid = pai.relid)
```

Nessa primeira execução do cenário 1, ocorre a coleta e processamento de informações de contexto a partir da relação existente entre as estações N10 e N16. Portanto, a resposta do banco de dados à requisição de visão do contexto seria como ilustrada na Figura 25.

Figura 25 - Seleção do contexto no ContextDatabase na 1ª Execução

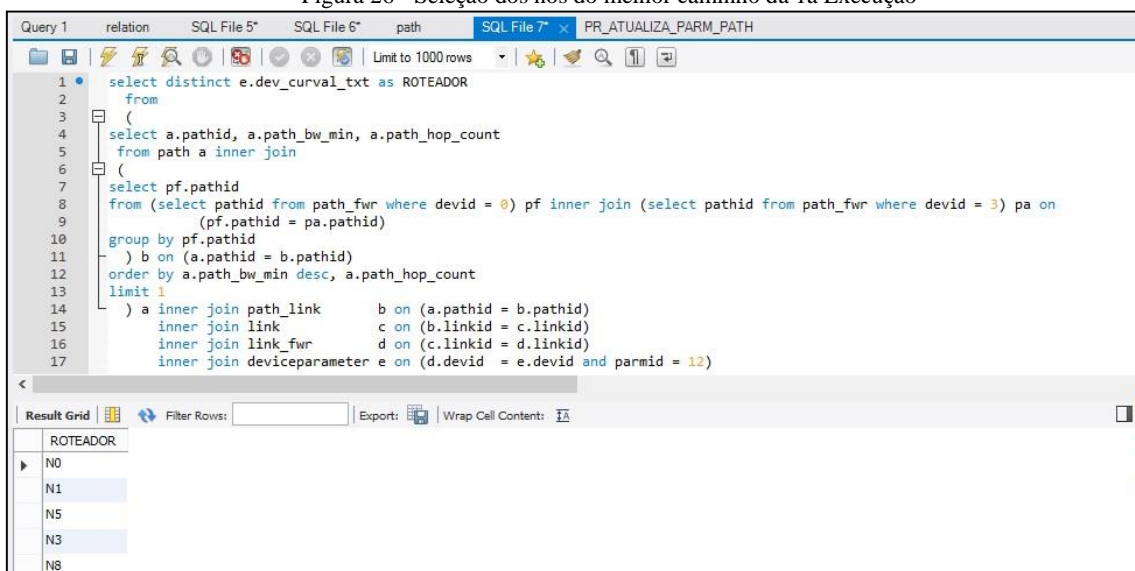


The screenshot shows a database query tool interface. The top toolbar includes icons for file operations and a 'Limit to 1000 rows' dropdown. The main area displays a SQL query: `select * from context_model_view`. Below the query, a 'Result Grid' is visible with the following data:

HOST_ORIGEM	IP_ORIGEM	REDE_ORIGEM	MASK_ORIGEM	PORTA_ORIGEM	IP_DESTINO	PORTA_DESTINO	APLICACAO
N10	192.168.1.10	192.168.1.0	255.255.255.0	22035	192.168.4.16	80	WEBEX

Baseado na consulta da Figura 26 são avaliadas as alternativas de caminho entre N10 e N16 e assim é feita a escolha do melhor caminho para que essa relação tenha a melhor qualidade.

Figura 26 - Seleção dos nós do melhor caminho da 1ª Execução



The screenshot shows a database query tool interface. The top toolbar includes icons for file operations and a 'Limit to 1000 rows' dropdown. The main area displays a complex SQL query:

```

1 select distinct e.dev_curval_txt as ROTEADOR
2   from
3   (
4     select a.pathid, a.path_bw_min, a.path_hop_count
5     from path a inner join
6     (
7       select pf.pathid
8       from (select pathid from path_fwr where devid = 0) pf inner join (select pathid from path_fwr where devid = 3) pa on
9        (pf.pathid = pa.pathid)
10      group by pf.pathid
11     ) b on (a.pathid = b.pathid)
12     order by a.path_bw_min desc, a.path_hop_count
13     limit 1
14   ) a inner join path_link      b on (a.pathid = b.pathid)
15     inner join link            c on (b.linkid = c.linkid)
16     inner join link_fwr       d on (c.linkid = d.linkid)
17     inner join deviceparameter e on (d.devid = e.devid and parmId = 12)

```

Below the query, a 'Result Grid' is visible with the following data:

ROTEADOR
N0
N1
N5
N3
N6

## 2ª Execução

Após consulta na visão de contexto *ContextModelView* dentro do *ContextDatabase*, conforme Figura 27 são obtidos os novos contextos para terem seus caminhos analisados e, caso seja conveniente para uma melhor comunicação, poderem ter seus caminhos alterados de forma dinâmica durante uma relação ativa entre duas (2) máquinas na rede.

Figura 27 - Seleção do contexto no ContextDatabase na 2ª Execução

The screenshot shows a SQL query execution window with the following query:

```
1 • select * from context_model_view
```

The results are displayed in a table with the following columns and data:

HOST_ORIGEM	IP_ORIGEM	REDE_ORIGEM	MASK_ORIGEM	PORTA_ORIGEM	IP_DESTINO	PORTA_DESTINO	APLICACAO
:N12	192.168.2.12	192.168.2.0	255.255.255.0	11000	192.168.3.14	80	SKYPE

Na Figura 28 são apresentados os nós do caminho escolhido pelo controlador de contexto para adaptar o fluxo VoIP entre as estações N12 e N14.

Figura 28 - Seleção dos nós do melhor caminho da 2ª Execução

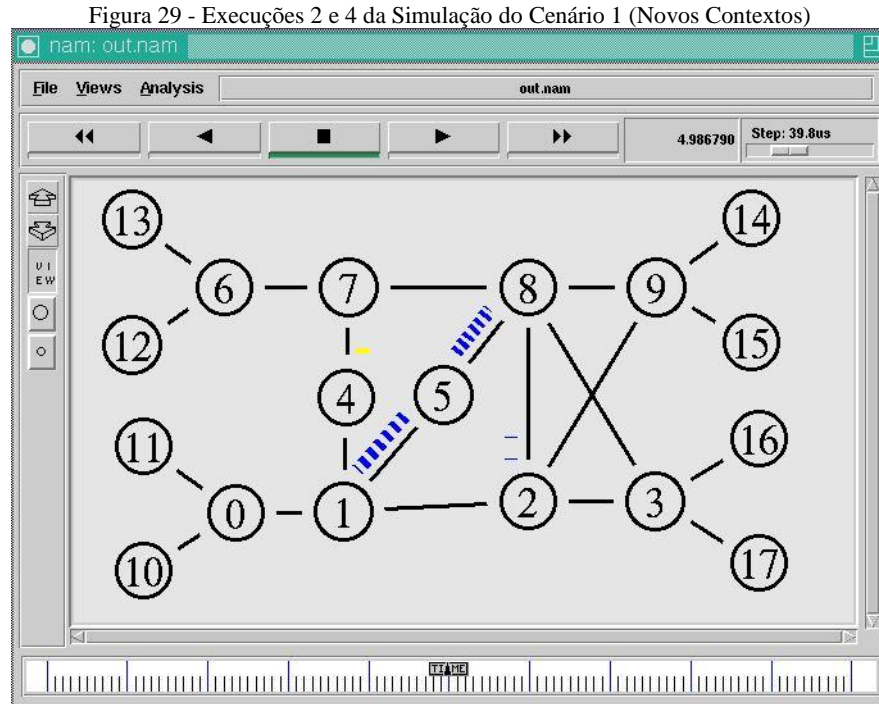
The screenshot shows a SQL query execution window with the following query:

```
1 • select distinct e.dev_curval_txt
2   from
3   (
4   select a.pathid, a.path_delay_sum, a.path_hop_count
5   from path a inner join
6   (
7   select pf.pathid
8   from (select pathid from path_fwr where devid = 6) pf inner join (select pathid from path_fwr where devid = 9) pa on
9   ((pf.pathid = pa.pathid)
10  group by pf.pathid
11  ) b on (a.pathid = b.pathid)
12  order by a.path_delay_sum, a.path_hop_count
13  limit 1
14  ) a inner join path_link      b on (a.pathid = b.pathid)
15  inner join link              c on (b.linkid = c.linkid)
16  inner join link_fwr          d on (c.linkid = d.linkid)
17  inner join deviceparameter e on (d.devid = e.devid and parmidx = 12)
18
```

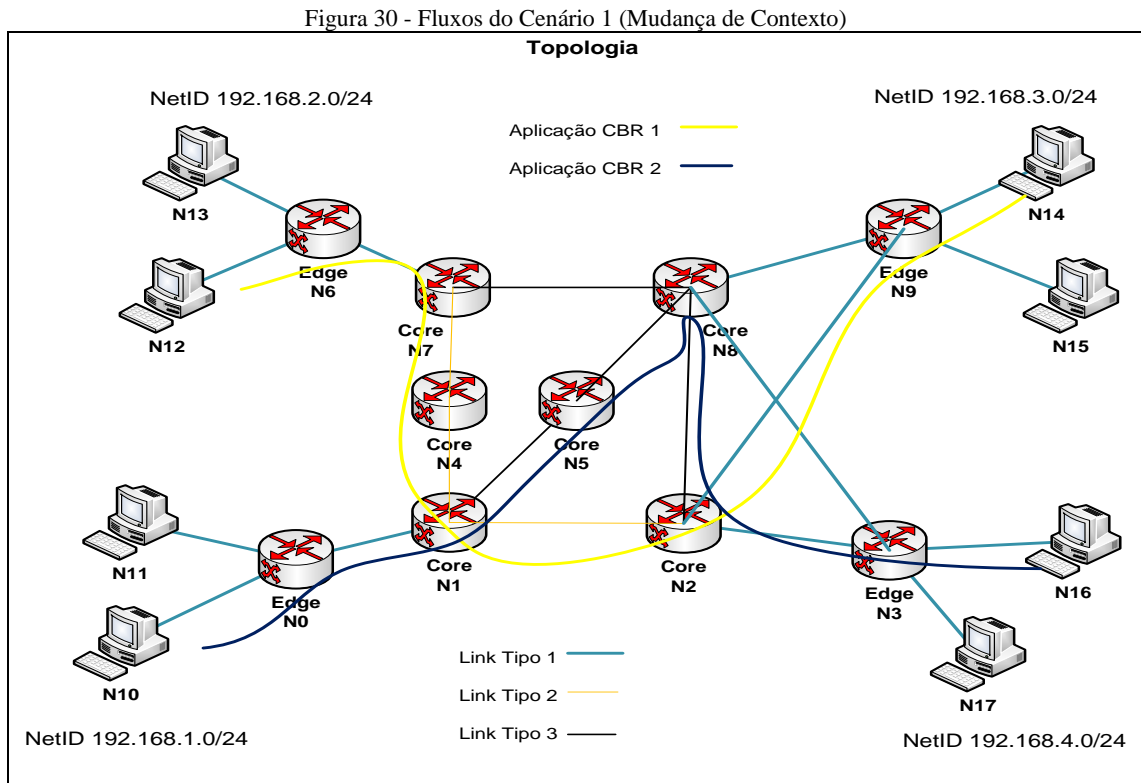
The results are displayed in a table with the following columns and data:

dev_curval_txt
N1
N2
N4
N9
N7
N6

Na Figura 29 é mostrada a execução da simulação com as mudanças indicadas pelo controlador de contexto.



Na Figura 30 é apresentada a ilustração do cenário 1 com os fluxos de VoIP e VoD adaptados conforme analisado e indicado pelo controlador de contexto.



Na Tabela 10 são mostrados alguns parâmetros que foram medidos no fim da segunda execução da simulação. Esses mesmos parâmetros foram apresentados antes da primeira execução da simulação.

Tabela 10 - Resultados do Cenário 1 – Após a 2ª Execução

<b>Aplicação</b>	<b>Transmitidos</b>	<b>Atraso Fim a Fim</b>	<b>Largura de Banda</b>
VoIP	460800 bits	0.008 ms	77.43 Kbps
VoD	10472000 bits	0.024 ms	1818.42Kbps

Depois de todas as execuções realizadas, é apresentada uma avaliação comparando as duas execuções e as mudanças sofridas pelos fluxos que foram analisados. Com os dados apresentados na Tabela 9, e em comparação com os dados da Tabela 10, pode-se perceber que, para o fluxo VoIP, houve uma melhoria em relação ao atraso, que é um importante requisito nas comunicações VoIP (CISCO A, 2015). Para o fluxo VoD, cuja largura de banda é o requisito mais importante (CISCO B, 2015), houve uma expressiva melhora com a mudança do caminho o que permitiu uma maior transmissão de bits e, como isso, utilizou, também, uma maior largura de banda usada por essa aplicação.

#### **Resumo das Execuções Realizadas**

Na Tabela 11 é mostrada a evolução do contexto no Cenário 1. Nessa Tabela são mostrados, em cada momento da evolução do Cenário 1, as entradas recebidas pelo controlador de contexto, qual o processamento realizado e a saída com a escolha do melhor caminho. Exceto, no início de uma relação, os momentos posteriores são avaliados pelo controlador de contexto. Sendo assim, como pode ser visto na 11, somente nos instantes 2 e 4, quando ocorre a coleta periódica, é que a decisão do controlador interfere na mudança de caminho em relação ao fluxo inicial das duas relações existentes, que escolheram inicialmente o caminho de melhor esforço.

Tabela 11 - Cenário 1 - Evolução do Contexto

Execução	Descrição	Instante (min)	Entradas	Evento de Contexto	Caminho
1	Início do Fluxo de CBR1	1	Envio do XML para o Controlador e escolha do caminho inicial	Nova Relação – Sem influência do controlador	CBR1 ->N0 – N1 – N2 – N3
2	Coleta Periódica	2	Envio do XML para o Controlador com mudança do caminho	Mudança de contexto - escolha de um novo caminho de maior largura de banda	CBR1 ->N0 - N1 - N5 - N8 - N2 - N3
3	Início do Fluxo de CBR2	3	Envio do XML para o Controlador e escolha do caminho inicial	Nova Relação – Sem influência do controlador	CBR2 ->N6 – N7 –N8 – N9
4	Coleta Periódica	4	Envio do XML para o Controlador com mudança do caminho do fluxo CBR2 e envio do XML do Fluxo CBR1 sem alteração	Mudança de contexto para o fluxo CBR - escolha de um novo caminho de menor atraso	CBR2 ->N6 - N7 - N4 - N1 - N2 - N9
5	Coleta Periódica	6	Envio dos XMLs para os fluxos CBR2 e CBR1 sem alterações		
6	Coleta Periódica	8	Envio dos XMLs para os fluxos CBR2 e CBR1 sem alterações		

Na Tabela 12 são mostradas as mudanças de contexto e os XMLs correspondentes, bem como os dados inseridos no banco usando a linguagem SQL e os scripts TCL que foram utilizados nas execuções apresentadas.



Tabela 12 - Cenário 1 - Contexto versus XML versus Banco versus TCL

Instante						
Execução	Descrição	(min)	Entradas	XML	SQL	TCL
			Envio do XML para o Controlador e escolha do caminho inicial	ANEXO E	ANEXO F	ANEXO G
1	Início do Fluxo CBR1	1	Envio do XML para o Controlador com mudança do caminho	ANEXO E	ANEXO F	ANEXO G
2	Periódica	2	Envio do XML para o Controlador e escolha do caminho inicial	ANEXO E	ANEXO F	ANEXO G
3	Início do Fluxo CBR2	3	Envio do XML para o Controlador com mudança do caminho do fluxo CBR2 e envio do XML do Fluxo CBR1	ANEXO E	ANEXO F	ANEXO G
4	Coleta Periódica	4	Envio dos XMLs para os fluxos CBR2 e CBR1 sem alterações	ANEXO E	ANEXO F	ANEXO G
5	Coleta Periódica	6	Envio dos XMLs para os fluxos CBR2 e CBR1 sem alterações	ANEXO E	ANEXO F	ANEXO G
6	Coleta Periódica	8	Envio dos XMLs para os fluxos CBR2 e CBR1 sem alterações	ANEXO E	ANEXO F	ANEXO G

Na Tabela 12 continua sendo mostrada a evolução do contexto, porém faz uma associação entre os XMLs recebidos pelo controlador, os comandos SQL que, durante a execução da simulação do Cenário 1, são executados e a mudança no script TCL que será executado no NS-2 para demonstrar o funcionamento da simulação.

### 5.3 SEGUNDO CENÁRIO

Nesse segundo cenário, foi caracterizada uma configuração de simulação com mais carga que no primeiro cenário, para continuar demonstrando que as informações contextuais são, igualmente, analisadas em cenários com mais carga, sem priorizar diferentes circunstâncias em um cenário de rede.

Da mesma forma que no cenário anterior é analisado de acordo com as informações contextuais obtidas, se determinada comunicação entre estações, usando determinada aplicação, pode sofrer mudança de caminho a fim de melhorar a qualidade daquela relação. Com as alternativas de caminho apresentadas, ocorre ou não a mudança de caminho para os fluxos analisados.

Portanto, o cenário apresentado nessa seção é um pouco mais complexo, porém com a mesma importância para o controlador no que diz respeito à análise das informações contextuais para efetivar ou não a mudança de caminho dos fluxos.

Nesse cenário é mantida a carga do cenário 1 com dois fluxos um VoIP e outro VoD e introduz mais dois fluxos VoIP e VoD que serão submetidos à análise do contexto. Sendo assim, existirão mais dois fluxos novos em comparação com o cenário 1 apresentado na seção anterior.

Além disso, é introduzido o aspecto de qualidade de experiência para o tráfego VoIP, onde o parâmetro MOS de sistema, que é calculado pela gerência de contexto, é avaliado antes de fazer a escolha do melhor caminho pela gerência de encaminhamento.

Na Tabela 13 são apresentados os tipos de fluxos que serão tratados no cenário dessa simulação e em que intervalo irão executar:

Tabela 13 - Fluxo e o intervalo de simulação em detalhe

TIPOS DE TRÁFEGO:	CAMINHO INICIAL:	DURAÇÃO
<b>CBR1- UDP (VoIP)</b>	$\{N6 \leftrightarrow N7 \leftrightarrow N8 \leftrightarrow N9\}$	3 a 8.5 min.
<b>CBR2 - UDP(VoD)</b>	$\{N0 \leftrightarrow N1 \leftrightarrow N2 \leftrightarrow N3\}$	1 a 7 min.
<b>CBR3- UDP (VoIP)</b>	$\{N0 \leftrightarrow N1 \leftrightarrow N2 \leftrightarrow N3\}$	4 a 9 min
<b>CBR4 – UDP (VoD)</b>	$\{N9 \leftrightarrow N2 \leftrightarrow N1 \leftrightarrow N0\}$	4 a 9,5 min

Figura 31 - Evolução dos fluxos na simulação (em minutos)

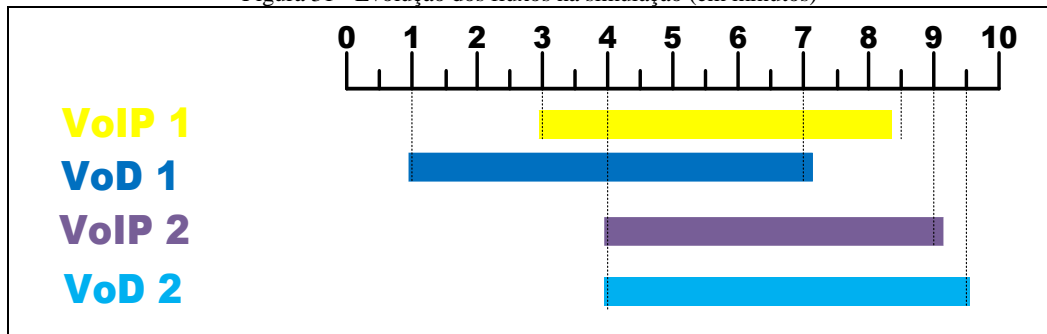


Figura 32 - Fluxos do Cenário 2 (Novas Relações)

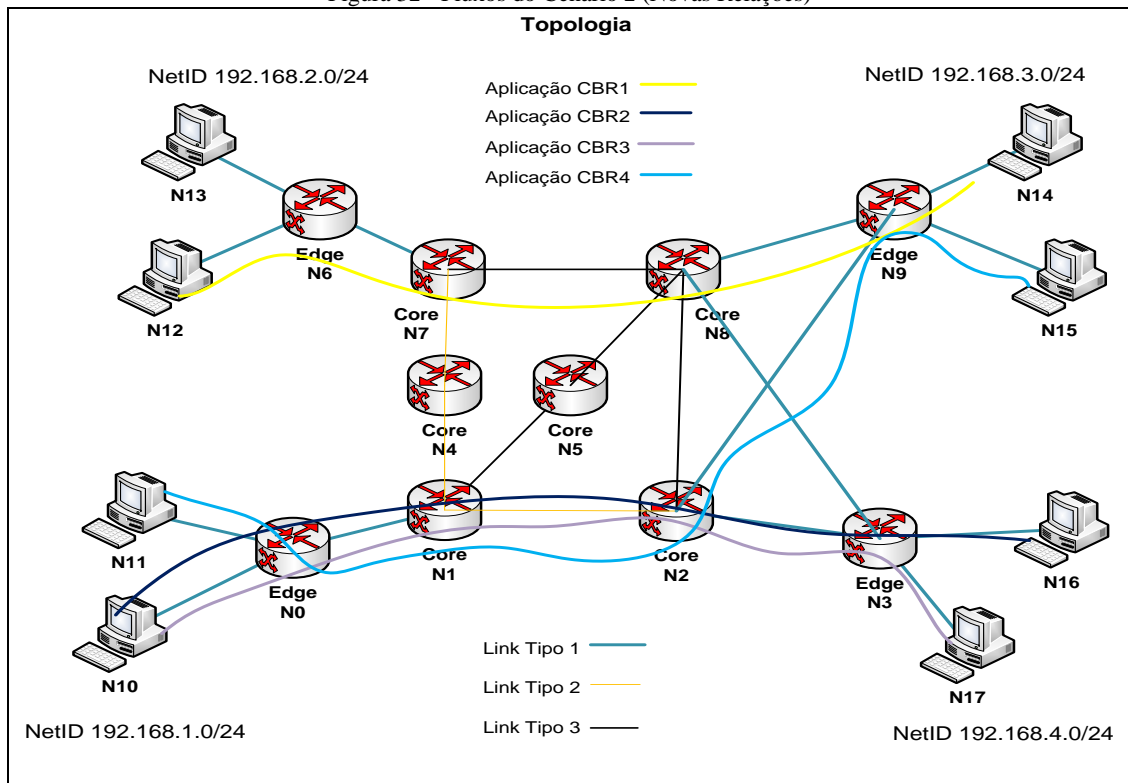


Tabela 14 - Resultados do Cenário 2–Antes da 1ª Execução

<b>Aplicação</b>	<b>Transmitidos</b>	<b>Atraso Fim a Fim</b>	<b>Largura de Banda</b>
VoIP 1	480000 bits	0.011 ms	80.21 Kbps
VoD 1	8648000 bits	0.046 ms	1501.39Kbps

### **1ª Execução**

Conforme citado no início dessa seção, essa primeira execução é semelhante à ocorrida no cenário 1 e, portanto, teve a mesma resposta do controlador de contexto com mudança de caminho para a melhoria da comunicação.

### **2ª Execução**

Da mesma forma, essa segunda execução também é semelhante à ocorrida no cenário 1 e, teve a mesma resposta do controlador de contexto com mudança de caminho para a melhoria da comunicação.

Antes da 3ª Execução,

Tabela 15 - Resultados do Cenário 2–Antes da 3ª Execução

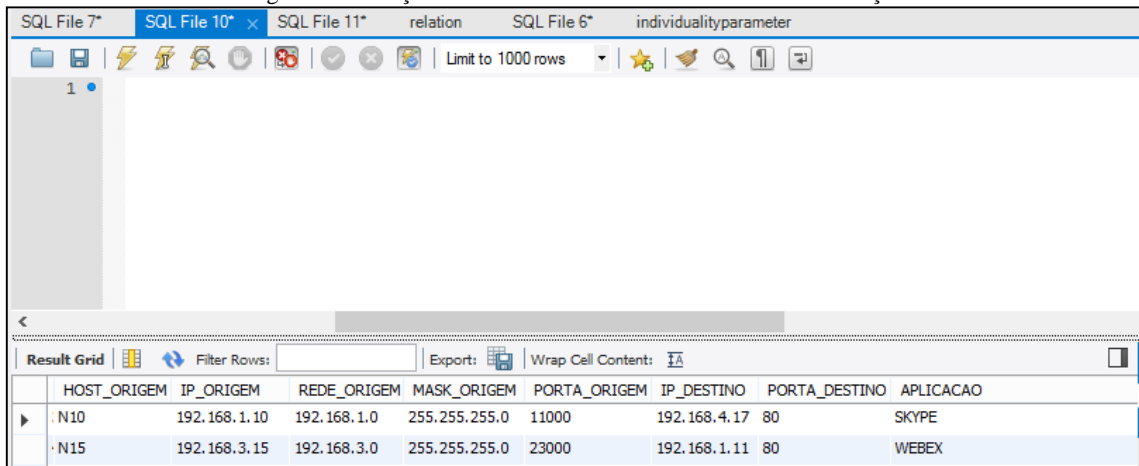
<b>Aplicação</b>	<b>Transmitidos</b>	<b>Atraso Fim a Fim</b>	<b>Largura de Banda</b>
VoIP 2	40960 bits	0.005 ms	8.65 Kbps
VoD 2	8648000 bits	0.046 ms	1501.39 Kbps

### **3ª Execução**

Após consulta na visão de contexto *ContextModelView* dentro do *ContextDatabase*, são obtidos os novos contextos para terem seus caminhos analisados e, caso seja conveniente para uma melhor comunicação, poderão ter seus caminhos alterados de forma dinâmica durante uma relação ativa entre duas (2) máquinas na rede.

Nessa execução, foram identificados dois novos fluxos, conforme mostrado na Figura 32 que iniciaram no mesmo instante e que tiveram mudança no contexto e, portanto, terão suas informações analisadas pelo módulo de gerência de encaminhamento para mudarem ou não de caminho.

Figura 33 - Seleção do contexto no ContextDatabase na 3ª Execução



	HOST_ORIGEM	IP_ORIGEM	REDE_ORIGEM	MASK_ORIGEM	PORTA_ORIGEM	IP_DESTINO	PORTA_DESTINO	APLICACAO
▶	N10	192.168.1.10	192.168.1.0	255.255.255.0	11000	192.168.4.17	80	SKYPE
	N15	192.168.3.15	192.168.3.0	255.255.255.0	23000	192.168.1.11	80	WEBEX

Figura 34 - Execução 3 da Simulação do Cenário 2 (Novos Contextos)

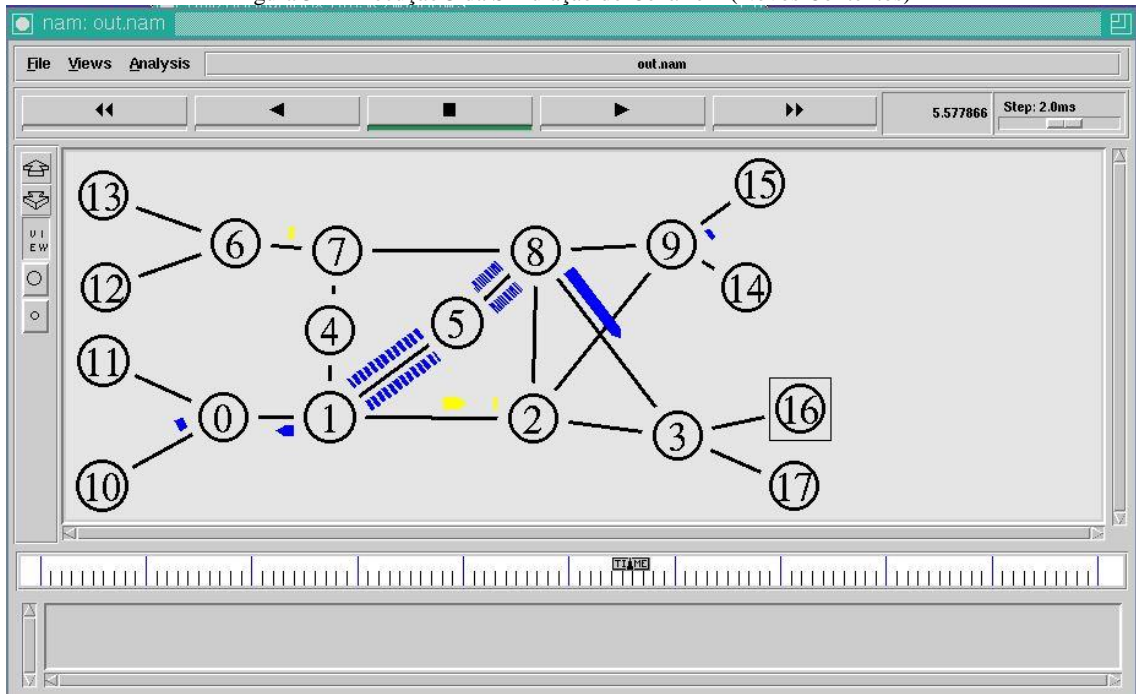


Figura 35 - Fluxos do Cenário 2 (Mudança de Contexto)

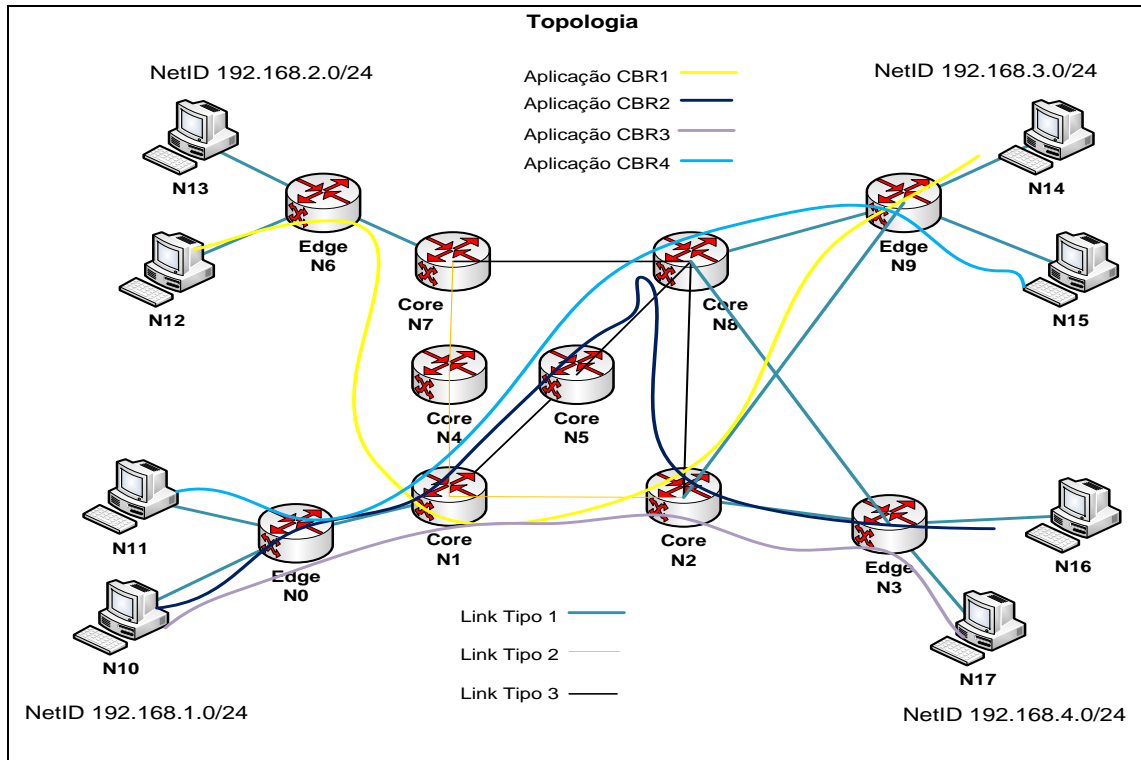


Figura 36 - Seleção dos nós do melhor caminho da 3a Execução para Tráfego CBR3

Query 1 SQL File 5\* SQL File 6\* SQL File 7\* SQL File 10\* SQL File 11\* x

```

1 select distinct e.dev_curval_txt
2 from
3 (
4 select a.pathid, a.path_delay_sum, a.path_hop_count
5 from path a inner join
6 (
7 select pf.pathid
8 from (select pathid from path_fwr where devid = 0) pf inner join (select pathid from path_fwr where devid = 3) pa on
9 (pf.pathid = pa.pathid)
10 group by pf.pathid
11 ) b on (a.pathid = b.pathid)
12 order by a.path_delay_sum, a.path_hop_count
13 limit 1
14 ) a inner join path_link b on (a.pathid = b.pathid)
15 inner join link c on (b.linkid = c.linkid)
16 inner join link_fwr d on (c.linkid = d.linkid)
17 inner join deviceparameter e on (d.devid = e.devid and parmId = 12)
    
```

Result Grid

dev_curval_txt
N0
N1
N2
N3

Figura 37 - Seleção dos nós do melhor caminho da 3ª Execução para Tráfego CBR4

```

1  select distinct e.dev_curval_txt as ROTEADOR
2  from
3  (
4  select a.pathid, a.path_bw_min, a.path_hop_count
5  from path a inner join
6  (
7  select pf.pathid
8  from (select pathid from path_fwr where devid = 0) pf inner join (select pathid from path_fwr where
9  devid = 9) pa on
10  (pf.pathid = pa.pathid)
11  group by pf.pathid
12  ) b on (a.pathid = b.pathid)
13  order by a.path_bw_min desc, a.path_hop_count
14  limit 1
15  ) a inner join path_link      b on (a.pathid = b.pathid)
16  inner join link              c on (b.linkid = c.linkid)
17  inner join link_fwr          d on (c.linkid = d.linkid)
18  inner join deviceparameter  e on (d.devid = e.devid and parmidx = 12)
19

```

Result Grid

ROTEADOR
N0
N1
N5
N8
N9

Tabela 16 - Resultados do Cenário 2 –Após a 3ª Execução

Aplicação	Transmitidos	Atraso Fim a Fim	Largura de Banda
VoIP 2	358400 bits	0.009 ms	75.15 Kbps
VoD 2	10744000 bits	0.022 ms	1920.29 Kbps

Depois de todas as execuções realizadas, é apresentada uma avaliação comparando as três execuções e as mudanças sofridas pelos fluxos que foram analisados. Com os dados apresentados na Tabela 15. Em comparação com os dados da Tabela 16, pode-se perceber que, para o fluxo VoIP, houve uma melhoria em relação ao atraso, que é um importante requisito nas comunicações VoIP (CISCO A, 2015). Para o fluxo VoD, também, ocorreu uma melhora pois, o canal utilizado por ele teve uma diminuição da carga ocorrida pela mudança de outros fluxos que estavam sobrecarregando o mesmo canal. Sendo assim, a escolha feita na gerência de encaminhamento foi a manutenção do mesmo canal.

### **Resumo das Execuções Realizadas**

Na Tabela 17 é mostrada a evolução do contexto no Cenário 2. Nessa Tabela são mostrados, em cada momento da evolução do Cenário 2, as entradas recebidas pelo controlador de contexto, qual o processamento realizado e a saída com a escolha do melhor caminho. Exceto no início de uma relação, os momentos posteriores são avaliados pelo controlador de contexto.

Sendo assim, como pode ser visto na Tabela 17, somente nos instantes 2, 4 e 5, quando ocorre a coleta periódica, é que a decisão do controlador interfere na mudança de caminho em relação ao fluxo inicial de três das quatro relações existentes, que escolheram inicialmente o caminho de melhor esforço.

Tabela 17 - Cenário 2 - Evolução do Contexto

Execução	Descrição	Instante		Entradas	Evento de Contexto	Caminho
		(min)				
1	Início do Fluxo de CBR1	1		Envio do XML para o Controlador e escolha do caminho inicial	Nova Relação CBR1 – Sem influência do controlador	CBR1 -> N0 - N1 - N2 - N3
2	Coleta Periódica	2		Envio do XML para o Controlador com mudança do caminho	escolha de um novo caminho de maior largura de banda	CBR1 -> N0 - N1 - N5 - N8 - N2 - N3
3	Início do Fluxo CBR2	3		Envio do XML para o Controlador e escolha do caminho inicial	Nova Relação CBR2 – Sem influência do controlador	CBR2 -> N6 - N7 - N8 - N9
4	Coleta Periódica			Envio do XML para o Controlador do fluxo CBR1 sem alteração	Mudança de contexto para o fluxo CBR2 -	
5	Coleta Periódica	4		Envio do XML para o Controlador com mudança do caminho do fluxo CBR2	escolha de um novo caminho de menor atraso	CBR2 -> N6 - N7 - N4 - N1 - N2 - N9
6	Início do Fluxo CBR3			Envio do XML para o Controlador com o fluxo CBR3	Nova Relação CBR3 – Sem influência do controlador	CBR3 -> N0 - N1 - N2 - N3
7	Início do Fluxo CBR4			Envio do XML para o Controlador com o fluxo CBR4	Nova Relação CBR4 – Sem influência do controlador	CBR4 -> N9 - N2 - N1 - N0
8	Coleta Periódica	5		Envio do XML para o fluxo CBR4	Mudança de contexto para o fluxo CBR4 - escolha de um novo caminho de maior largura de banda	CBR4 -> N9 - N8 - N5 - N1 - N0
9	Coleta Periódica			Envio do XML para o fluxo CBR3		



Execução	Descrição	Instante		Evento de Contexto	Caminho
		(min)	Entradas		
			Envio do XML para o fluxo		
10	Coleta Periódica		CBR1		
			Envio do XML para o fluxo		
11	Coleta Periódica		CBR2		
12	Coleta Periódica	6			
13	Coleta Periódica	7			

Na Tabela 18 são mostradas as mudanças de contexto e os XMLs correspondentes, bem como os dados inseridos no banco usando a linguagem SQL e os scripts TCL que foram utilizados nas execuções apresentadas.

Tabela 18 - Cenário 2 - Contexto versus XML versus Banco versus TCL

Execução	Descrição	Instante		XML	SQL	TCL
		(min)	Entradas			
			Envio do XML para o		ANEXO G	ANEXO H
	Início do Fluxo de		Controlador e escolha do			
1	CBR1	1	caminho inicial	ANEXO E		
			Envio do XML para o		ANEXO F	ANEXO G
	Coleta		Controlador com mudança			
2	Periódica	2	do caminho	ANEXO E		
			Envio do XML para o		ANEXO F	ANEXO G
	Início do Fluxo		Controlador e escolha do			
3	CBR2	3	caminho inicial	ANEXO V		
			Envio do XML para o		ANEXO F	ANEXO G
			Controlador com mudança do caminho do fluxo			
	Coleta		CBR2,envio do			
4	Periódica	4	XML do Fluxo	ANEXO E		

Execução	Descrição	Instante				
		(min)	Entradas	XML	SQL	TCL
			CBR1 sem alteração, Envio do XML para o Controlador com o fluxo CBR3 e Envio do XML para o Controlador com o fluxo CBR4			
			Envio dos XMLs para os fluxos CBR2, CBR1 e CBR3 sem alterações e Envio dos XMLs para os fluxos CBR4 com mudança de caminho	ANEXO E	ANEXO F	ANEXO G
5	Coleta Periódica	6	Envio dos XMLs para os fluxos CBR2, CBR1, CBR3 e CBR4 sem alterações	ANEXO E	ANEXO F	ANEXO G
6	Coleta Periódica	8		ANEXO E		

Na Tabela 18 continua sendo mostrada a evolução do contexto, porém faz uma associação entre os XMLs recebidos pelo controlador, os comandos SQL que, durante a execução da simulação do Cenário 2, são executados e a mudança no script TCL que será executado no NS-2 para demonstrar o funcionamento da simulação.

#### 5.4 TERCEIRO CENÁRIO

Nesse terceiro cenário, foi caracterizada uma configuração de simulação com a mesma carga que no segundo cenário, para continuar demonstrando que as informações contextuais são, igualmente, analisadas em cenários com mais carga, sem priorizar diferentes circunstâncias em um cenário de rede.

Da mesma forma que no cenário anterior, é analisado, de acordo com as informações contextuais obtidas, se determinada comunicação entre estações, usando determinada aplicação, pode sofrer mudança de caminho, a fim de melhorar a qualidade daquela relação. Com as alternativas de caminho apresentadas, ocorre ou não a mudança de caminho para os fluxos analisados.

Nesse cenário é mantida a carga do cenário 2, conforme já citado, porém o fluxo do segundo VoIP, que sofrera influência do MOS calculado pela gestão de contexto, não é impactado nesse cenário.

A outra mudança no cenário 3 em relação ao cenário 2 é a introdução do aspecto de qualidade de dispositivo (QoD) na comunicação ocorrida no segundo VoD, quando os equipamentos da relação não atendem aos requisitos mínimos de hardware para efetuar o uso, com boa qualidade, desse software de VoD específico. Dessa forma, a impressão do usuário em relação àquela comunicação tende a não mudar, pois o equipamento não está adequado para aquele tipo de comunicação. Sendo assim, seria ineficiente a mudança de caminho dessa relação.

Na Tabela 19 são apresentados os tipos de fluxos que serão tratados no cenário dessa simulação e em que intervalo irão executar:

Tabela 19 - Fluxo e o intervalo de simulação em detalhe

<b>TIPOS DE TRÁFEGO:</b>	<b>CAMINHO INICIAL:</b>	<b>DURAÇÃO</b>
<b>CBR1- UDP (VoIP)</b>	<b>{N6 ↔ N7 ↔ N8 ↔ N9}</b>	<b>3 a 8.5 min.</b>
<b>CBR2 - UDP(VoD)</b>	<b>{N0 ↔ N1 ↔ N2 ↔ N3}</b>	<b>1 a 7 min.</b>
<b>CBR3- UDP (VoIP)</b>	<b>{N0 ↔ N1 ↔ N2 ↔ N3}</b>	<b>4 a 9 min</b>
<b>CBR4 – UDP (VoD)</b>	<b>{N9 ↔ N2 ↔ N1 ↔ N0}</b>	<b>4 a 9,5 min</b>

Figura 38 - Evolução dos fluxos na simulação (em minutos)

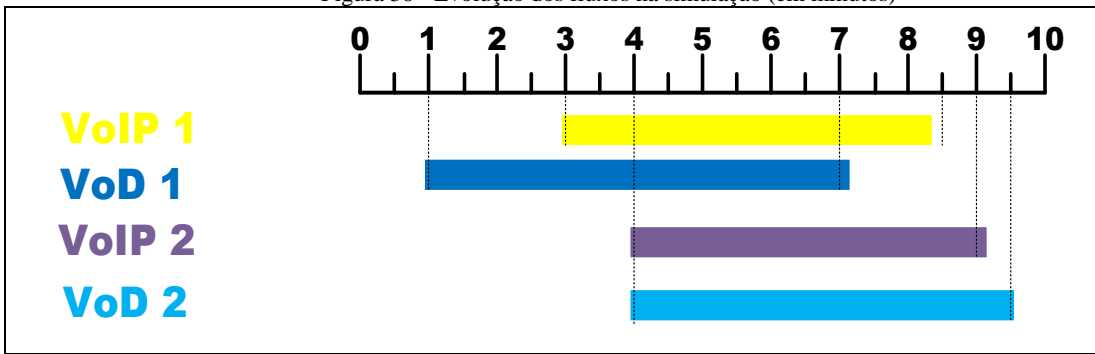
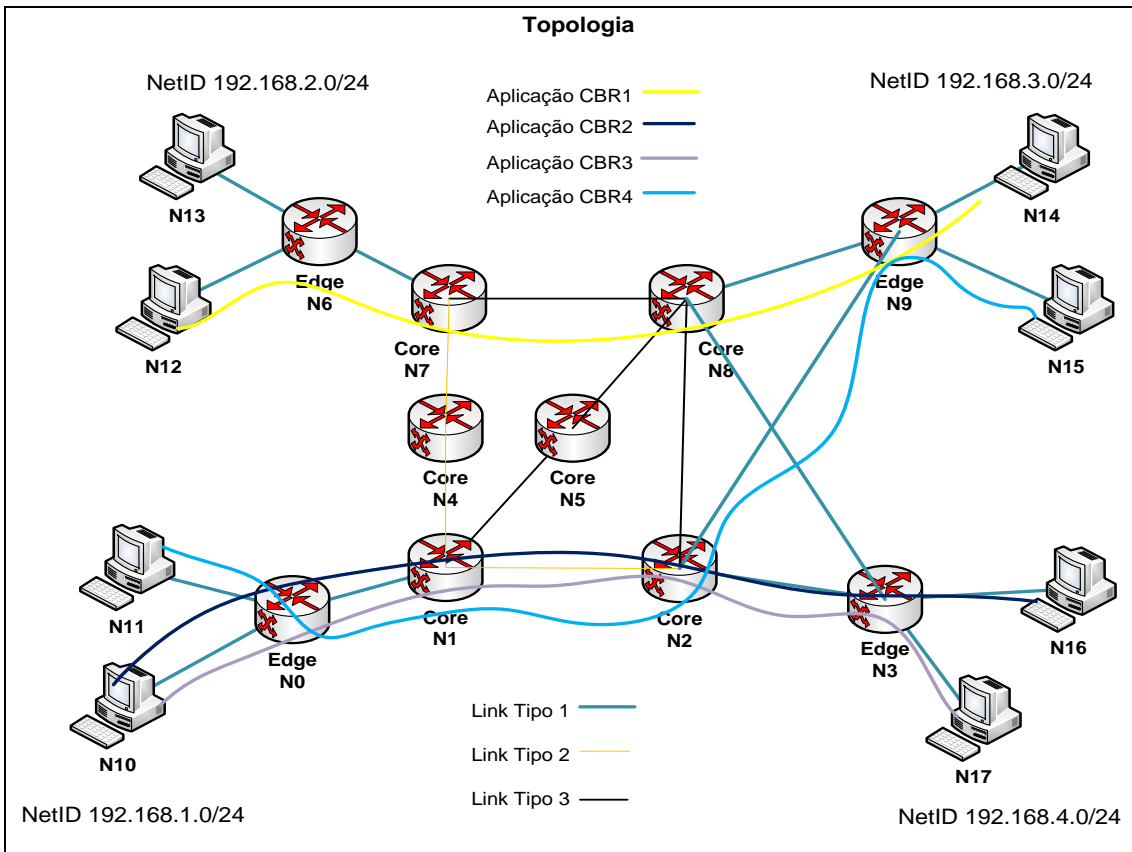


Figura 39 - Fluxos do Cenário 3 (Novas Relações)



**1ª Execução**

Conforme citado no início dessa seção, essa primeira execução é semelhante à ocorrida no cenário 1 e, portanto, teve a mesma resposta do controlador de contexto com mudança de caminho para a melhoria da comunicação.

### 2ª Execução

Da mesma forma, essa segunda execução é semelhante à ocorrida no cenário 1 e, portanto, teve a mesma resposta do controlador de contexto com mudança de caminho para a melhoria da comunicação.

Antes da 3ª Execução, é mostrado na Tabela 20 algumas informações:

Tabela 20 - Resultados do Cenário 2–Antes da 3ª Execução

<b>Aplicação</b>	<b>Transmitidos</b>	<b>Atraso Fim a Fim</b>	<b>Largura de Banda</b>
VoIP 2	40960 bits	0.005 ms	8.65 Kbps
VoD 2	8648000 bits	0.046 ms	1501.39 Kbps

### 3ª Execução

Após consulta na visão de contexto *ContextModelView* dentro do *ContextDatabase*, são os obtidos os novos contextos para terem seus caminhos analisados e, caso seja conveniente para uma melhor comunicação, poderem ter seus caminhos alterados de forma dinâmica durante uma relação ativa entre duas (2) máquinas na rede.

Nessa execução, foram identificados dois novos fluxos, conforme mostrado na Figura 39, que iniciaram no mesmo instante e que tiveram mudança no contexto e, portanto, terão suas informações analisadas pelo módulo de gerência de encaminhamento para mudarem ou não de caminho.

Figura 40 - Seleção dos nós do melhor caminho da 3a Execução para Tráfego CBR3

```

1  select distinct e.dev_curval_txt as ROTEADOR
2  from
3  (
4  select a.pathid, a.path_bw_min, a.path_hop_count
5  from path a inner join
6  (
7  select pf.pathid
8  from (select pathid from path_fwr where devid = 0) pf inner join (select pathid from path_fwr where devid = 0)
9  (pf.pathid = pa.pathid)
10 group by pf.pathid
11 ) b on (a.pathid = b.pathid)
12 order by a.path_bw_min desc, a.path_hop_count
13 limit 1
14 ) a inner join path_link      b on (a.pathid = b.pathid)
15 inner join link              c on (b.linkid = c.linkid)
16 inner join link_fwr          d on (c.linkid = d.linkid)
17 inner join deviceparameter e on (d.devid = e.devid and parmidx = 12)
18

```

ROTEADOR
N0
N1
N5
N3
N8

Figura 41 - Execução 3 da Simulação do Cenário 3 (Novos Contextos)

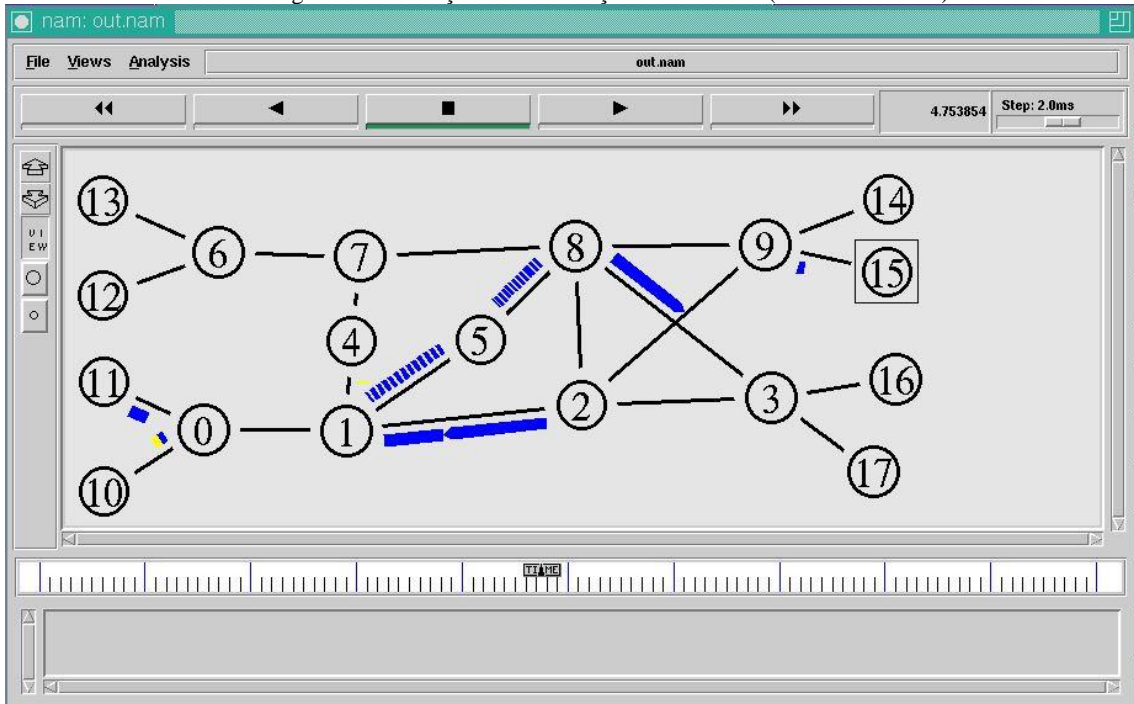


Figura 42 - Fluxos do Cenário 3 (Mudança de Contexto)

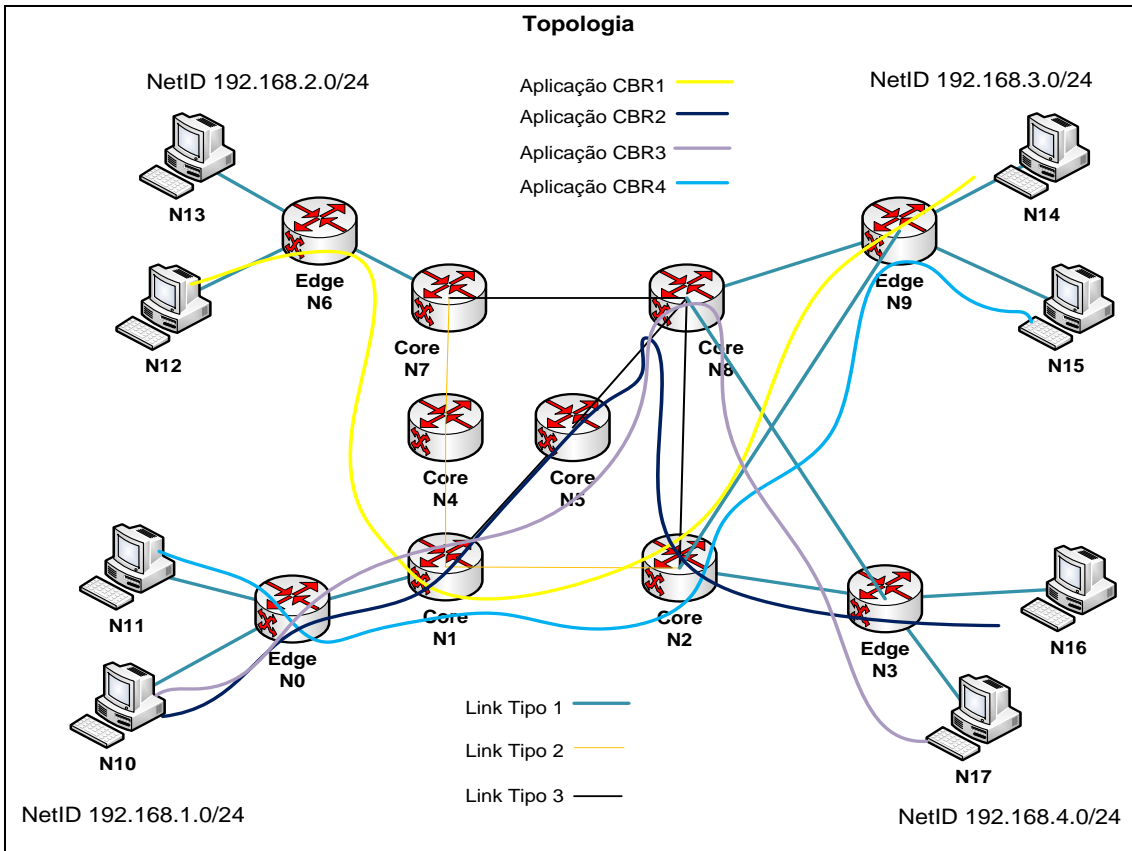


Tabela 21 - Resultados do Cenário 3 – Após a 3ª Execução

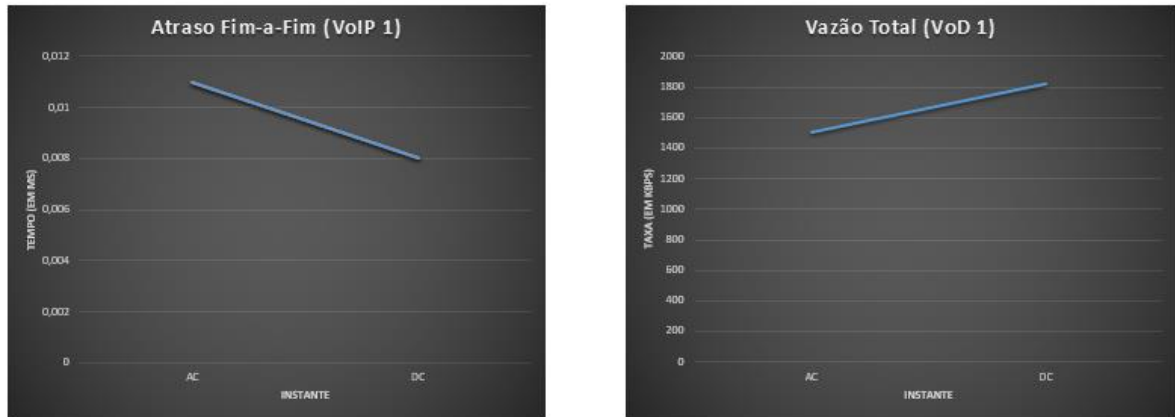
Aplicação	Transmitidos	Atraso Fim a Fim	Largura de Banda
VoIP 2	389120 bits	0.016 ms	80.00 Kbps
VoD 2	8648000 bits	0.046 ms	1501.39 Kbps

Depois de todas as execuções realizadas, é apresentado uma avaliação comparando as três execuções e as mudanças sofridas pelos fluxos que foram analisados. Com os dados apresentados na Tabela 20, e em comparação com os dados da Tabela 21, pode-se perceber que, para o fluxo VoIP, houve uma melhoria em relação à banda por conta da avaliação do QoE feita antes da escolha do melhor caminho. Para o fluxo VoD, não houve mudança de fluxo, como pode ser observado em comparação com os dados das tabelas de resultados.

Conforme pode ser observado nas Figuras 43 e 44, os gráficos mostram a melhoria dos tráfegos depois da mudança de contexto. Pode ser verificado no gráfico do fluxo VoIP que o atraso fim-a-fim diminuiu em 28%. Para o tráfego VoD, teve uma melhora na vazão total em 21%. Para o

tráfego do segundo VoIP, que é sensível à largura de banda, houve uma melhora de 824% na vazão total.

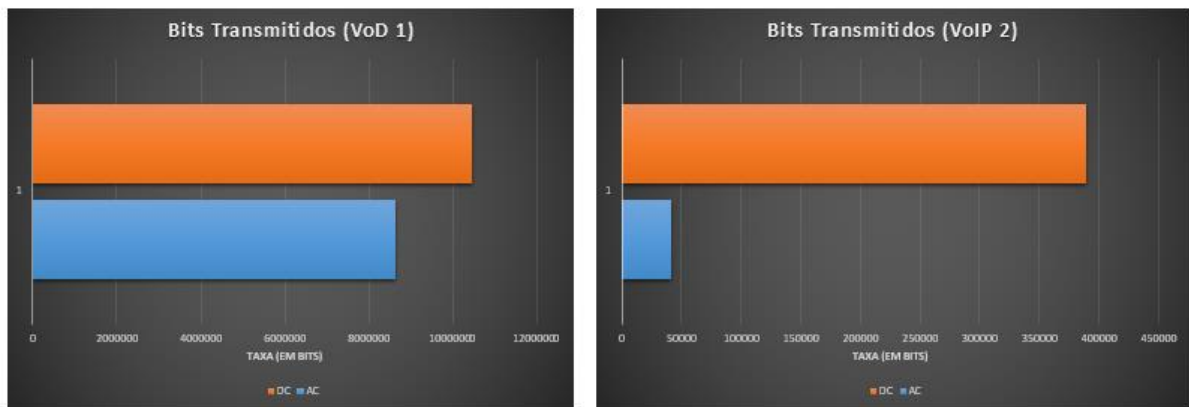
Figura 43 - Gráficos Comparativos (Atraso e Vazão Total)



Melhora de 28% no atraso fim-a-fim

Melhora de 21% na vazão total

Figura 44 - Gráficos Comparativos (Bits Transmitidos totais)



Melhora de 21% na Quantidade de Bits Transmitidos na comunicação total

Melhora de 824 % na Quantidade de Bits Transmitidos na comunicação total

**Resumo das Execuções Realizadas**

Na Tabela 22 **Tabela 22** é mostrada a evolução do contexto no Cenário 3. Nessa Tabela são mostrados, em cada momento da evolução do Cenário 2, as entradas recebidas pelo controlador de contexto, qual o processamento realizado e a saída com a escolha do melhor caminho. Exceto, no início de uma relação, os momentos posteriores são avaliados pelo controlador de contexto. Sendo assim, como pode ser visto na Tabela 22 somente nos instantes 2, 4 e 5, quando ocorre a coleta periódica, é que a decisão do controlador interfere na mudança de caminho em relação ao fluxo inicial de três das quatro relações existentes, que escolheram inicialmente o caminho de melhor esforço.



Tabela 22 - Cenário 3 - Evolução do Contexto

Execução	Descrição	Instante (min)	Entradas	Evento de Contexto	Caminho
1	Início do Fluxo de CBR1	1	Envio do XML para o Controlador e escolha do caminho inicial	Nova Relação CBR1 – Sem influência do controlador Mudança de contexto para o fluxo CBR1 - escolha de um novo caminho de maior largura de banda	CBR1 -> N0 - N1 - N2 - N3
2	Coleta Periódica	2	Envio do XML para o Controlador com mudança do caminho	envio de um novo caminho de maior largura de banda	CBR1 -> N0 - N1 - N5 - N8 - N2 - N3
3	Início do Fluxo CBR2	3	Envio do XML para o Controlador e escolha do caminho inicial	Nova Relação CBR2 – Sem influência do controlador	CBR2 -> N6 - N7 - N8 - N9
4	Coleta Periódica		Envio do XML para o Controlador do fluxo CBR1 sem alteração		
5	Coleta Periódica	4	Envio do XML para o Controlador com mudança do caminho do fluxo CBR2	Mudança de contexto para o fluxo CBR2 - escolha de um novo caminho de menor atraso Nova Relação CBR3	CBR2 -> N6 - N7 - N4 - N1 - N2 - N9
6	Início do Fluxo CBR3		Envio do XML para o Controlador com o fluxo CBR3	– Sem influência do controlador Nova Relação CBR4	CBR3 -> N0 - N1 - N2 - N3
7	Início do Fluxo CBR4		Envio do XML para o Controlador com o fluxo CBR4	– Sem influência do controlador Mudança de contexto para o fluxo CBR3 - escolha de um novo caminho de maior largura de banda	CBR4 -> N9 - N2 - N1 - N0
8	Coleta Periódica		Envio do XML para o fluxo CBR3		CBR3 -> N0 – N1 - N5 – N8 – N3
9	Coleta Periódica		Envio do XML para o fluxo CBR4		-
10	Coleta Periódica	5	Envio do XML para o fluxo CBR1		-
11	Coleta Periódica		Envio do XML para o fluxo CBR2		-
12	Coleta Periódica	6			
13	Coleta Periódica	7			

Na Tabela 23 são mostradas as mudanças de contexto e os XMLs correspondentes, bem como os dados inseridos no banco usando a linguagem SQL e os scripts TCL que foram utilizados nas execuções apresentadas.

Tabela 23 - Cenário 3 - Contexto versus XML versus Banco versus TCL

Execução	Descrição	Instante				
		(min)	Entradas	XML	SQL	TCL
1	Início do Fluxo de CBR1	1	Envio do XML para o Controlador e escolha do caminho inicial	ANEXO E	ANEXO F	ANEXO G
			Envio do XML para o Controlador com mudança do caminho	ANEXO E	ANEXO F	ANEXO G
2	Coleta Periódica	2	Envio do XML para o Controlador e escolha do caminho inicial	ANEXO E	ANEXO F	ANEXO G
			Envio do XML para o Controlador com mudança do caminho do fluxo CBR2, envio do XML do Fluxo CBR1 sem alteração, Envio do XML para o Controlador com o fluxo CBR3 e Envio do XML para o Controlador com o fluxo CBR4	ANEXO E	ANEXO F	ANEXO G
3	Início do Fluxo CBR2	3	Envio do XML para o Controlador com mudança do caminho do fluxo CBR2, envio do XML do Fluxo CBR1 sem alteração, Envio do XML para o Controlador com o fluxo CBR3 e Envio do XML para o Controlador com o fluxo CBR4	ANEXO E	ANEXO F	ANEXO G
			Envio dos XMLs para os fluxos CBR2, CBR1 e CBR4 sem alterações e Envio dos XMLs para os fluxos CBR3 com mudança de caminho	ANEXO E	ANEXO F	ANEXO G
4	Coleta Periódica	4	Envio dos XMLs para os fluxos CBR2, CBR1 e CBR4 sem alterações e Envio dos XMLs para os fluxos CBR3 com mudança de caminho	ANEXO E	ANEXO F	ANEXO G
			Envio dos XMLs para os fluxos CBR2, CBR1, CBR3 e CBR4 sem alterações	ANEXO E	ANEXO F	ANEXO G
5	Coleta Periódica	6	Envio dos XMLs para os fluxos CBR2, CBR1, CBR3 e CBR4 sem alterações	ANEXO E	ANEXO F	ANEXO G
			Envio dos XMLs para os fluxos CBR2, CBR1, CBR3 e CBR4 sem alterações	ANEXO E	ANEXO F	ANEXO G
6	Coleta Periódica	8	Envio dos XMLs para os fluxos CBR2, CBR1, CBR3 e CBR4 sem alterações	ANEXO E	ANEXO F	ANEXO G
			Envio dos XMLs para os fluxos CBR2, CBR1, CBR3 e CBR4 sem alterações	ANEXO E	ANEXO F	ANEXO G

Na Tabela 23 continua sendo mostrada a evolução do contexto, porém faz uma associação entre os XMLs recebidos pelo controlador, os comandos SQL que, durante a execução da simulação

do Cenário 3, são executados e a mudança no script TCL que será executado no NS-2 para demonstrar o funcionamento da simulação.

## 5.5 DISCUSSÃO

Conforme descrito no capítulo 4, nos requisitos funcionais e não funcionais, pode-se destacar, depois dos testes realizados nos 3 cenários apresentados, os seguintes aspectos que puderam ser representados nas simulações, que são:

- Durante as simulações, todos os contextos definidos na gerência de contexto, conforme definido no capítulo 4, já tinham sido disponibilizados com os dados sem qualquer inconsistência ou dados incompletos para avaliação das regras de encaminhamento;
- Em todas as simulações, todos contextos novos e existentes ficaram disponíveis, após a avaliação de contexto para terem suas regras de encaminhamento validadas e seus novos caminhos avaliados;
- A qualquer momento, os administradores de rede podem definir novas regras de encaminhamento sem que haja indisponibilidade do controlador de contexto;
- Todo nó de encaminhamento, conforme definido no capítulo 4, é verificado quando se avalia o melhor caminho. Sendo assim, os nós inativos não são usados na avaliação do melhor caminho, e;
- Após a avaliação do melhor caminho, o submódulo chamado *Flow Adaptation* tem disponível os novos caminhos para serem adaptados na camada de encaminhamento, conforme definido no capítulo 4.

Em relação aos requisitos não funcionais puderam ser verificados os seguintes aspectos:

- Os módulos de encaminhamento foram definidos no arcabouço e implementados na simulação independentes de tecnologia, conforme definido no capítulo 4, independente da tecnologia de rede utilizada. O modelo de dados do arcabouço é suficientemente flexível para se adaptar a qualquer estrutura;
- Da mesma forma que o módulo de encaminhamento, o submódulo *Flow Adaptation* está capacitado a interagir funcionalmente, aplicando os novos caminhos, a qualquer tecnologia de rede, sendo fiel ao modelo de dados definido para o arcabouço;
- O desempenho das funções realizadas pelo módulo de encaminhamento mostrou-se com um tempo de resposta linear mesmo com o aumento do fluxo apresentado nos cenários

com mais carga. Sendo assim, o tempo para a escolha do melhor caminho foi bastante satisfatório conforme mostrado nas Figuras 45 e 46.

Figura 45 - Tempo de Resposta para escolha do melhor caminho

```

1 select distinct e.dev_curval_txt as ROTEADOR
2   from
3   (
4     select a.pathid, a.path_bw_min, a.path_hop_count
5     from path a inner join
6     (
7       select pf.pathid
8       from (select pathid from path_fwr where devid = 0) pf inner join (select pathid from path_fwr where devid = 3) pa on
9        (pf.pathid = pa.pathid)
10      group by pf.pathid
11      ) b on (a.pathid = b.pathid)
12      order by a.path_bw_min desc, a.path_hop_count
13      limit 1
14    ) a inner join path_link      b on (a.pathid = b.pathid)
15      inner join link            c on (b.linkid = c.linkid)
16      inner join link_fwr       d on (c.linkid = d.linkid)
17      inner join deviceparameter e on (d.devid = e.devid and parmId = 12)

```

Query Statistics

**Timing (as measured at client side):**  
Execution time: 0:00:0.00000000

For more details, enable "statement instrumentation" for the Performance Schema and "Query -> Collect Performance Schema Stats".

Figura 46 - Seleção das Relações com mudança de contexto

```

1 select * from context_model_view

```

Query Statistics

**Timing (as measured at client side):**  
Execution time: 0:00:0.01600000

For more details, enable "statement instrumentation" for the Performance Schema and "Query -> Collect Performance Schema Stats".

- A escalabilidade da estrutura foi mostrada com o aumento da carga, mesmo que tenha sido pequena, mas que não houve mudança de tempo expressiva entre os cenários de baixa carga e os de maior carga e avaliação de outros aspectos de experiência e dispositivo, e;
- A estrutura desenhada do controlador de contexto foi projetada para permitir adaptações sem que haja grandes mudanças na estrutura de funcionamento do controlador. Sendo assim, caso haja, novas regras a serem implementadas ou mesmo novos parâmetros a serem inseridos para avaliação, o modelo de dados do controlador permite tal adaptação sem mudança do modelo de dados.

## 5.6 CONCLUSÃO

A partir dos Estudos de caso apresentados nesse capítulo pode-se provar, além dos requisitos de QoS, que aspectos relacionados a QoE e QoD pudessem influenciar efetivamente a avaliação do melhor caminho, a ponto de resultar na mudança de determinado fluxo e a melhoria de experiências de usuário na rede.

Em todos os três cenários apresentados nesse capítulo, a mudança de contexto ocorreu, motivado por aspectos diferentes, mas se comportou efetivamente, quando necessário, com a mudança para um melhor caminho, conforme mostrado nos resultados finais de largura de banda e atraso dos cenários.

Portanto, além dos parâmetros de QoS se mostrarem novamente efetivos na mudança de contexto, fato apresentado no cenário 1, o cenário 2 mostrou a interação de parâmetros de QoS e QoD e no último cenário, foi mostrada a interação de todos os parâmetros citados nesse trabalho. Dessa forma, pode ser provado, que os parâmetros de QoS, QoD e QoE, em conjunto podem ser eficazes na avaliação do melhor caminho para determinado fluxo.

## 6 CONCLUSÕES E PERSPECTIVAS FUTURAS

Neste último capítulo são apresentadas as principais conclusões que puderam ser observadas nesse projeto de mestrado. Em seguida, são propostas algumas perspectivas futuras para a continuação deste trabalho.

### 6.1 CONCLUSÕES

A partir do que foi desenvolvido nesse trabalho foi possível entender melhor os requisitos dos usuários, não somente em relação ao comportamento da aplicação como, também, relacionado aos requisitos dos dispositivos durante a comunicação. Além disso, foi possível avaliar os três grupos de parâmetros em conjunto para a escolha do melhor caminho.

Dessa forma, nesse trabalho foi possível evidenciar os impactos dos parâmetros de QoD (capacidade da CPU e quantidade de memória da estação) e QoE (MOS) durante uma comunicação. Além do QoS, já bastante utilizado, a avaliação de aspectos de QoD e QoE mostraram-se efetivos e úteis na escolha do melhor caminho, proporcionando uma melhor experiência para o usuário no uso de determinadas aplicações e, também, uma melhor condição de uso da rede para todos os usuários.

Inicialmente, através da descrição de um modelo de contexto, foi possível apresentar, conceitualmente, a proposta a ser implementada e que permitisse a interação de requisitos de QoS, QoD e QoE sendo validados por políticas de QoC.

Posteriormente, através da modelagem do arcabouço genérico CAARF (*Context-Aware Adaptive Routing Framework*) apresentado nesse trabalho, foi possível apresentar o modelo de contexto de maneira que pudesse ser implementado, descrevendo tecnicamente a interação entre esses três conjuntos de parâmetros (QoS, QoD e QoE). Dessa forma, possibilitando implementar tanto a gerência de contexto, realizando a análise de tempos em tempos do contexto baseado nas regras de QoC, quanto a gerência de encaminhamento para a obtenção do melhor caminho e otimização de recursos, através da avaliação das regras de encaminhamento definidas no arcabouço.

Através das simulações, foi possível provar a eficácia da gerência de encaminhamento na tomada das decisões de melhor caminho dentro do arcabouço. Foi possível com as simulações,

realizadas em diferentes cenários, que fossem avaliadas essas diferentes situações e pudesse avaliar a resposta do arcabouço diante do ocorrido.

Durante a realização do estudo de caso, com a evolução dos cenários de simulação, foi possível avaliar inicialmente a melhoria do contexto somente considerando os parâmetros de QoS, com efetiva mudança de caminho. Posteriormente, durante o segundo cenário, foi possível validar o arcabouço avaliando conjuntamente os parâmetros de QoS e QoE, obtendo, também, uma efetiva mudança de caminho. E, no último cenário, foram avaliados em conjunto, conforme objetivo desse trabalho, os três grupos de parâmetros, QoS, QoD e QoE, permitindo também, a mudança de caminho.

Portanto, com os resultados obtidos nos três cenários, usando os parâmetros de QoS, QoD e QoE, os mesmos puderam ser validados como tendo influência efetiva na escolha do melhor caminho e em proporcionar melhor experiência e melhor condição de uso da rede por parte dos usuários.

Os estudos de casos usados para a prova do funcionamento do arcabouço, apesar de serem cenários propostos, serviram como ambiente de validação do contexto e, mesmo com o aumento da complexidade, o arcabouço mostrou um comportamento esperado e com um tempo de resposta na escolha dos caminhos satisfatório.

Esse fato mostra que o arcabouço poderá ter respostas positivas, mesmo com um aumento de fluxo considerável de dados, pois os tempos obtidos nas consultas, feitas ao banco de dados para a escolha do melhor caminho, foram satisfatórios.

Um aspecto que se mostrou bastante eficiente no arcabouço apresentado nesse trabalho foi a capacidade de adaptação da estrutura do controlador para a adição de mais parâmetros de avaliação do contexto de rede. É possível, portanto, que um novo parâmetro possa ser incluído para avaliação durante o funcionamento do controlador sem causar indisponibilidade. Além disso, a adição de novas regras, também, é possível sem necessidade de parar o controlador do contexto. Com isso a flexibilidade e adaptabilidade proposta pelo arcabouço para a adaptação de caminho também são extensíveis à própria estrutura do arcabouço.

Portanto, o estudo do roteamento adaptativo continua sendo um assunto extremamente importante e explorado pelos vários pesquisadores do mundo, pois procura, através de inúmeras soluções, entregar para o usuário final a melhor qualidade possível para uma comunicação sem

que isso afete negativamente outros usuários na rede e buscando conceder um canal o mais democrático em termos de qualidade.

## 6.2 PERSPECTIVAS FUTURAS

Nessa seção são apresentadas algumas perspectivas futuras que foram identificadas e que podem ser objeto de estudo de pesquisadores interessados.

Dentre o que foi identificado temos:

- Simulação do módulo de gestão de encaminhamento no simulador NS3;
- Implementação do módulo de gestão de encaminhamento;
- Implementação da interação do Controlador de Contexto, dentro da gestão de encaminhamento, com a camada de roteamento;
- Análise do perfil das mudanças de encaminhamento através dos dados obtidos de banco;
- Adaptação dos protocolos de roteamento com a gestão de encaminhamento do controlador de contexto para interação na melhor escolha do caminho para determinados fluxos;
- Implementação da interação do Controlador de Contexto, dentro da gestão de encaminhamento, com a camada de comutação;
- Implementação da interação do Controlador de Contexto, dentro da gestão de encaminhamento, com a camada de aplicação, permitindo encaminhamento por nome de estação, e;
- Implementação da interação do Controlador de Contexto, dentro da gestão de encaminhamento, para a gestão de filas.



## REFERÊNCIAS

- AGBOMA, F.; LIOTTA, A. QoE-aware QoS management. In: INTERNATIONAL CONFERENCE ON ADVANCES IN MOBILE COMPUTING AND MULTIMEDIA, 6., 2008. **Proceedings....** 2008.
- ALI, Rashid J. Al et al. G-QoS: Grid service discovery using QoS properties. **Computing and Informatics**, v. 21, n. 4, p. 363–382, 2012.
- ALRESHOODI, Mohammed; WOODS, John. Survey on QoE\ QoS correlation models for multimedia services. **arXiv preprint arXiv:1306.0221**, 2013.
- AWDUCHE, D. et al. “**Requirements for Traffic Engineering over MPLS**”. [S.l.]: Internet RFC 2702, 1999.
- BRADEN, Robert; CLARK, David; SHENKER, Scott. **Integrated services in the internet architecture: an overview**. [S.l.]: [s.n.], 1994.
- BUCHHOLZ, T.; KÜPPER, A.; SCHIFFERS, M. Quality of Context: What It Is and Why We Need It. In: INTERNATIONAL WORKSHOP OF THE HP OPENVIEW UNIVERSITY ASSOCIATION (HPOVUA), 10., 2003. **Proceedings...** p.1-14, 2003.
- CAO, Z.; WANG, Z.; ZEGURA, E. W. Performance of hashing-based schemes for Internet load balancing. In: IEEE INFOCOM CONFERENCE, 2000. **Proceedings...** p.1-14, 2000.
- CISCO A. **Voice Over IP - Per Call Bandwidth Consumption**. Disponível em: <<http://www.cisco.com/c/en/us/support/docs/voice/voice-quality/7934-bandwidth-consume.pdf>>. Acesso em: 3 jun. 2015.
- CISCO B. **Cisco WebEx Network Bandwidth - White Paper**. Disponível em: <[http://www.cisco.com/c/en/us/products/collateral/conferencing/webex-meeting-center/white\\_paper\\_c11-691351.pdf](http://www.cisco.com/c/en/us/products/collateral/conferencing/webex-meeting-center/white_paper_c11-691351.pdf)>. Acesso em: 1 jul. 2015.
- CHEN, G.; KOTZ, D. **A survey of context-aware mobile computing research**. Technical Report. Hanover, NH, USA: Dartmouth College, 2000.
- CHIOCCETTI, R. et al. INFORM: a dynamic Interest FORWARDING Mechanism for Information Centric Networking. [S.l.]: [s.n.], 2013.
- CHOWDHURY, R.; BHANDARKAR, P.; PARASHAR, M. Adaptive QoS Management for Collaboration in Heterogeneous Environments; Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS 2002), communication ecosystem. **Communications Magazine, IEEE**, v.50, n.4, p.58-65, 2000.
- DEORA, V. et al. A Quality of Service Management framework Based on User Expectations. In: INTERNACIONAL CONFERENCE ON SERVICE ORIENTED COMPUTING (ICSOC03), 5., 2003. **Proceedings...** 2003.
- DEY, A. K. **Providing Architectural Support for Building Context-Aware Applications**. 2000. Thesis. (Presented to The Academic Faculty)- Georgia Institute of Technology., 2000.

- DEY, A. K.; ABOWD, G. D.; SALBER, D. A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. **Journal Human-Computer Interaction archive**, v. 16, n.2, p. 97-166, 2001.
- EMMANOUILIDIS, C.; KOUTSIAMANIS, R.-A.; TASIDOU, A. Mobile guides: Taxonomy of architectures, context awareness, technologies and applications. **Journal of Network and Computer Applications**, 2012.
- ELWALID, A. et al. **MATE**: MPLS Adaptive Traffic Engineering. [S.l.]: [s.n.], 2001.
- FISCHER, S.; KAMMENHUBER, N.; FELDMANN, A. REPLEX - Dynamic Traffic Engineering Based on Wardrop Routing Policies. In: ACM CONEXT CONFERENCE. 2006. **Proceedings...** 2006.
- FORTZ, B.; REXFORD, J.; THORUP, M. Traffic engineering with traditional IP routing protocols. **IEEE Communications Magazine**, 2002.
- HONG, D. W.; HONG, C. S. A. QoS management framework for distributed multimedia system. **International Journal Of Network Management**, v.13, p.115–127, 2003. (DOI: 10.1002/nem.465)
- KANDULA, S. et al. **Walking the Tightrope**: responsive yet stable. [S.l.]: [s.n.], 2005.
- KARTHIGA, S; BALAMURUGAN, M. S. **Traffic engineering system based on adaptive multipath routing**. [S.l.]: [s.n.], 2013.
- KIM, Y. ; LEE, K. A quality measurement method of context information in ubiquitous environments. In: HYBRID INFORMATION TECHNOLOGY, 2006. ICHIT'06. INTERNATIONAL CONFERENCE ON. IEEE, 2006. **Proceedings...** 2006.
- KRAUSE, M.; HOCHSTATTER, I. **Challenges in Modelling and Using Quality of Context (QoC)**. Mobility Aware Technologies and Applications: LNCS, 3744, 2005. p.324–333.
- KRAUSE, Michael; HOCHSTATTER, Iris. Challenges in modelling and using quality of context (qoc). In: MOBILITY aware technologies and applications. Springer Berlin Heidelberg, 2005. p. 324-333.
- KUSMIEREK, E. et al. An Integrated Network Resource and QoS Management Framework. In IEEE WORKSHOP ON IP OPERATIONS AND MANAGEMENT (IPOM), 2002, Dallas, USA, 2002. **Proceedings...** 2002. p. 68-72.
- LAGHARI, K. U. R.; CONNELLY, K. Toward total quality of experience: A QoE model in a communication ecosystem. **Communications Magazine, IEEE**, v.50, n.4, p.58-65, 2012.
- LEINER, Barry M. et al. The past and future history of the Internet. **Communications of the ACM**, v. 40, n. 2, p. 102-108, 1997.
- LU, G. **Communication and Computing for Distributed Multimedia Systems**. Norwood: Artech House, 1996.
- LUO REN, L.; JON TONG SENG, Q. Towards context information refinement for proximity mobile service using quality of context. In: INTERNATIONAL CONFERENCE ON MOBILE. 2009. **Proceedings...** 2009.

MASCOLO, C.; MUSOLESI, M. SCAR: Context aware Adaptive Routing in Delay Tolerant Mobile Sensor Networks. In: INTERNATIONAL CONFERENCE ON WIRELESS COMMUNICATIONS AND MOBILE COMPUTING. IWCMC '06. 2006. **Proceedings...** 2006.

MELLOUK, A. **Quality of service mechanisms in next generation heterogeneous networks**. [S.l.]: Wiley-IEEE Press., 2008.

MITRA, K.; ZASLAVSKY, A.; AHLUND, C. A probabilistic context-aware approach for quality of experience measurement in pervasive systems. In: ACM SYMPOSIUM ON APPLIED COMPUTING, 2011. **Proceedings...** 2011. p. 419-424.

MYSQL COMMUNITY EDITION. Disponível em: <<http://www.mysql.com/products/community/>>. Acesso em: 3 jun. 2015.

NAM: Network Animator. Disponível em: <<http://www.isi.edu/nsnam/nam/>>. Acesso em: 3 jun. 2015.

NAZARIO, D. C.; DANTAS, M. A. R.; TODESCO, J. L. Taxonomia das publicações sobre Qualidade de Contexto. **Sustainable Business International Journal**, n.20, 2012.

NSNAM. Disponível em: <<http://sourceforge.net/projects/nsnam/>>. Acesso em: 3 jun. 2015.

OLIVEIRA, A. **Mecanismo de notificação baseado em contexto para encaminhamento adaptativo**. 2015. Dissertação (Mestrado)- Universidade Salvador (UNIFACS), Salvador, 2015.

OLIVEIRA, A. L. C. et al. Context-aware framework for adaptive routing. In: INTERNATIONAL WORKSHOP ON ADVANCES IN ICP INFRASTRUCTURES AND SERVICES, 3., 2014, Miami, EUA. **Proceedings...** 2014.

PETZ, A. et al. An architecture for context-aware adaptation of routing in delay-tolerant networks. In: EXTREME CONFERENCE ON COMMUNICATION, 4., 2012, Zurich, Switzerland. **Paper...** 2012.

PATEL, Chintan; SUPEKAR, Kaustubh; LEE, Yugyung. A QoS oriented framework for adaptive management of web service based workflows. In: DATABASE and Expert Systems Applications. Berlin: Springer Berlin Heidelberg, 2003. p. 826-835.

PFLEEGER, S. L. **Engenharia de software: teoria e prática**. 2.ed. . [S.l.]: Prentice Hall do Brasil, 2004.

ROBERTSON, S.; ROBERTSON, C. **Mastering requirements process**. 2nd ed. . [S.l.]: [s.n.], 2006.

ROSEN, E. et al. **Multiprotocol label switching architecture, internet draft**. Disponível em: <draft-ietf-mpls-arch-05.txt> Acesso em: 22 abr. 1999.

SCHUSTER, D.; ROSI, A.; MAMEI, M. et al. Pervasive Social Context-Taxonomy and Survey. **ACM Transactions on Intelligent Systems and Technology**, v.9, n. 4, p. 1-22, 2012.

SILVA, J. P. S. **Implementação de uma arquitetura para o arcabouço Context-Aware Adaptive Routing Framework (CAARF)**. 2015. Dissertação (Mestrado)-Universidade Salvador – UNIFACS. Salvador, 2015.

SKORIN-KAPOV, L. ; VARELA, M. A multi-dimensional view of QoE: the ARCU model. In: MIPRO, INTERNATIONAL CONVENTION, 35., 2012. **Proceedings...** 2012.

SOMMERVILLE, I. **Software requirements, software engineering**. 8th ed. [S.l.]: [s.n.], 2008.

THE NETWORK Simulator - ns-2. Disponível em: <<http://www.isi.edu/nsnam/ns/>>. Acesso em: 3 jun. 2015.

VIANA, W. et al. A Semantic Approach and a Web Tool for Contextual Annotation of Photos Using Camera Phones. In: WISE, 2007. **Proceedings...** 2007.

VIEIRA, V.; SALGADO, A. C.; TEDESCO, P. C. A. R. Modelos e Processos para o Desenvolvimento de Sistemas Sensíveis ao Contexto. In: JAI - JORNADAS DE ATUALIZAÇÃO EM INFORMÁTICA, 28., 2009, Bento Gonçalves, RS. **Anais...** 2009.

ZIMMERMANN, A.; LORENZ, A.; OPPERMANN, R. An Operational Definition of Context. In: INTERNATIONAL AND INTERDISCIPLINARY CONFERENCE ON MODELING AND USING CONTEXT (CONTEXT'07), 6., 2007. **Proceedings...** 2007.

WANG, N.; HO, K. H.; PAVLOU, G. AMPLE: An Adaptive Traffic Engineering System Based on Virtual Routing Topologies. **Communications Magazine, IEEE**, v.50, n.3, 2012.

WEISER, M. The computer for the 21st century. **Scientific American**, v. 265, n. 3, p. 94-104, 1991.

WENNING, B.; TIMM-GIEL, A.; GÖRG, C. A generic framework for context-aware routing and its implementation in wireless sensor networks. In: ITG-Fachbericht-Mobilkommunikation-Technologien und Anwendungen. [S.l.]: [s.n.]: 2009.

YASAR, A.; PREUVENEERS, D.; BERBERS, Y. Evaluation Framework for Adaptive Context-aware Routing in Large Scale Mobile Peer-to-peer Systems. **Peer-to-Peer Networking and Applications**, v. 4, n. 1, p. 37-49, 2010.

YERIMA, S. Implementation and evaluation of measurement based admission control schemes within a converged networks QoS management framework. **International Journal of Computer Networks & Communications**, v. 3, n. 4, p. 137-152, 2011.

XIE, H. et al. **On self adaptive routing in dynamic environments**: an evaluation and design using a simple. [S.l.]: Probabilistic Scheme, 2005.

XML Tutorial. Disponível em: <<http://www.w3schools.com/xml/>>. Acesso em: 3 jun. 2015.

ZHAO, W.; OLSHESKI, D.; SCHULZRINNE, H. **Internet Quality of Service**: an overview. Relatório Técnico CUCS-003-00. [S.l.]: Columbia University, 2000.

**APÊNDICE A - PUBLICAÇÕES DO AUTOR**

SILVA, J. P. S.; OLIVEIRA, A.; MUAHAD, F.; SPINOLA, S. *Context-Aware Framework for Adaptive Routing*. In: International Workshop on ADVANCES in ICT Infrastructures and Services, (ADVANCE 2014), Miami, USA. September 2014.

SILVA, J. P. S.; OLIVEIRA, A.; MUAHAD, F.; SPINOLA, S. *Providing Adaptive Traffic Routing Based on User and Network Context*. Submetido para publicação.

SILVA, J. P. S.; OLIVEIRA, A.; MUAHAD, F.; SPINOLA, S. *Adaptive Traffic Routing Based on Context*. Submetido para publicação.

## ANEXO A – REGRAS DE ENCAMINHAMENTO

<b>Informação Contextual</b>	<b>Agente de Coleta</b>	<b>Evento</b>	<b>Regras</b>
QoE / QoD / QoS	Context Management	Mudança de Contexto	Verificar a cada verificação no repositório de contexto se existe solicitação do usuário
QoD	Link	Indisponibilidade e	Verificar a cada coleta de dados dos encaminhadores se existe disponibilidade do link
QoS	Link	Parâmetros de rede	Verificar a cada coleta de dados dos encaminhadores se estão mantidos os parâmetros de rede como atraso, banda, jitter
QoD	Link	Sobrecarga	Verificação feita a cada mudança de contexto
QoD	Router	Timeout	Verificar de tempos em tempos se a comunicação com os routers está intermitente
QoD	Router	Indisponibilidade e	Verificar de tempos em tempos se os routers estão ativos. Depois de várias verificações que o dado é invalidado no banco
QoS	Router	Mudança de Capacidade	Verificar de tempos em tempos se os routers ativos continuam com a mesma capacidade de processamento
QoD	Router	Overhead de Processamento	Verificar de tempos em tempos se os routers ativos continuam com a mesma capacidade de processamento
QoD	Router	Falha de Hardware	Verificar de tempos em tempos se os routers ativos continuam com a mesma disponibilidade de recursos de hardware

## ANEXO B – EVENTOS DO ARCABOUÇO

### Context Handler

- Recebimento ativação de Agente;
- Recebimento de desligamento normal de Agente;
- Recebimento de *Keep Alive*;
- Recebimento Nova Relação;
- Recebimento Atualização de Relação por coleta periódica;
- Recebimento Atualização de Relação pelo Usuário;
- Recebimento Encerramento de Relação;
- Recebimento Atualização do estado de caminho;
- Recebimento Atualização de parâmetros de caminho;

### Notificação

- Gera notificação de *Keep Alive* das relações ativas para *Model*;
- Gera notificação de desligamento normal de Agente;
- Gera notificação de Nova relação;
- Gera notificação de Atualização de relação, seja por usuário ou outra;
- Gera notificação de Encerramento de relação;
- Gera notificação de Atualização de estado do caminho ou mudança de parâmetros do caminho;
- Notificação da inclusão de novas notificações a serem tratadas;

### Outros Eventos

- Gravação dos Eventos relevantes no *Context Database*;
- Notificação de *Overload* para o *Context Model Mangement*
- Leitura das informações contextuais notificadas pelos *Handlers* nas bases *Context Database locais*;
- Salva no *Context Model Database*;
- Aplicação dos Filtros de QoC
- Mudança de Status dos eventos no *Context Model Database*;
- Desativação do registro de contexto anteriormente utilizado;

### ***Context Model Management***

- Eventos de Notificação;
  - Notificação da inclusão de novas notificações a serem tratadas;
- Outros Eventos;
  - Leitura das informações contextuais notificadas pelos *Handlers* nas bases *Context Database* locais;
  - Salva no *Context Model Database*;
  - Aplicação dos Filtros de QoC
  - Mudança de Status dos eventos no *Context Model Database*;
  - Desativação do registro de contexto anteriormente utilizado.

### ***Context-based Forwarding Management***

- Eventos de Recebimento;
  - Recebimento da inclusão de novas notificações a serem tratadas;
- Eventos de Notificação;
  - Notificação de novos fluxos para o *Flow Adaptation*;
- Outros Eventos;
  - Leitura das informações de contexto atualizadas no *Context Model Database*;
  - Processamento das regras de fluxo pré-definidas;
  - Salva das novas regras de encaminhamento no *Flow Repository*;

### ***Flow Adaptation***

- Eventos de Recebimento;
  - Recebimento da inclusão de novas regras de encaminhamento;
- Outros Eventos;
  - Consulta ao *Flow Repository*;
  - Aplicação das novas regras de encaminhamento nos comutadores.



## ANEXO C – EXEMPLO DE XML

```

<?xml version="1.0" encoding="UTF-8"?>
<Entity>

<ContextVersion>
  1
</ContextVersion>

<Individuality>
<Collected_Username>Joe</Collected_Username>
  <Collected_Hostname>wks1</Collected_Hostname>
  <Subscribe>
    <Provided_Subscribe_ID>11</Provided_Subscribe_ID>
    <Provided_Optional_String>Text</Provided_Optional_String>
    <Provided_Optional_Integer>1000</Provided_Optional_Integer>
    <Provided_Optional_Real>1000</Provided_Optional_Real>
  </Subscribe>
  <Collected_Domainname>lab.unifacs.br</Collected_Domainname>
  <Collected_MAC>84-8F-69-CA-72-CE</Collected_MAC>
  <Collected_IPv4>10.0.0.1</Collected_IPv4>
</Individuality>

<Time>
<Collected_NTP>Yes</Collected_NTP>
  <Collected_LastNTPupdate>1</Collected_LastNTPupdate>
<Collected_LocalTime>2014-07-09T19:20-03:00</Collected_LocalTime>
</Time>

<Location>
  <Collected_GPScoordinates>12°58'47.75"S
38°27'31.37"O</Collected_GPScoordinates>
  <Collected_Network>10.0.0.0/24</Collected_Network>

```

```

    <Provided_City>Salvador</Provided_City>
    <Provided_Near>Caminho das Árvores</Provided_Near>
    <Provided_Near>Unifacs</Provided_Near>
  </Location>

  <Activity>
    <Act>
      <Collected_rID>1</Collected_rID>
      <Provided_Subscribe_ID>11</Provided_Subscribe_ID>
      </Act>
      <Act>
        <Collected_rID>2</Collected_rID>
      </Act>
    </Activity>

  <Relations>
    <Rel>
      <Collected_rID>1</Collected_rID>
      <Collected_Conected>172.16.0.2</Collected_Conected>
      <Collected_ApplicationName>Skype</Collected_ApplicationName>
      <Collected_ApplicationExecutable>skype.exe</Collected_ApplicationExecuta
ble>
      <Collected_ApplicationVersion>7.0</Collected_ApplicationVersion>
      <Collected_SourcePort>3032</Collected_SourcePort>
      <Collected_DestinationPort>80</Collected_DestinationPort>
      <Collected_TransportProtocol>TCP</Collected_TransportProtocol>
      <Collected_ApplicationProtocol>HTTP</Collected_ApplicationProtocol>
    </Rel>
    <Rel>
      <Collected_rID>2</Collected_rID>
      <Collected_Conected>10.0.0.33</Collected_Conected>
      <Collected_ApplicationName>Viber</Collected_ApplicationName>
      <Collected_ApplicationExecutable>viber.exe</Collected_ApplicationExecuta
ble>

```

```

    <Collected_ApplicationVersion>5.0</Collected_ApplicationVersion>
    <Collected_SourcePort>3033</Collected_SourcePort>
    <Collected_DestinationPort>443</Collected_DestinationPort>
    <Collected_TransportProtocol>TCP</Collected_TransportProtocol>
    <Collected_ApplicationProtocol>HTTPS</Collected_ApplicationProtocol>
  </Rel>
</Relations>

```

```

<QoE>
  <Peer>
    <Collected_rID>1</Collected_rID>
    <Collected_MOSs>4.4</Collected_MOSs>
    <Provided_MOSu>2</Provided_MOSu>
    <Collected_RFactor>96</Collected_RFactor>
    <Collected_Noise>100</Collected_Noise>
  </Peer>
  <Peer>
    <Collected_rID>2</Collected_rID>
    <Collected_MOSs>5</Collected_MOSs>
  </Peer>
</QoE>

```

```

<QoD>
  <Collected_HasGPS>Yes</Collected_HasGPS>
    <Collected_GPSprecision>10</Collected_GPSprecision>
    <Collected_ProcessorCores>4</Collected_ProcessorCores>
    <Collected_ProcessorClock>2,6</Collected_ProcessorClock>
  <Collected_Memory>6</Collected_Memory>
  <Nic>
    <Collected_NIC_ID>1</Collected_NIC_ID>
    <Collected_Bandwidth>10000</Collected_Bandwidth>
    <Collected_HasToE>Yes</Collected_HasToE>
    <Collected_MAC>84-8F-69-CA-72-CE</Collected_MAC>
    <Collected_IPv4>10.0.0.1</Collected_IPv4>

```

</Nic>

</QoS>

<QoS>

<Nic>

<Collected\_NIC\_ID>1</Collected\_NIC\_ID>

<Collected\_Throughput>0,7</Collected\_Throughput>

<Collected\_Loss>3</Collected\_Loss>

<Collected\_Latency>3</Collected\_Latency>

<Collected\_Jitter>2</Collected\_Jitter>

</Nic>

</QoS>

</Entity>

**ANEXO D – METADADOS DAS TABELAS DO CONTEXT DATABASE**

TABELA	DESCRIÇÃO DA TABELA	CAMPO	TIPO	PK/ FK	RELACIONADO A	OBS
PARAMETERGROUP	ARMAZENA OS GRUPOS DE PARAMETROS	PARMGRPID	INT	PK		IDENTIFICAÇÃO ÚNICA DO GRUPO DE PARÂMETROS
	DE QOD, QOE e QOS	DESCRIPTION	VARCHAR(10)			DESCRIÇÃO DO GRUPO DE PARÂMETROS
PARAMETER	ARMAZENA OS PARÂMETROS UTILIZADOS	PARMID	INT	PK		IDENTIFICAÇÃO ÚNICA DOS PARÂMETROS
	NA ARQUITETURA	DESCRIPTION	VARCHAR(30)			DESCRIÇÃO DO PARÂMETRO
		MEASURE	VARCHAR(20)			UNIDADE DE MEDIDA DO PARÂMETRO
		PARMGRPID	INT	FK	PARAMETERGROUP	IDENTIFICAÇÃO DO GRUPO DE PARAMETROS
DEVICE	ARMAZENA TODOS OS DISPOSITIVOS	DEVID	INT	PK		IDENTIFICAÇÃO ÚNICA DOS DISPOSITIVOS
	DENTRO DO DOMÍNIO DO CONTROLLER	DEV_STA_FWR	INT			STATUS DO DISPOSITIVO SE É DE ENCAMINHAMENTO (0 - NÃO, 1 - SIM)
	SEJAM DISPOSITIVOS DE ENCAMINHAMENTO-	DEV_STATUS	INT			STATUS DO DISPOSITIVO (0 - INATIVO, 1 - ATIVO, 2 - TIMEOUT, 3 - OVERLOAD, 4 - FAIL)

TABELA	DESCRIÇÃO DA TABELA	CAMPO	TIPO	PK/ FK	RELACIONADO A	OBS
	TO OU NÃO	DEV_TS	TIMESTAMP			ULTIMA ATUALIZAÇÃO DO REGISTRO
DEVICEPARAMETER	ARMAZENA TODOS OS PARÂMETROS	DEVID	INT	PK/ FK	DEVICE	IDENTIFICAÇÃO DO DEVICE (PARTE 1 DA CHAVE)
	COLETADOS DOS DISPOSITIVOS QUE SÃO	PARMID	INT	PK/ FK	PARAMETER	IDENTIFICAÇÃO DO PARAMETER (PARTE 2 DA CHAVE)
	AVALIADOS DENTRO DO CONTROLLER	CURVALUE	VARCHAR(30)			VALOR ATUAL DO PARÂMETRO
		MINVALUE	INT			VALOR MÍNIMO ( 0 - NÃO PRESENTE )
		MAXVALUE	INT			VALOR MÁXIMO (1 - PRESENTE )
		STA_QOC	INT			STATUS SE QOC OU NÃO (0 - NÃO, 1 - SIM )
		STA_CTX	INT			STATUS SE CONTEXTO OU NÃO (0 - NÃO, 1 - SIM )
INDIVIDUALITY	ARMAZENA TODAS AS INDIVIDUALIDADES QUE	INDID	INT	PK		IDENTIFICAÇÃO DO USUÁRIO
	PERTENCEM AO DOMÍNIO DA ARQUITETURA	DEVID	INT	FK	DEVICE	IDENTIFICAÇÃO DO DISPOSITIVO
		IND_TS	TIMESTAMP			DATA E HORA DA ÚLTIMA ATUALIZAÇÃO

TABELA	DESCRIÇÃO DA TABELA	CAMPO	TIPO	PK/ FK	RELACIONADO A	OBS
		IND_IS_ACT	SIM/NÃO			A ASSINATURA DO ASSINANTE EM RELAÇÃO A SERVIÇO ESPECÍFICO
INDIVIDUALITYPARAMETER	ARMAZENA OS DADOS DE PARÂMETROS	INDID	INT	PK/ FK	INDIVIDUALITY	IDENTIFICAÇÃO DA INDIVIDUALIDADE
	COLETADOS DA INDIVIDUALIDADE	PARMID	INT	PK/ FK	PARAMETER	IDENTIFICAÇÃO DO PARÂMETRO
		IND_CURVAL	VARCHAR(30)			VALOR ATUAL DO PARÂMETRO
		IND_STA_QOC	INT			STATUS SE QOC OU NÃO (0 - NÃO, 1 - SIM )
APPLICATIONGROUP	ARMAZENA O GRUPO DE APLICAÇÕES	APPLGRPID	INT	PK		IDENTIFICAÇÃO DO GRUPO DE APLICAÇÕES
	TRATADOS NA ARQUITETURA	DESCRIPTION	VARCHAR(30)			DESCRIÇÃO DO GRUPO DE APLICAÇÕES
		GRP_PRIORITY_ORDER	INT			
APPLICATION	ARMAZENA AS APLICAÇÕES QUE SÃO	APPLID	INT	PK		IDENTIFICAÇÃO DA APLICAÇÃO
	TRATADAS NA ARQUITETURA	APPL_GRP_ID	INT	FK	APPLICATION_GROUP	IDENTIFICAÇÃO DO GRUPO DA APLICAÇÃO

TABELA	DESCRIÇÃO DA TABELA	CAMPO	TIPO	PK/ FK	RELACIONADO A	OBS
		APPL_DESC	VARCHAR(30)			DESCRIÇÃO DA APLICAÇÃO
		APPL_PRIO_ORDE	INT			FOR EXAMPLE: STANDARD APPLICATION IS SKYPE BUT YOU CAN USE VIBER (USED FOR STANDARD APPLICATIONS OF COMPANY)
		APPL_TS	TIMESTAMP			ÚLTIMA DATA/HORA
APPLICATIONPARAMETER	ARMAZENA OS PARÂMETROS BÁSICOS DE	APPLID	INT	PK/ FK	APPLICATION	IDENTIFICAÇÃO DA APLICAÇÃO (PARTE 1 DA CHAVE)
	AVALIAÇÃO DAS APLICAÇÕES	PARMID	INT	PK/ FK	PARAMETER	IDENTIFICAÇÃO DO PARÂMETRO (PARTE 2 DA CHAVE)
		MINVALUE	INT			FAIXA MÍNIMA DO PARÂMETRO DA APLICAÇÃO
		MAXVALUE	INT			FAIXA MÁXIMA DO PARÂMETRO DA APLICAÇÃO



TABELA	DESCRIÇÃO DA TABELA	CAMPO	TIPO	PK/ FK	RELACIONADO A	OBS
		STRVALUE	VARCHAR(50)			ARMAZENA DADOS ALFANUMÉRICOS COMO NOME DO PROGRAMA OU NOME DO EXECUTÁVEL
RELATION	ARMAZENA OS DADOS DAS RELAÇÕES QUE	RELID	INT	PK		IDENTIFICAÇÃO DA RELAÇÃO
	USAM O CONTEXTO	USERID_SRC	INT	FK	USER	IDENTIFICAÇÃO DO USUÁRIO ORIGEM
		USERID_DST	INT	FK	USER	IDENTIFICAÇÃO DO USUÁRIO DESTINO
		DEVID_SRC	INT	FK	DEVICE	IDENTIFICAÇÃO DO DISPOSITIVO ORIGEM
		DEVID_TRG	INT	FK	DEVICE	IDENTIFICAÇÃO DO DISPOSITIVO DESTINO
		IS_OF	INT			STATUS QUE INDICA SE A COMUNICAÇÃO USA PROTOCOLO OPENFLOW OU NÃO(0 - NÃO, 1 - SIM)
		TS_REL	TIMESTAMP			ATUALIZAÇÃO DO REGISTRO

TABELA	DESCRIÇÃO DA TABELA	CAMPO	TIPO	PK/ FK	RELACIONADO A	OBS
ACTIVITY	ARMAZENA DADOS ATUAIS DE CADA APLICAÇÃO	RELID	INT	PK/ FK	RELATION	IDENTIFICAÇÃO DA RELAÇÃO
	USADA EM CADA RELAÇÃO	APPLID	INT	PK/ FK	APPLICATION	IDENTIFICAÇÃO DA APLICAÇÃO
		PARMID	INT	PK/ FK	PARAMETER	IDENTIFICAÇÃO DO PARÂMETRO
		ACT_CURVAL	INT			VALOR ATUAL DO PARÂMETRO
		ACT_STA_QOC	INT			STATUS SE QOC OU NÃO (0 - NÃO, 1 - SIM)
		ACT_STA_CTX	INT			STATUS CONTEXTO OU NÃO (0 - NÃO, 1 - SIM, 2 - ANTIGO)
PATH	ARMAZENA OS DADOS DE CAMINHO	PATHID	INT	PK		IDENTIFICAÇÃO DO CAMINHO
	ENTRE ROTEADOR ORIGEM E DESTINO	FWRID_INI	INT	FK	DEVICE	IDENTIFICAÇÃO DO DISPOSITIVO DE ENCAMINHAMENTO ORIGEM
		FWRID_FIN	INT	FK	DEVICE	IDENTIFICAÇÃO DO DISPOSITIVO DE ENCAMINHAMENTO DESTINO
		HOP_COUNT	INT			QUANTIDADE DE SALTOS DESSE CAMINHO

TABELA	DESCRIÇÃO DA TABELA	CAMPO	TIPO	PK/ FK	RELACIONADO A	OBS
LINK	ARMAZENA OS DADOS DE LINK ENTRE	LINKID	INT	PK		IDENTIFICAÇÃO DO LINK
	ROTEADOR ORIGEM E DESTINO	FWRID_INI	INT	FK	DEVICE	IDENTIFICAÇÃO DO DISPOSITIVO DE ENCAMINHAMENTO ORIGEM
		FWRID_FIN	INT	FK	DEVICE	IDENTIFICAÇÃO DO DISPOSITIVO DE ENCAMINHAMENTO DESTINO
		STATUS	INT			STATUS DO LINK (0 - INATIVO, 1 - ATIVO, 2 - TIMEOUT, 3 - OVERLOAD, 4 - FAIL)
LINKPARAMETER	ARMAZENA OS DADOS DE PARÂMETROS	LINKID	INT	PK/ FK	LINK	IDENTIFICAÇÃO DO LINK
	COLETADOS DO LINK	PARMID	INT	PK/ FK	PARAMETER	IDENTIFICAÇÃO DO PARÂMETRO
		CURVALUE	INT			VALOR ATUAL DO PARÂMETRO DO LINK
PATHLINK	ARMAZENA OS DADOS DE CAMINHO E	PATHID	INT	PK/ FK		IDENTIFICAÇÃO DO CAMINHO
	LINK ASSOCIADOS	LINKID	INT	PK/ FK		IDENTIFICAÇÃO DO LINK

TABELA	DESCRIÇÃO DA TABELA	CAMPO	TIPO	PK/ FK	RELACIONADO A	OBS
QUEUE	ARMAZENA OS DADOS DE TIPOS DE FILA	QUEUEID	INT	PK		IDENTIFICAÇÃO DO TIPO DA FILA
		DESCRIPTION	VARCHAR(30)			DESCRIÇÃO DA FILA
FORWARDINGEVENT	ARMAZENA OS DADOS DE EVENTOS E A	EVTFWRID	INT	PK		IDENTIFICAÇÃO DO EVENTO DE ENCAMINHAMENTO
	ESCOLHA DO NOVO CAMINHO	RELID	INT	FK	EVENT	IDENTIFICAÇÃO DO EVENTO
		PATHID	INT	FK	PATH	IDENTIFICAÇÃO DO CAMINHO ESCOLHIDO
		QUEUEID	INT	FK	QUEUE	IDENTIFICAÇÃO DA FILA ESCOLHIDA
		TS_FWR	TIMESTAMP			ÚLTIMA ATUALIZAÇÃO DO EVENTO
RULES	ARMAZENA TODAS AS REGRAS DE QOC E	TYPE	INT	PK		RULES ( 1 - QOC, 2 - FORWARDING ) ( PARTE 1 DA CHAVE )
	ENCAMINHAMENTO	RULESID	INT	PK		IDENTIFICAÇÃO DA REGRA ( PARTE 2 DA CHAVE )
		TABLEID	INT			TIPO DO ID ( 1- DEVICE, 2 - LINK , 3 - APPLICATION, 4 - INDIVIDUALITY )
		PARMID	INT	FK	PARAMETER	IDENTIFICAÇÃO DO PARÂMETRO

## ANEXO E – XMLs DOS CENÁRIOS DO ESTUDO DE CASO

```

<?xml version="1.0" encoding="UTF-8"?>
<Entity>

<ContextVersion>
  1
</ContextVersion>

<Individuality>
<Collected_Username>User-N10</Collected_Username>
  <Collected_Hostname>Host-N10</Collected_Hostname>
  <Collected_Domainname>lab.unifacs.br</Collected_Domainname>
  <Collected_MAC>00-53-00-00-10</Collected_MAC>
  <Collected_IPv4>192.168.1.10</Collected_IPv4>
</Individuality>

<Time>
<Collected_NTP>No</Collected_NTP>
  <Collected_LastNTPupdate>0</Collected_LastNTPupdate>
<Collected_LocalTime>2015-07-30T08:01:0-03:00</Collected_LocalTime>
</Time>

<Location>
  <Collected_Network>192.168.1.0/24</Collected_Network>
</Location>

<Activity>
<Act>
  <Collected_rID>1</Collected_rID>
</Act>
</Activity>

<Relations>
  <Rel>
    <Collected_rID>1</Collected_rID>
    <Collected_Conected>192.168.4.16</Collected_Conected>
    <Collected_ApplicationName>FileZilla</Collected_ApplicationName>
    <Collected_ApplicationExecutable>filezilla.exe</Collected_ApplicationExecutable>
    <Collected_ApplicationVersion>3.13</Collected_ApplicationVersion>
    <Collected_SourcePort>10101</Collected_SourcePort>
    <Collected_DestinationPort>21</Collected_DestinationPort>
    <Collected_TransportProtocol>TCP</Collected_TransportProtocol>
  </Rel>
</Relations>
</Entity>

```

## ANEXO F – SQLs DOS CENÁRIOS DO ESTUDO DE CASO

```

--- pegar o melhor link
select
  from linkparameter
  where linkid in
    (
select distinct linkid as conexao
  from link_fwr
  where devid in (src_devid,trg_devid)
  and linkid not in (select linkid from link where link_status = 0)          ---- restrição de algumas regras de encaminhamento
(links ativos)
  and linkid in (select linkid from linkparameter where parmId in (1,2,3,4,5,6) and link_curval >= link_minval          ---- restrição de
algumas regras de encaminhamento (links com baixo desempenho)
    )
  and parmId in (1,2,3,4,5,6)
order by link_curval;

---- todos os caminhos com a média de atraso e throughput

select c.pathid , parmId, sum(link_minval)/count(*), sum(link_maxval)/count(*)
  from linkparameter a inner join link b on (a.linkid = b.linkid and link_status = 1 and parmId in (1,2))
        inner join path_link c on (a.linkid = c.linkid)
group by c.pathid, parmId

---- selecionando informações entre origem e destino

create view vw_contexto
as
select * from relation where rel_is_ctx = 1

create view vw_indiv_device_mask
as
select b.* from individuality a inner join individualityparameter b on (a.indid = b.indid and b.parmId = 47)

create view vw_indiv_device_ipv4
as select b.* from individuality a inner join individualityparameter b on (a.indid = b.indid and b.parmId = 26)

create view vw_activity_throughput
as select b.*, a.relid, a.act_curval from activity a inner join applicationparameter b on (a.applid = b.applid and a.parmId = b.parmId and
b.parmId = 1)

create view vw_activity_delay
as select b.*, a.relid, a.act_curval from activity a inner join applicationparameter b on (a.applid = b.applid and a.parmId = b.parmId and
b.parmId = 2)

create view vw_activity_applname
as select b.*, a.relid from activity a inner join applicationparameter b on (a.applid = b.applid and a.parmId = b.parmId and b.parmId
= 31)

```

## ANEXO G – TCLs DOS CENÁRIOS DO ESTUDO DE CASO

```

# MPLS Lables
# $ns at 0.5 "[N0 get-module MPLS] ldp-trigger-by-withdraw 10 -1"
$ns at 0.5 "[N3 get-module MPLS] ldp-trigger-by-withdraw 16 -1"
$ns at 0.5 "[N9 get-module MPLS] ldp-trigger-by-withdraw 14 -1"
$ns at 0.5 "[N0 get-module MPLS] ldp-trigger-by-withdraw 11 -1"

# UDP2
$ns at 1.0 "[N6 get-module MPLS] pft-dump"
$ns at 1.0 "[N6 get-module MPLS] erb-dump"
$ns at 1.0 "[N6 get-module MPLS] lib-dump"

$ns at 1.1 "[N6 get-module MPLS] make-explicit-route 9 6_7_4_1_2_9 1003 -1"
# $ns at 4.0 "[N6 get-module MPLS] make-explicit-route 9 6_7_4_1_5_8_9 1003 -1"
$ns at 1.2 "[N6 get-module MPLS] flow-aggregation 9 -1 3 -1"
$ns at 1.3 "[N6 get-module MPLS] flow-erlsp-install 14 -1 1003"

$ns at 1.6 "[N6 get-module MPLS] pft-dump"
$ns at 1.6 "[N6 get-module MPLS] erb-dump"
$ns at 1.6 "[N6 get-module MPLS] lib-dump"

$ns at 7.5 "[N6 get-module MPLS] ldp-trigger-by-release 14 1003"

# udp0
$ns at 3.9 "[N0 get-module MPLS] pft-dump"
$ns at 3.9 "[N0 get-module MPLS] erb-dump"
$ns at 3.9 "[N0 get-module MPLS] lib-dump"

$ns at 4.0 "[N0 get-module MPLS] make-explicit-route 3 0_1_5_8_3 1001 -1"
$ns at 4.1 "[N0 get-module MPLS] make-explicit-route 3 0_1_5_8_3 1001 -1"
$ns at 4.2 "[N0 get-module MPLS] flow-aggregation 0 -1 3 -1"
$ns at 4.3 "[N0 get-module MPLS] flow-erlsp-install 16 -1 1001"

$ns at 4.5 "[N0 get-module MPLS] pft-dump"
$ns at 4.5 "[N0 get-module MPLS] erb-dump"
$ns at 4.5 "[N0 get-module MPLS] lib-dump"

$ns at 7.5 "[N0 get-module MPLS] ldp-trigger-by-release 16 1001"

$ns at 7.6 "[N0 get-module MPLS] pft-dump"
$ns at 7.6 "[N0 get-module MPLS] erb-dump"
$ns at 7.6 "[N0 get-module MPLS] lib-dump"

# udp3
$ns at 3.9 "[N9 get-module MPLS] pft-dump"
$ns at 3.9 "[N9 get-module MPLS] erb-dump"
$ns at 3.9 "[N9 get-module MPLS] lib-dump"

$ns at 5.0 "[N9 get-module MPLS] make-explicit-route 0 9_8_5_1_0 1004 -1"

```

\$ns at 5.1 "[n9 get-module MPLS] make-explicit-route 0 9\_8\_5\_1\_0 1004 -1"

\$ns at 5.2 "[n9 get-module MPLS] flow-aggregation 3 -1 0 -1"

\$ns at 5.3 "[n9 get-module MPLS] flow-erlsp-install 11 -1 1004"

\$ns at 5.5 "[n9 get-module MPLS] pft-dump"

\$ns at 5.5 "[n9 get-module MPLS] erb-dump"

\$ns at 5.5 "[n9 get-module MPLS] lib-dump"

\$ns at 9.7 "[n9 get-module MPLS] ldp-trigger-by-release 11 1004"

\$ns at 9.8 "[n9 get-module MPLS] pft-dump"

\$ns at 9.8 "[n9 get-module MPLS] erb-dump"

\$ns at 9.8 "[n9 get-module MPLS] lib-dump"



## ANEXO H – COMANDOS SQL EXECUTADOS NO CONTROLADOR

- a) Seleção do contexto atual dentro do controlador

```
select * from relation where rel_is_ctx = 1
```

- b) Popula a tabela de eventos de encaminhamento

```
Insert into forwardingevent (relid,ts_fwr,is)
```

```
select relid, current_timestamp from relation where rel_is_ctx = 1
```

- c) Seleção dos dados de contexto aptos para análise de encaminhamento

```
select pathid, path_link.linkid, dev_curval_txt
```

```
from (select distinct src.PATH
```

```
from
```

```
( select dp1.dev_curval_txt src_network, dp2.dev_curval_txt trg_network
```

```
from relation rr1 inner join individuality ii1 on rr1.indid_src = ii1.indid
```

```
inner join relation rr2 on rr1.relid_trg = rr2.relid
```

```
inner join individuality ii2 on rr2.indid_src = ii2.indid
```

```
inner join device dv1 on ii1.devid = dv1.devid
```

```
inner join deviceparameter dp1 on dv1.devid = dp1.devid
```

```
inner join parameter_dev pd1 on dp1.parmid = pd1.parmid
```

```
inner join device dv2 on ii2.devid = dv2.devid
```

```
inner join deviceparameter dp2 on dv2.devid = dp2.devid
```

```
inner join parameter_dev pd2 on dp2.parmid = pd2.parmid
```

```

where pd1.description like 'NETWORK IPV4%'

AND pd2.description like 'NETWORK IPV4%') rel

inner join

( select pf1.pathid path, dp1.dev_curval_txt src_network

from path_fwr pf1 inner join device      df1 on pf1.devid      = df1.devid

and df1.dev_sta_fwr = 1

inner join deviceparameter dp1 on df1.devid      = dp1.devid

inner join parameter_dev  pd1 on dp1.parmid      = pd1.parmid

where pd1.description like 'NETWORK IPV4%') src          on
rel.src_network = src.src_network

inner join

( select pf2.pathid path, dp2.dev_curval_txt trg_network

from path_fwr pf2 inner join device      df2 on pf2.devid      = df2.devid

and df2.dev_sta_fwr = 1

inner join deviceparameter dp2 on df2.devid      = dp2.devid

inner join parameter_dev  pd2 on dp2.parmid      = pd2.parmid

where pd2.description like 'NETWORK IPV4%') trg          on
rel.trg_network = trg.trg_network ) path

inner join path_link on (path.path = path_link.pathid)

inner join link_fwr on (path_link.linkid = link_fwr.linkid)

inner join link      on (link_fwr.linkid = link.linkid)

inner join device    on (link_fwr.devid = device.devid)

inner join deviceparameter on (device.devid = deviceparameter.devid)

```

```

        inner join parameter_dev on (deviceparameter.parmid =
parameter_dev.parmid)

```

```

where parameter_dev.description like 'NETWORK IPV4%'

```

```

and device.dev_sta_fwr = 1

```

```

and device.dev_status = 1

```

```

and link.link_status = 1

```

d) Seleção dos dados de contexto aptos para análise de encaminhamento

```

select linkid, parmid, link_curval

```

```

from link_parameter lp inner join

```

```

( select ac.parmid as parmid, act_curval, appl_minval, appl_maxval

```

```

from activity ac inner join relation rr (ac.relid = rr.relid and rr.rel_is_ctx = 1)

```

```

        inner join applicationparameter ap (ac.applid =
ap.applid and ac.parmid = ap.parmid)) ac on (lp.parmid = ac.parmid)

```

```

where lp.link_curval >= appl_minval

```

```

and lp.link_curval <= appl_maxval

```

e) Configuração do novo caminho para o evento

```

Update forwardingevent set pathid = caminho_escolhido , is_new = 1 where
forwardingevent = id

```

f) Seleciona os novos caminhos a serem aplicados na camada de encaminhamento

```

Select * from forwardingevent where is_new = 1

```