

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**UM INTERPRETADOR DE REGRAS
PARA GERENCIAMENTO DE AÇÕES
DE CONTROLE DE ACESSO E
ALARME**

TRABALHO DE GRADUAÇÃO

Rodolfo Leffa de Oliveira

Santa Maria, RS, Brasil

2008

**UM INTERPRETADOR DE REGRAS PARA
GERENCIAMENTO DE AÇÕES DE CONTROLE DE
ACESSO E ALARME**

por

Rodolfo Leffa de Oliveira

Trabalho de Graduação apresentado ao Curso de Ciência da Computação
da Universidade Federal de Santa Maria (UFSM, RS), como requisito
parcial para a obtenção do grau de
Bacharel em Ciência da Computação

Orientador: Prof. Dr. Benhur de Oliveira Stein

**Trabalho de Graduação N° 256
Santa Maria, RS, Brasil**

2008

**Universidade Federal de Santa Maria
Centro de Tecnologia
Curso de Ciência da Computação**

A Comissão Examinadora, abaixo assinada,
aprova o Trabalho de Graduação

**UM INTERPRETADOR DE REGRAS PARA GERENCIAMENTO
DE AÇÕES DE CONTROLE DE ACESSO E ALARME**

elaborado por
Rodolfo Leffa de Oliveira

como requisito parcial para obtenção do grau de
Bacharel em Ciência da Computação

COMISSÃO EXAMINADORA:

Prof. Dr. Benhur de Oliveira Stein
(Presidente/Orientador)

Prof^a Dr^a Andrea Schwertner Charão (UFSM)

Prof. MSc. João Carlos Damasceno Lima (UFSM)

Santa Maria, 31 de janeiro de 2008.

RESUMO

Trabalho de Graduação
Curso de Ciência da Computação
Universidade Federal de Santa Maria

UM INTERPRETADOR DE REGRAS PARA GERENCIAMENTO DE AÇÕES DE CONTROLE DE ACESSO E ALARME

Autor: Rodolfo Leffa de Oliveira

Orientador: Prof. Dr. Benhur de Oliveira Stein

Local e data da defesa: Santa Maria, 31 de janeiro de 2008.

Este trabalho faz parte do projeto CONVERGE UFSM, que tem por objetivo a integração entre diversas tecnologias em uma rede de sensores-atuadores, para o desenvolvimento de um sistema de controle de acesso e alarme. A aplicação desenvolvida neste trabalho consiste no núcleo de *software* do projeto, com o papel de analisar eventos gerados por sensores e comandar a execução de ações em resposta e esses eventos, de acordo com um conjunto de regras. Estas regras definem as políticas de acesso e de segurança do sistema. O trabalho aqui relatado também consiste na comunicação entre a máquina de regras implementada e os sensores e atuadores do projeto. A aplicação desenvolvida foi testada no ambiente do projeto CONVERGE e funcionou de maneira adequada. Uma avaliação de desempenho também foi realizada para avaliar o tempo gasto para a interpretação de um evento.

Palavras-chave: Redes de sensores-atuadores, tabela de decisões, máquina de regras.

ABSTRACT

Graduation Work
Graduation Program in Computer Science
Federal University of Santa Maria

A RULE INTERPRETER FOR THE MANAGEMENT OF ACCESS CONTROL AND ALARM ACTIONS

Author: Rodolfo Leffa de Oliveira
Advisor: Prof. Dr. Benhur de Oliveira Stein

This work is part of the CONVERGE UFSM project, which aims at the integration of several technologies in a network of sensors and actuators, for the development of a access and alarm control system. The application develop in this work is the software core of the project, with the role of analysing events generated by sensors and command the execution of actions in response of those events, according to a set of rules. These rules define the policies of access and security of the system. The work reported in this document also correspond in the communication between the application and the sensors and actuators of the project. The developed application was tested in the enviroment of the CONVERGE project, and it has worked. An evaluation of performance was also realized to evaluate the time spent on the interpretation an event.

Keywords: Networks of Sensor and Actuators, Decision Tables, Rule Machine.

LISTA DE FIGURAS

Figura 2.1 – Tabela de Decisões - Estrutura Clássica	12
Figura 3.1 – Projeto CONVERGE - Aspectos Físicos.....	15
Figura 3.2 – Projeto CONVERGE - Aspectos Lógicos.....	16
Figura 4.1 – Máquina de Regras.....	18
Figura 4.2 – Tradutor de Eventos	20
Figura 4.3 – Tradutor de Ações.....	21
Figura 4.4 – Diagrama de Classes - Tabela de decisões	23
Figura 4.5 – Interface Web - Modelo de configuração de regra e variáveis	24
Figura 5.1 – Regras definidas na interface de configuração.....	28
Figura 5.2 – E-mail recebido para a regra de senha errada	29

LISTA DE ABREVIATURAS E SIGLAS

VoIP	Voz sobre IP
UFSM	Universidade Federal de Santa Maria
PABX	Private Automatic Branch Exchange (Troca Automática de Ramais Privados)
LSC	Laboratório de Sistemas de Computação
GMob	Grupo de Sistema de Computação Móvel
FXO	Foreign eXchange Office (Estação Estrangeira de Troca)

SUMÁRIO

1	INTRODUÇÃO	9
2	REVISÃO DE LITERATURA	11
2.1	Redes de Sensores-Atuadores	11
2.2	Tabelas de Decisões	11
3	PROJETO CONVERGE UFSM	13
3.1	Introdução	13
3.2	Objetivos	14
3.3	Arquitetura Proposta	14
4	NÚCLEO DE REGRAS E AÇÕES	17
4.1	Visão Geral	17
4.2	Solução Proposta	18
4.2.1	Máquina de Regras	18
4.2.2	Interface de comunicação	19
4.2.3	Interface de Configuração	20
4.3	Implementação	21
4.3.1	Máquina de Regras	21
4.3.2	Interface de Comunicação	22
4.3.3	Interface de Configuração	24
5	AVALIAÇÃO	26
5.1	Avaliação da ferramenta no ambiente do projeto CONVERGE	26
5.1.1	Ambiente de Teste	26
5.1.2	Sensores e Atuadores	26
5.1.3	Regras Definidas	27
5.1.4	Resultados Obtidos	29
5.2	Simulação de Ambiente Complexo	30
5.2.1	Regras e Eventos Simulados	30
5.2.2	Teste Realizado	30
5.2.3	Resultados Obtidos	31
6	CONCLUSÃO	32
	REFERÊNCIAS	33

1 INTRODUÇÃO

Na sociedade atual, observa-se um aumento na necessidade por segurança (SCHWARTZ, 1994). Cada vez mais se fazem necessários sistemas que ofereçam segurança, e os custos de aquisição, instalação e manutenção destes muitas vezes são elevados. Além disso, sistemas convencionais de segurança freqüentemente necessitam de interação humana, o que prejudica sua automatização e adiciona a eles possíveis pontos de falha. Essa realidade não é diferente na Universidade Federal de Santa Maria, onde existem muitos laboratórios com equipamentos que necessitam ser protegidos de intrusões.

Além disso com o desenvolvimento científico das últimas décadas novas tecnologias foram desenvolvidas, e a cada dia muitas outras estão surgindo. Juntamente com os benefícios que estas tecnologias nos fornecem, observa-se o aumento da complexidade de gerenciamento e operação destas. Nota-se ainda que estas novas tecnologias geralmente possuem funções específicas, e não são trivialmente agregadas umas com as outras.

O projeto CONVERGE, em desenvolvimento no Laboratório de Sistemas de Computação (LSC) e no Grupo de Sistemas de Computação Móvel (GMob) tem por objetivo promover uma convergência tecnológica das tecnologias Voz sobre IP (*VoIP*) (GOODE, 2002), Redes sem Fio (RAPPAPORT, 1996) e PABX (RYU et al., 1995) em uma rede de sensores e atuadores, para implementação de um sistema integrado de controle de acesso, alarme e telefonia digital.

O objetivo do trabalho aqui relatado é oferecer a camada de *software* necessária para a realização da integração idealizada pelo projeto CONVERGE. Este *software* corresponde ao núcleo de regras do projeto, com a função de tratar os eventos gerados pelos sensores do sistema, interpretar o contexto atual do ambiente, e comandar os atuadores de acordo com regras pré-definidas.

O restante deste documento se encontra organizado da seguinte maneira: no capítulo

2 é feita uma revisão de literatura sobre conceitos pertinentes ao escopo deste trabalho; após, no capítulo 3, é apresentado o projeto CONVERGE; no capítulo 4 é abordada a aplicação desenvolvida, mostrando seu papel perante ao projeto CONVERGE, bem como suas características e detalhes de sua implementação; após, no capítulo 5, é realizada uma avaliação da aplicação; por fim, no capítulo 6, é feita a conclusão do trabalho e são relatados os trabalhos futuros a serem realizados.

2 REVISÃO DE LITERATURA

Neste capítulo são introduzidos conceitos importantes para a realização deste trabalho. Estes conceitos são utilizados tanto na implementação do projeto CONVERGE, quanto no desenvolvimento do programa relatado neste documento.

2.1 Redes de Sensores-Atuadores

Redes de sensores fornecem acesso a sensores, controladores e outros equipamentos distribuídos através de uma rede. Estas redes se tornaram possíveis e se tornam cada vez mais capazes devido à evolução das tecnologias de circuitos integrados e redes sem fio, representando uma nova capacidade de monitoramento e controle de ambientes ou dispositivos. A aplicabilidade destas redes é extensa, como nas áreas de saúde, transporte e segurança, entre outras (POTTIE; KAISER, 2000).

Uma rede de sensores-atuadores pode ser classificada como uma rede de sensores. A diferença neste caso é a utilização de mecanismos de atuação, com o objetivo de corrigir falhas do meio ou controlar o ambiente monitorado (ANTONIO A.F. LOUREIRO JOSE MARCOS S. NOGUEIRA, 2003). Um dos fatores relevantes para a utilização desta estrutura é a coleta de dados nos diversos sensores, sua interpretação e a respectiva tomada de ação (RUIZ et al., 2004).

Tal funcionalidade pode ser obtida através da utilização de uma estrutura de controle, como uma tabela de decisões, por exemplo.

2.2 Tabelas de Decisões

Tabelas de decisões são uma forma de representação tabular de uma lógica de decisões. Assim como sentenças *if-then-else* ou *switch-case* utilizadas em linguagens de programação, estas tabelas associam condições com ações a serem realizadas. No en-

tanto, diferentemente das estruturas citadas, as tabelas são mecanismos mais legíveis para seres humanos, apresentando uma maneira mais elegante e simples de associar diversas condições independentes com ações (KIRK, 1965).

A utilização de tabelas de decisões teve início na década de 1950, inicialmente em modelagem de sistemas complexos e após com o intuito de geração de código, como uma alternativa às linguagens de programação de baixo nível existentes na época (POOCH, 1974).

A figura 2.1 apresenta a estrutura clássica de uma tabela de decisões, representando regras aplicáveis ao projeto CONVERGE que serão vistas mais a frente neste documento. Esta estrutura é composta por quatro quadrantes, dois à esquerda e dois à direita. Nos quadrantes da esquerda, as partes superior e inferior correspondem às definições de todas as condições e ações possíveis, respectivamente. Cada coluna do lado direito da tabela representa uma regra de decisão, tendo-se as entradas possíveis para serem tratadas, e abaixo as ações a serem tomadas. Uma regra de decisão define o conjunto de ações que devem ser tomadas quando reconhecido um conjunto de condições como verdadeiras (POOCH, 1974).

Condições		Regras de Decisão		
		Regra 1	Regra 2	Regra 3
IF	PortaAbriu	X		X
AND	LoginOk		X	X
AND	PontoEntrada		X	
THEN	AbrePorta		X	
AND	BateFoto	X		X
Ações				

Figura 2.1: Tabela de Decisões - Estrutura Clássica

3 PROJETO CONVERGE UFSM

Neste capítulo é apresentado o projeto CONVERGE UFSM: Rede de Sensores-Atuadores para Controle de Acesso em Laboratórios Usando VoIP e Rede Sem Fio, falando de suas características, objetivos e relatando o modelo proposto, bem como o ambiente em que este está sendo desenvolvido e testado.

3.1 Introdução

Atualmente, tem-se diversos equipamentos digitais com funções específicas, como centrais telefônicas com *VoIP*, centrais de alarme e dispositivos eletrônicos para porteiro; porém, faltam soluções de baixo custo que os integre. O aspecto negativo dessa situação é que, com a utilização de soluções disponíveis no mercado, o usuário tem uma mesa repleta de equipamentos e torna-se responsável por utilizar cada um separadamente, dependendo da ação que deseja realizar, e por conhecer os comandos distintos para cada ação. Uma solução mais eficiente e eficaz é aquela que adota uma visão pervasiva/ubíqua da computação, onde os sistemas digitais são integrados, com sensores presentes no ambiente na forma de uma rede de sensores-atuadores que atuam pró-ativamente em face a uma situação contextualizada (SATYANARAYANAN, 2001).

Para atingir a integração tecnológica desejada, várias áreas de estudos estão fazendo suas contribuições, como a Computação Sensível ao Contexto (Context-Aware Computing) (SCHILIT et al., 1994) que cria redes de sensores-atuadores para desempenho de funções específicas e atendimento pró-ativo aos usuários, possuindo essas redes aplicações na área de segurança.

Nesse sentido, o projeto CONVERGE UFSM propõe uma aplicação de convergência de tecnologias para controle de acesso aos laboratórios nas instituições, com o intuito de melhorar o sistema de segurança atual da UFSM - Campus, o qual não prevê câmeras

ou outro mecanismo para segurança dos vários Laboratórios da instituição, que contêm recursos que devem ser resguardados. Acredita-se que esta seja também a realidade em muitas outras instituições.

3.2 Objetivos

Os principais objetivos do CONVERGE são a integração de uma central telefônica analógica com VoIP, e destes com uma rede sem fio, uma central de alarme e um sistema de controle de acesso. Deseja-se com isso gerar uma solução de baixo custo para a segurança dos laboratórios da UFSM. Esta solução também demonstra que a tecnologia VoIP pode ser usada como plataforma base para o desenvolvimento de outras aplicações que anexam novas funcionalidades aos aparelhos telefônicos.

Deve-se observar que este sistema de segurança poderia ser utilizado também em residências e outros estabelecimentos. A grande proliferação dos computadores pessoais fez com que a maioria das casas possuam pelo menos um hoje em dia. Ao invés de se desligar o computador ao sair, por exemplo, este poderia ser alocado para a execução dos programas de controle de acesso e alarme, servindo assim para proteção do patrimônio do local.

3.3 Arquitetura Proposta

O modelo proposto para a implementação do projeto CONVERGE UFSM parte da utilização da estrutura telefônica já existente como dispositivo de acesso ao local. Esta abordagem diminui o custo de implantação do sistema, e possibilita uma integração entre diversos dispositivos, tanto sensores como atuadores, o que é um dos objetivos do projeto.

A figura 3.1 esquematiza a solução de integração das tecnologias de *PABX - VoIP* - dispositivos sem fio sob o ponto de vista físico (*hardware*). O sistema é formado por uma rede de sensores e atuadores que executam as funcionalidades programadas. Nessa rede, sensores podem ser de dois tipos: (i) sensores de *hardware*, como dispositivos de detecção de presença, por exemplo; (ii) sensores de *software*, como uma mensagem SMS ou um software de detecção de movimento de uma câmera de vídeo. Atuadores também podem ser de dois tipos: (i) atuadores de *hardware*, como um mecanismo que abre uma porta por exemplo; (ii) e atuadores de *software*, como uma mensagem por e-mail ou uma mensagem por telefone ou programas específicos desenvolvidos por terceiros.

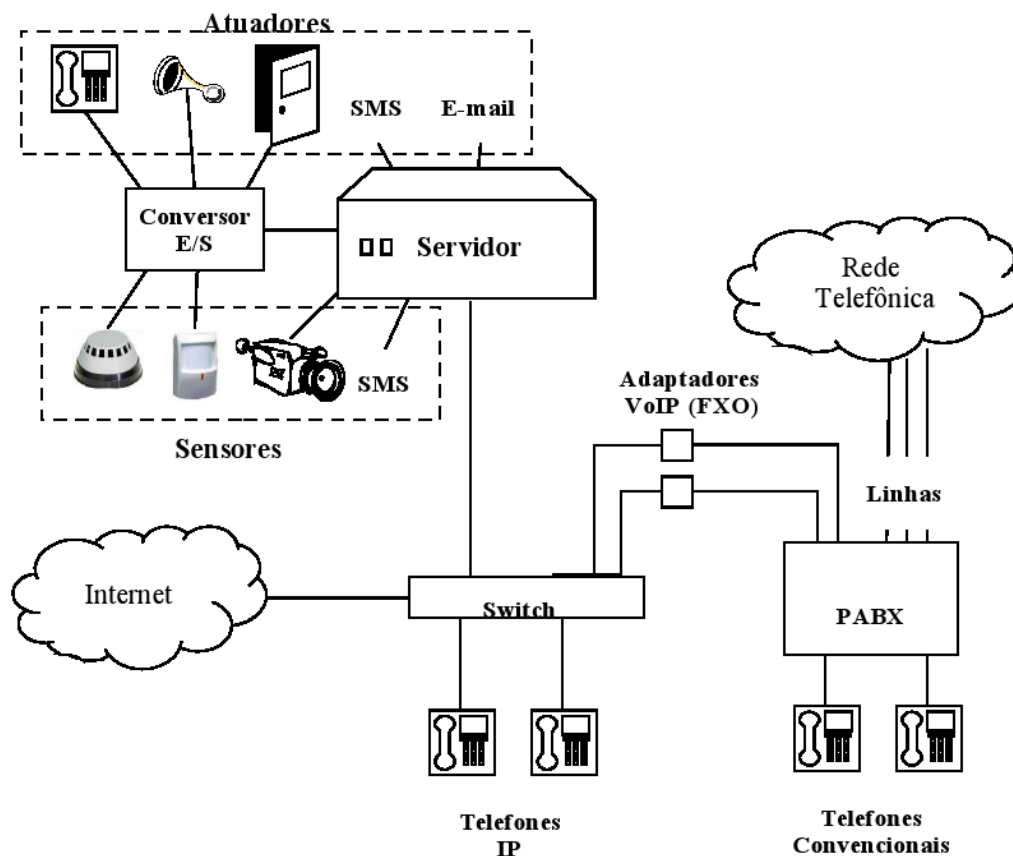


Figura 3.1: Projeto CONVERGE - Aspectos Físicos

Ainda observando-se a figura 3.1, o software para implantação de VoIP, ASTERISK (Digium, 2007), pode atuar como sensor ou atuador de software, dependendo da função ser geradora de informação ou consumidora de informação (ação). Com a adição de adaptadores VoIP do tipo FXO (*Foreign eXchange Office*), que são adaptadores que fornecem uma interface entre um telefone analógico e o padrão VoIP, a proposta da figura 1, é possível também configurar o sistema como uma passarela (gateway) entre VoIP e telefonia convencional.

Sob o ponto de vista de software, sensores e atuadores são reunidos através de regras de interpretação, que contêm a inteligência do sistema, e usam o banco de dados do sistema com informações sobre os usuários, papéis que os usuários exercem (perfil), dispositivos, regras de interpretação. Os papéis que um usuário pode exercer estabelecem as permissões/restrições de acesso que este possui e as ações que pode executar. A figura 3.2 ilustra esse relacionamento.

Pode-se observar na Figura 3.2 que a estrutura de software necessária ao projeto é composta por cinco módulos distintos e interconectados. A interface *Web*, juntamente

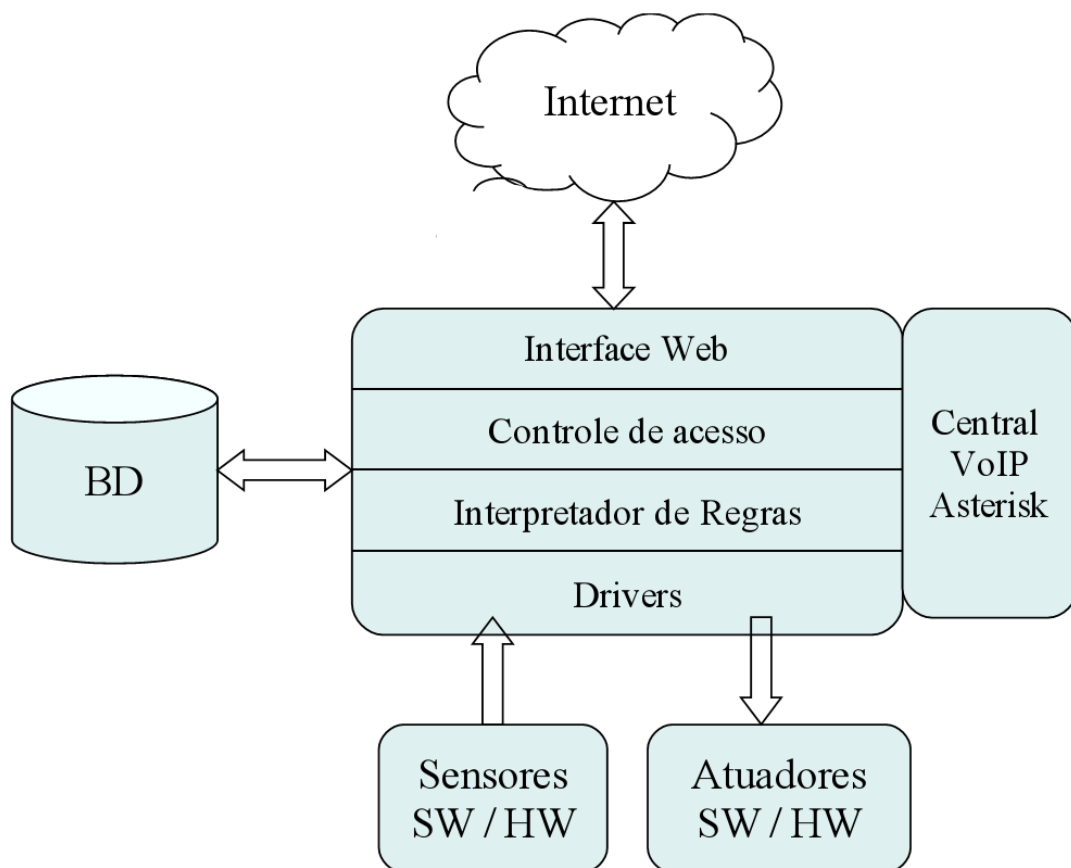


Figura 3.2: Projeto CONVERGE - Aspectos Lógicos

com o sistema de controle aos usuários do sistema estão sendo implementados em um outro trabalho de graduação (ROCHA, 2007). Para a implantação da central VoIP, o *software* necessário se resume ao Asterisk. Os *drivers* necessários para a comunicação do sistema com os sensores e atuadores são implementados, se já não existem, a medida que novos sensores são adicionados ao sistema. Finalmente, o trabalho relatado neste documento é um esforço para a implementação da camada denominada Interpretador de Regras, e para a conexão deste módulo com os outros citados.

4 NÚCLEO DE REGRAS E AÇÕES

Este capítulo relata o desenvolvimento do núcleo interpretador de regras do sistema, descrevendo seus requisitos, sua estrutura proposta e detalhes de sua implementação.

4.1 Visão Geral

O programa desenvolvido é o núcleo de regras do projeto CONVERGE. Este núcleo é responsável pelo gerenciamento das ações de controle de acesso e alarme que se desejam aplicar com o projeto.

O interpretador de regras tem por função analisar eventos gerados por sensores e a partir destas informações comandar a ativação de atuadores. As regras registram quais eventos ou combinação deles define uma situação a ser reconhecida, e quais ações devem ser tomadas frente a esta.

O trabalho realizado também tem como objetivos a definição e implementação de uma interface de comunicação entre a máquina de regras e os *drivers*, tanto de entrada (sensores), como de saída (atuadores). Outra funcionalidade necessária a este programa é a disponibilidade de um modelo de configuração das regras.

A interface de comunicação com os diferentes sensores e atuadores deve ser padronizada, de uma maneira que outros possam ser facilmente adicionados ao ambiente. Esta abordagem visa isolar a máquina dos diferentes *drivers*, facilitando a adição de novos dispositivos, ou a modificação de *drivers* já existentes.

O modelo de configuração neste sistema é essencial para que o administrador possa mudar regras e ações, tornando o sistema dinâmico e facilmente reprogramável. Esta possibilidade de configuração também é fundamental para a adição ou remoção de sensores/atuadores, pois com isto se faz necessária uma reconfiguração do sistema para interpretar, ou deixar de interpretar, o novo dispositivo.

O objetivo da implementação destas características é que o *software* desenvolvido supra as necessidades que o projeto CONVERGE impõe. Esta abordagem se dispõe a oferecer a escalabilidade necessária para a convergência tecnológica desejada.

4.2 Solução Proposta

4.2.1 Máquina de Regras

Visando cumprir com as características necessárias, foi projetada a implementação de uma tabela de decisões, sob a forma de uma máquina de regras. Esta máquina possui a entrada na forma de eventos que são gerados pelos sensores, uma memória composta por um conjunto de variáveis, e sua saída, que são as ações a serem tomadas pelos atuadores, como mostra simplificadamente a figura 4.1.

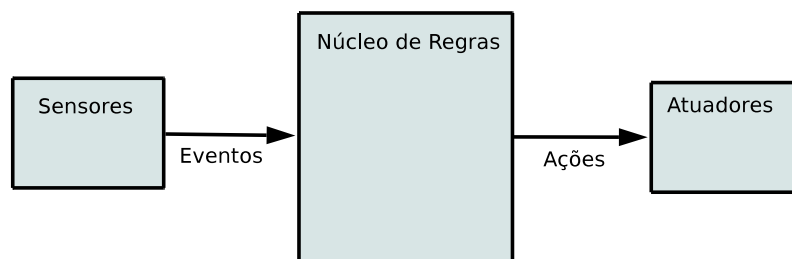


Figura 4.1: Máquina de Regras

A proposta para a interpretação das regras é baseada na modelagem de uma tabela de decisões, representando as regras do sistema como suas condições e ações.

Um evento é uma ou mais informações que representam um acontecimento percebido por algum sensor, que devem ser interpretadas pela máquina. Esta interpretação é a comparação dos valores recebidos com as condições programadas na tabela de decisões. Estas condições são comparações dos valores recebidos entre si, com valores da memória do programa, ou com valores constantes. Quando todas as condições que definem uma regra são reconhecidas como verdadeiras através desta comparação, as ações correspondentes para esta regra de decisão devem ser executadas.

Estas ações a serem executadas podem ser internas ou externas. Uma ação interna é a mudança do valor de uma variável da memória do programa. Ações externas podem ser chamadas para os *drivers* dos atuadores, ou *scripts* escritos para realizar alguma operação, ou ainda chamadas para execução de outros programas.

Para exemplificar este fluxo de dados através da máquina de regras, podemos relatar

uma situação real que poderia acontecer no ambiente de testes proposto. Primeiramente, definimos uma regra de decisão para reconhecer um evento e dizer quais ações se deseja tomar. Tal regra poderia ter como condição que se a porta do laboratório for aberta após as 22 e antes das 7 horas, devem ser tomadas as ações de bater fotos com a câmera instalada na porta e enviar um *e-mail* notificando o evento e enviando as fotos.

Com esta configuração programada, digamos que alguém entre no laboratório as 23 horas. Assim o sensor instalado na porta, que reconhece quando esta é aberta, geraria um evento informando para a máquina de regras que a porta foi aberta. Com esta informação, juntamente com a hora em que o evento foi recebido, a máquina de regras percorreria a tabela de decisões e tentaria reconhecer este evento entre as condições armazenadas. Quando encontrasse a regra programada no parágrafo acima, reconheceria através da comparação dos valores programados com os do evento recebido que aquela regra é verdadeira, e tomaria as ações definidas, sendo no caso a execução do programa que pega imagens da câmera, e o envio destas imagens para o *e-mail* informado, através de um *script* que envia *e-mail*.

Para padronizar e facilitar a comunicação dos sensores e atuadores com a máquina de regras, foram também modelados dois programas auxiliares encarregados de fazer a tradução dos eventos gerados pelos sensores, e a tradução das ações a serem submetidas para os atuadores. O modelo proposto para esta interface de comunicação é descrito na subseção seguinte.

4.2.2 Interface de comunicação

Para o desenvolvimento da interface de comunicação foram projetados dois programas auxiliares, um tradutor de eventos e um tradutor de ações. Estes programas fazem uma ponte de comunicação entre a máquina de regras e os sensores e atuadores do ambiente.

A utilização destes programas agrega na máquina de regras a utilização de um padrão de comunicação, tanto no recebimento de eventos quanto no envio das ações. Este padrão traz benefícios para o administrador do sistema, pois isola a máquina dos sensores e atuadores, facilitando a configuração e a adição ou remoção de dispositivos.

O tradutor de eventos tem por função receber os eventos gerados pelos sensores, fazer a tradução para o formato de eventos padrão, e enviar estes para a máquina de regras, como pode ser visto na Figura 4.2. O arquivo de configuração que a figura apresenta

contém as informações sobre os sensores, os associando à qual evento representam. Assim, na necessidade de modificação na configuração dos sensores pelo administrador, esta pode ser facilmente realizada através da edição deste arquivo.

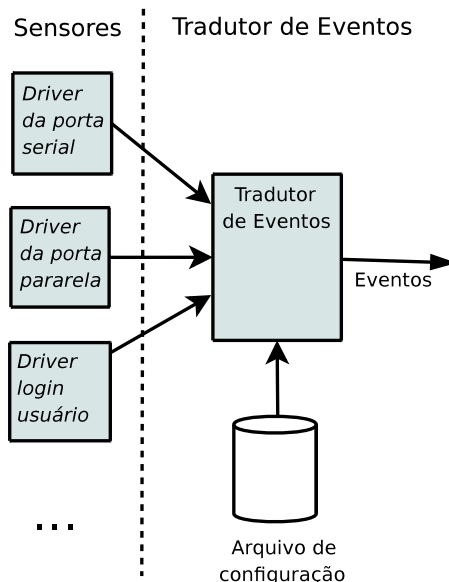


Figura 4.2: Tradutor de Eventos

O tradutor de ações tem uma funcionalidade semelhante ao de eventos, traduzindo informações através de um arquivo de configuração pré-definido. Essas informações, no entanto, são as ações que a máquina de regras deseja efetuar, e o trabalho do tradutor é mapear qual atuador, seja ele de *software* ou de *hardware*, corresponde a cada ação a ser executada. A Figura 4.3 apresenta esta estrutura.

4.2.3 Interface de Configuração

A interface de configuração projetada corresponde a uma integração da máquina de regras implementada com o sítio do projeto CONVERGE, que fornece a interface de gerenciamento do projeto (ROCHA, 2007).

O objetivo da interface é fornecer um método simples para a programação das regras. A proposta é que o administrador consiga inserir eventos através da programação de condições e ações. É interessante também que eventos já cadastrados possam ser visualizados e alterados, facilitando o processo de configuração do sistema.

Outro requisito é a visualização e edição das variáveis que compõem a memória da máquina, como também a adição de novas que possam vir a ser necessárias.

Esta interface traz benefícios ao administrador do sistema, pois centraliza as tarefas

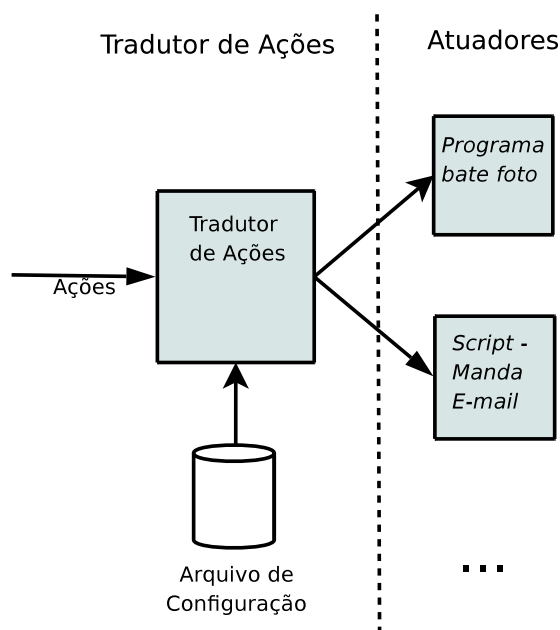


Figura 4.3: Tradutor de Ações

administrativas, o que é um dos objetivos do portal *web* do projeto. A modificação na estrutura dos sensores ou atuadores também é facilitada pela possibilidade de configuração das regras.

4.3 Implementação

4.3.1 Máquina de Regras

No intuito de se implementar a máquina de regras proposta, adotou-se uma abordagem Orientada a Objetos, com a utilização da linguagem de programação Java.

A solução implementada visa a divisão do programa em várias classes distintas e simples, fornecendo estas conjuntamente a funcionalidade desejada. Esta metodologia simplifica o processo de programação e facilita a legibilidade do código através da separação lógica entre as classes, seus atributos e seus métodos. Estas classes representam os objetos do sistema como por exemplo eventos, ações e condições.

A máquina de regras implementada é uma classe bastante simples, que possui como atributos apenas as variáveis que formam a memória do sistema e uma configuração, que corresponde a tabela de decisões implementada. Como métodos principais da máquina de regras, podemos citar apenas três: um que carrega a configuração para a máquina de regras, montando assim a tabela de decisões; um responsável por ler um evento; e outro para localizar um evento, se existente, na tabela de decisões, retornando as ações

correspondentes. Com esta representação, o algoritmo principal do programa fica bastante simples se resumindo em carregar as configuração do sistema, e após isto repetidamente ler um evento e buscar as ações que devem ser tomadas para este evento na tabela de decisões.

A tabela de decisões foi implementada como pode ser visto no diagrama de classes apresentado na Figura 4.4. A classe *Configuracao* representa a tabela de decisões, que é formada por várias regras de decisões. Cada uma destas regras correspondem a um par de condição-ação. Cada condição é um conjunto de expressões a serem comparadas, e caso todas comparações sejam verdadeiras o evento foi validado nesta regra. Quando isto ocorre um objeto que implementa uma ação é retornado para a máquina de regras. Esta ação é na verdade um conjunto de ações individuais, que são executadas uma a uma.

Para a representação de um evento dentro da máquina de regras se utilizou um modelo de mapa de dados, que corresponde a um conjunto de identificadores, cada um associado à um valor. A fim de padronizar a representação de dados dentro do sistema, as variáveis globais da máquina de regras também foram implementadas desta maneira.

4.3.2 Interface de Comunicação

Para o desenvolvimento da interface de configuração foram implementados os dois tradutores. Estes tradutores são programas independentes que se comunicam com os dispositivos e com a máquina de regras através de *sockets*. A informação enviada é uma linha de texto representando um mapa de dados com nomes e valores, mantendo assim o mesmo padrão usado na máquina de regras para as informações.

A utilização de *sockets* transforma a comunicação independente da linguagem de programação ou plataforma. Esta abordagem também traz a possibilidade da execução remota da máquina de regras ou dos tradutores, possibilitando a tomada de decisões ou de ações em outro computador.

O tradutor de eventos recebe conexões e eventos dos sensores, e os traduz através dos eventos mapeados no arquivo de configuração para o formato de informações utilizado pela máquina de regras. Após traduzir o evento, abre uma conexão com a máquina e o envia.

O tradutor de ações funciona da mesma maneira, porém traduzindo ações oriundas da máquina de regras para chamadas de programas que comandam os atuadores do sistema.

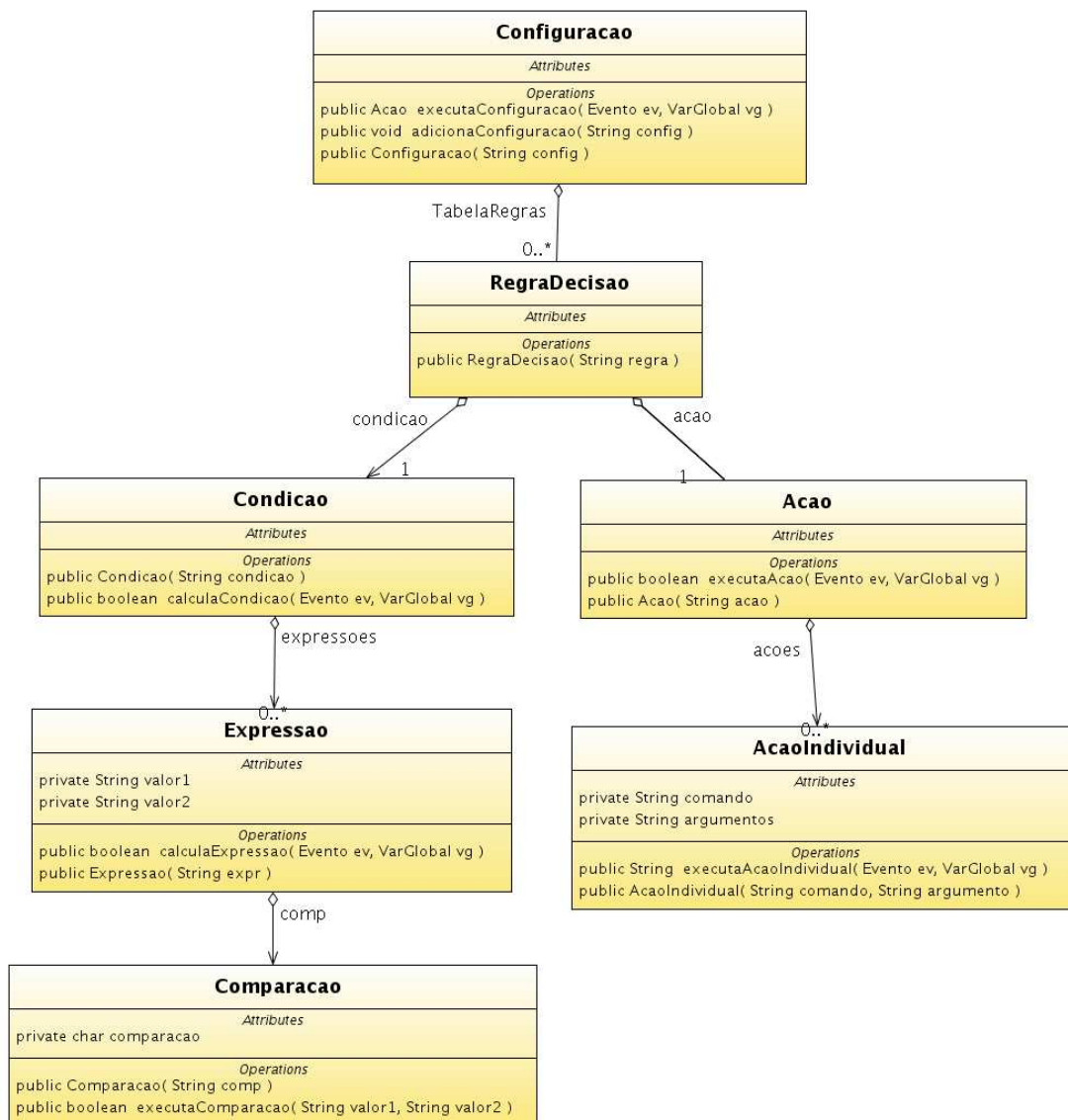


Figura 4.4: Diagrama de Classes - Tabela de decisões

Os tradutores são configurados através de um arquivo de configuração. Neste arquivo são mapeados sinais para eventos, no tradutor de eventos, e ações para chamadas de programas, no tradutor de ações. Estes arquivos são editados manualmente pelo administrador do sistema. Esta edição corresponde a inserção de equivalências, como por exemplo "s1=SensorPresença". Neste caso, quando o tradutor de eventos receber um evento do sensor "s1", ele enviará para a máquina de regras este evento identificado como "SensorPresença".

4.3.3 Interface de Configuração

Para a efetivação do modelo de configuração proposto, foi acrescentada a funcionalidade de adição, remoção e edição de regras e variáveis no portal *web* do projeto. Deve-se lembrar que esta interface foi implementada em outro trabalho (ROCHA, 2007).

A figura 4.5 apresenta a interface de configuração de regras, que está inserida entre as funcionalidades disponíveis no portal. Pode-se ver na figura o mecanismo de cadastro e edição de regras, como também o similar para as variáveis. É visível nesta imagem também a possibilidade de adicionar variáveis durante a programação de uma regras, caso venha a ser necessário.

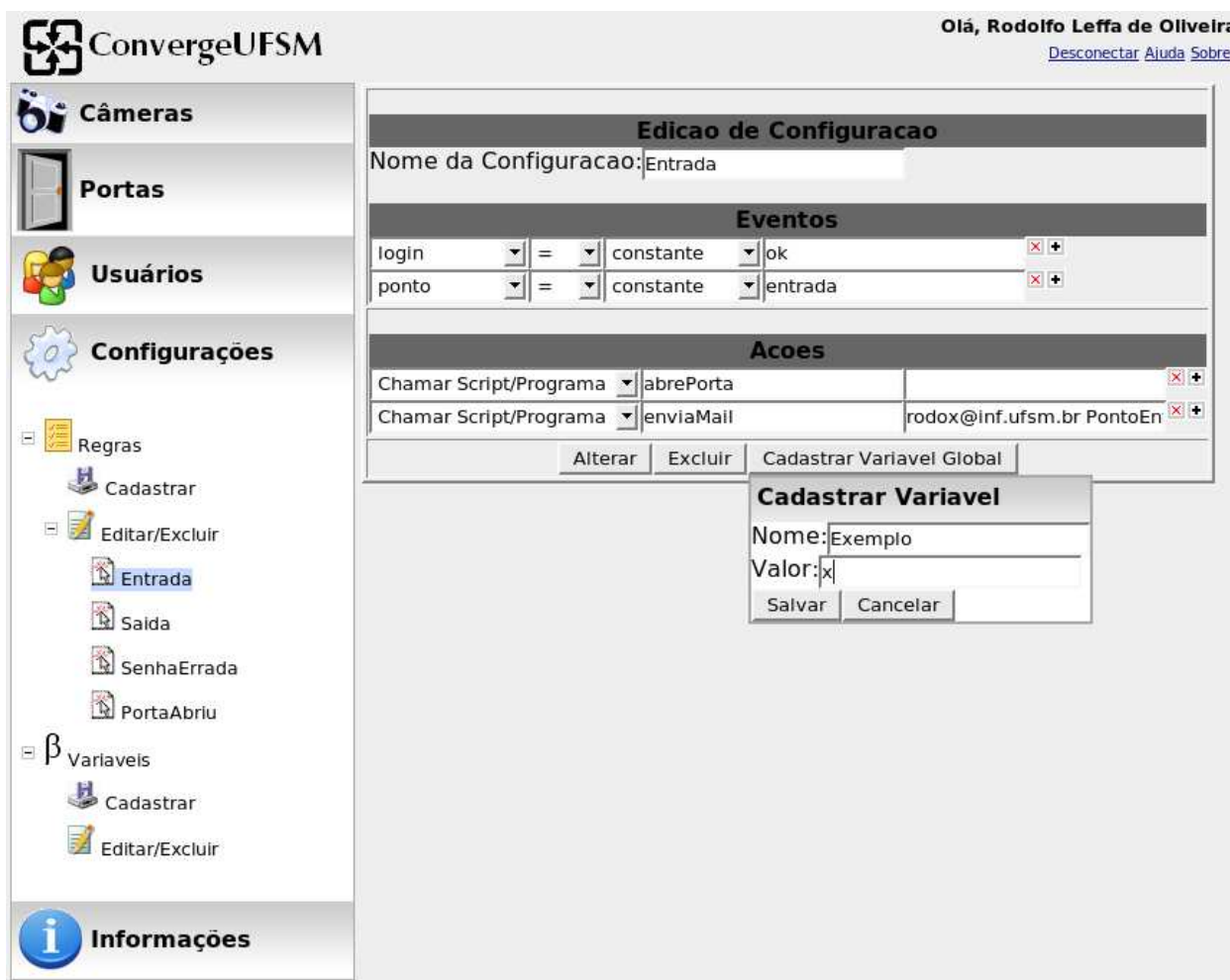


Figura 4.5: Interface Web - Modelo de configuração de regra e variáveis

Para a transferência das regras e variáveis programadas na interface *web* para a máquina de regras, se utilizou a estratégia de persistência dos dados através de arquivos XML (ROCHA, 2007).

Assim, as regras e variáveis programadas são serializadas para arquivos XML. No

começo da execução da máquina de regras, o primeiro passo é carregar os arquivos, montando assim a tabela de decisões e a memória do sistema. A partir deste ponto o núcleo de regras pode começar a receber eventos e interpretar estes.

5 AVALIAÇÃO

Este capítulo apresenta a avaliação da ferramenta desenvolvida. Para tal, foram utilizadas duas abordagens distintas. A primeira tem por objetivo a avaliação da ferramenta no ambiente de testes do projeto CONVERGE. A outra é a simulação de um ambiente complexo. Os experimentos realizados são relatados nas seções seguintes.

As duas seções de avaliação apresentam o objetivo da avaliação, as regras programadas e os eventos a serem reconhecidos. Após, são relatados os resultados obtidos e a análise destes.

5.1 Avaliação da ferramenta no ambiente do projeto CONVERGE

A avaliação da ferramenta perante aos sensores e atuadores que compõem o projeto CONVERGE tem por objetivo analisar o comportamento da máquina de regras frente a uma situação real. Além disso, a ferramenta foi projetada e desenvolvida para suprir as necessidades do projeto.

5.1.1 Ambiente de Teste

O ambiente de realização do projeto CONVERGE corresponde aos laboratórios dos grupos de pesquisa LSC e GMob, que ficam localizados no terceiro andar do prédio de pós-graduação, anexo ao Centro de Tecnologia da UFSM. Este ambiente é composto por laboratórios com salas de pesquisadores, salas de estudos para alunos de graduação e pós-graduação e laboratórios de pesquisa.

5.1.2 Sensores e Atuadores

Os sensores e atuadores utilizados na validação da ferramenta são os que foram instalados ou implementados durante a execução do projeto.

Como sensores tem-se instalado um sensor magnético na porta de acesso aos labora-

tórios do LSC e GMob. O mesmo é ligado a uma placa de aquisição de dados, e a saída desta placa está conectada na porta serial de um computador. O *driver* deste sensor corresponde a uma aplicação desenvolvida que faz monitoramento constante da porta serial. Esse programa gera eventos relacionados aos sinais provenientes do sensor magnético, informando quando a porta é aberta ou fechada.

Outro sensor funcional é formado pela integração de um ramal telefônico com um adaptador VoIP, que faz uma chamada para o *software* Asterisk. Este telefone se encontra localizado na porta principal de acesso aos laboratórios. O usuário digita então um identificador e senha, que são direcionados para um programa de validação, que consulta a base de dados com os usuários cadastrados e verifica se o usuário existe e a senha está correta. A resposta desta autenticação é enviada para a máquina de regras, tanto em caso de estar certa ou errada, neste último caso gerando a informação de que uma senha errada foi digitada. Esta aplicação também gera os eventos relacionados ao controle de acesso, sob a forma de um ponto eletrônico. O usuário antes de digitar sua senha digita um código que informa se ele está entrando ou saindo do laboratório.

Existe um atuador que dispara a abertura da porta. Este atuador é uma aplicação desenvolvida que envia um sinal para a porta paralela do computador. A porta paralela está conectada a uma placa de saída de dados que manda um sinal elétrico para a fechadura eletrônica que se encontra na porta dos laboratórios.

Outro atuador disponível é um programa que bate fotos com a câmera instalada na porta de acesso aos laboratórios. Este programa comanda um *driver* implementado, que faz requisições e recebe imagens da câmera.

Por fim, como atuador também se tem disponível um *script* que manda *e-mail*, podendo ser utilizado de diversas formas, como por exemplo para envio de fotos, aviso de entrada no laboratório e registro de *log* remoto.

5.1.3 Regras Definidas

Para a execução dos testes da máquina de regras interagindo com o ambiente real, foram definidas quatro situações a serem reconhecidas, correspondentes a quatro eventos distintos, programados sob a forma de regras através da interface de configuração, como pode ser visto na figura 5.1.

O primeiro evento mostrado na figura corresponde a entrada de algum usuário do

The screenshot shows the 'ConvergeUFSM' configuration interface. The top right corner displays the user name 'Olá, Rodolfo Leffa de Oliveira' and links for 'Desconectar', 'Ajuda', and 'Sobre'. The left sidebar contains navigation options: 'Câmeras', 'Portas', 'Usuários', 'Configurações', and 'Informações'. Under 'Configurações', there are sub-menus for 'Regras' (with a sub-menu for 'Cadastrar', 'Editar/Excluir', 'Entrada', 'Saida', 'SenhaErrada', 'PortaAbriu') and 'Variáveis' (with sub-menus for 'Cadastrar', 'Editar/Excluir').

The main content area is titled 'Edicao de Configuracao'. It shows the configuration for a rule named 'SenhaErrada'. The 'Eventos' section is set to 'senhaErrada' with an operator '=' and a value of 'constante' and '3'. The 'Acoes' section has two actions: 'Chamar Script/Programa' with 'bateFoto' and 'Chamar Script/Programa' with 'enviaMall' and 'rodox@inf.ufsm.br Assunto'. Buttons for 'Alterar', 'Excluir', and 'Cadastrar Variavel Global' are visible at the bottom of the configuration area.

Figura 5.1: Regras definidas na interface de configuração

sistema no laboratório, através da digitação de seu *login* e senha no ramal telefônico da porta de acesso aos laboratórios. Quando o programa de validação gerar um evento informando que o *login* foi realizado com sucesso e o código digitado pelo usuário foi de entrada, as ações a serem tomadas são a de abertura da porta e de envio de um *e-mail*, utilizado aqui com a funcionalidade de ponto, registrando a hora de entrada do usuário.

A segunda regra programada representa a saída do usuário do laboratório, batendo o ponto aberto pela regra acima. O funcionamento desta regra é semelhante ao do evento de entrada no sistema, com a diferença que o evento recebido possui a informação de saída de ponto, e a ação a ser tomada se limita ao envio do *e-mail* registrando o horário de saída do usuário.

O terceiro evento que se deseja reconhecer é a digitação de três senhas erradas consecutivas no ramal telefônico, situação que pode ser uma tentativa de invasão aos laboratórios, ou a tentativa de um usuário de acertar a senha de outro. Então, no recebimento de um evento de senha errada gerado pelo programa de autenticação, a máquina de regras deve encaminhar para o tradutor de ações as ações de bater fotos e enviar um *e-mail* com

as fotos e o evento detectado. O programa que faz a autenticação dos usuários controla o número de senhas erradas digitadas, e envia um evento de senha errada quando três senhas erradas seguidas forem digitadas.

Por último, tem-se uma regra que define as ações a serem tomadas quando a porta de acesso aos laboratórios for aberta sem a autenticação dos usuários. Neste caso, as ações programadas são iguais as do evento acima (bater fotos e as enviar para um *e-mail*).

5.1.4 Resultados Obtidos

Uma vez definidas as regras contendo as condições e ações, a máquina de regras se encontra configurada e pronta para execução.

A máquina de regras executou durante aproximadamente dois dias para a observação dos resultados da interação desta com os sensores e atuadores do ambiente.

Os resultados observados mostram que o programa desenvolvido recebeu e analisou os eventos dos sensores, respondendo com as ações programadas. A figura 5.2 mostra uma ação de envio de *e-mail* tomada, relatando um evento de digitação de senha errada mais de três vezes.

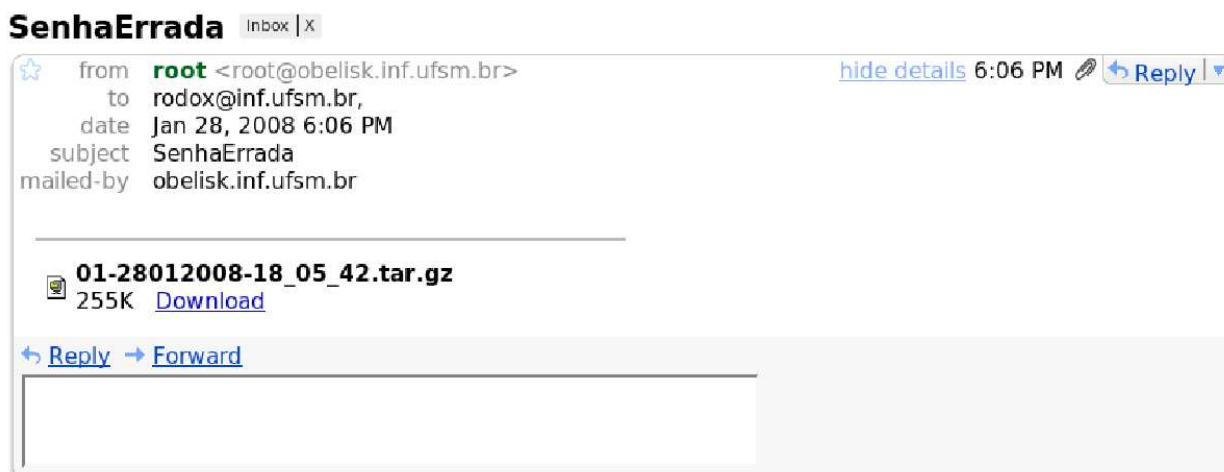


Figura 5.2: E-mail recebido para a regra de senha errada

Foi possível observar no experimento realizado que a máquina de regras implementada atendeu aos requisitos desejados. A interface de comunicação também pôde ser testada, e funcionou da maneira desejada. O mesmo ocorreu com a interface de configuração, e a facilidade de programar ou editar regras com esta foi evidente.

5.2 Simulação de Ambiente Complexo

O teste realizado com o sistema real do projeto CONVERGE, por conter poucos eventos e regras, e por estes serem simples, motivou a realização de um teste mais complexo. Este teste foi a simulação de um ambiente complexo, com várias regras cadastradas e vários eventos sendo recebidos pela máquina de regras.

O objetivo deste experimento é a avaliação do desempenho da ferramenta desenvolvida. O teste também pode ser utilizado como uma verificação de que a máquina alocada para ser o servidor do projeto CONVERGE consiga exercer sua função. Esta máquina tem a tarefa de executar os *drivers* dos sensores e atuadores, os *softwares* necessários ao projeto, como o Motion e o Asterisk, o portal *web*, além da máquina de regras propriamente dita, juntamente com os tradutores. O objetivo do projeto CONVERGE é gerar uma solução de baixo custo, se utilizando um computador comum para exercer essas tarefas. Assim, a avaliação da sobrecarga de execução do programa desenvolvido pode ser uma informação relevante.

5.2.1 Regras e Eventos Simulados

Para a execução do teste foram programadas cem regras, composta por vinte e cinco condições e cinco ações cada uma. Os eventos gerados foram todos iguais, com vinte e cinco sinais e valores. Assim, simulou-se o pior caso em tempo de processamento, ou seja, aquele em que todos os eventos são validados, tomando-se todo o caminho das informações dentro da máquina de regras. Se o evento simulado não fosse composto de somente condições verdadeiras, na primeira condição falsa o evento seria rejeitado para aquela regra, e pelo mesmo motivo para as demais.

5.2.2 Teste Realizado

O teste realizado foi uma sobrecarga da máquina de regras com eventos, afim de se medir quantos eventos ela consegue tratar em um determinado período de tempo. Deve-se notar que a máquina de regras foi executada no computador responsável pela execução de todos os componentes de *software* necessários ao projeto CONVERGE. Estes programas também estavam executando durante a interpretação dos eventos. Este computador é um *Pentium IV* de 1.6 GHz, com 512 *MegaBytes* de memória, rodando o sistema operacional Linux, distribuição Ubuntu.

O objetivo deste teste é verificar se o computador possui poder computacional suficiente para rodar todos os programas necessários ao projeto CONVERGE.

A metodologia utilizada para a medição dos tempos foi a coleta de *timestamps* antes e depois da interpretação de cada evento pela máquina de regras. Assim, se obteve o tempo em milisegundos necessário para o tratamento de um evento.

5.2.3 Resultados Obtidos

O resultado da execução desta simulação de eventos e regras complexas foi que mesmo com o computador em questão executando várias outras aplicações, ele consegue ter uma vazão suficiente de resposta a eventos. O tempo levado para se comparar cada evento com as cem regras definidas foi, na média, de 2,5 segundos.

O tempo obtido mostra que se eventos chegaram para a máquina, carregada com esta configuração, em um intervalo de tempo menor de que 2,5 segundos, provavelmente haverá atrasos na tomada de decisões, o que pode comprometer o funcionamento do sistema.

Ao mesmo tempo, se verificou que a solução implementada supera a exigência que o modelo real utilizado no ambiente de testes impõe. Este modelo possui regras simples, sendo muito mais rápida a interpretação destas frente aos eventos recebidos. Além disso, dificilmente se obtenha eventos com intervalos menores que o tempo de resposta do sistema obtido para esta configuração no ambiente real de testes.

Para se certificar que este tempo de interpretação das regras do modelo real é menor, o teste de desempenho foi aplicado também a este. A metodologia de avaliação utilizada para este teste foi a mesma. As regras programadas para interpretação foram as mesmas relatadas na primeira abordagem de avaliação, como os eventos.

O resultado da execução comprovou que o tempo de interpretação dos eventos perante as regras reais que definem o ambiente do projeto CONVERGE é muito menor que o teste complexo realizado. Este tempo foi em média de 0,13 segundos. Assim, somente se o sistema recebe eventos a uma taxa maior que este tempo ele atrasaria a tomada de decisões.

Como eventos não são gerados a uma frequência tão grande no ambiente real, mais uma vez se validou a ferramenta desenvolvida perante ao ambiente do projeto CONVERGE UFSM, desta vez no aspecto de desempenho da aplicação.

6 CONCLUSÃO

Este documento relatou a implementação de uma máquina de regras, com a funcionalidade de tomada de decisões com base em condições pré-programadas frente a ocorrência de eventos no ambiente. Estas decisões correspondem a tomada de ações, e tem por objetivo implementar um sistema de gerenciamento de ações de controle de acesso e alarme, com a integração entre várias tecnologias.

O programa corresponde ao núcleo de *software* do projeto CONVERGE UFSM, em desenvolvimento no Laboratório de Sistemas de Computação e no Grupo de Sistemas de Computação Móvel, ambos do curso de Ciência da Computação, da Universidade Federal de Santa Maria.

O processo de desenvolvimento desta ferramenta passou pelas fases de análise dos requisitos, projeto da ferramenta e implementação da mesma. Para atender as necessidades do projeto CONVERGE, também foram desenvolvidos programas auxiliares com o intuito de estabelecer uma interface de comunicação, e também foi projetada uma interface de configuração para o *software* desenvolvido.

A validação da ferramenta foi realizada pela integração desta com o ambiente que compõe o projeto CONVERGE, com sensores e atuadores. A ferramenta se comportou de maneira adequada, reconhecendo as regras programadas e acionando os atuadores devidos. Também foi realizado um teste de desempenho da ferramenta, com o intuito de se verificar se a utilização de um mesmo computador para executar toda a estrutura de *software* necessária ao projeto é possível. Os resultados deste teste apontaram que esta abordagem é factível.

Como trabalhos futuros, tem-se a melhoria no poder de expressão das regras, através da adição da possibilidade de inserir condições compostas por expressões lógicas ou aritméticas dentro das regras.

REFERÊNCIAS

ANTONIO A.F. LOUREIRO JOSE MARCOS S. NOGUEIRA, R. A. d. F. Redes de Sensores Sem Fio. In: SIMPOSIO BRASILEIRO DE REDES DE COMPUTADORES, 21., 2003. **Anais...** [S.l.: s.n.], 2003.

Digium. **Asterisk**. 2007.

GOODE, B. Voice over Internet protocol(VoIP). **Proceedings of the IEEE**, [S.l.], v.90, n.9, p.1495–1517, 2002.

KIRK, H. Use of decision tables in computer programming. **Communications of the ACM**, [S.l.], v.8, n.1, p.41–43, 1965.

POOCH, U. Translation of Decision Tables. **ACM Computing Surveys (CSUR)**, [S.l.], v.6, n.2, p.125–151, 1974.

POTTIE, G. J.; KAISER, W. J. Wireless integrated network sensors. **Commun. ACM**, New York, NY, USA, v.43, n.5, p.51–58, 2000.

RAPPAPORT, T. **Wireless Communications: principles and practice**. [S.l.]: IEEE Press Piscataway, NJ, USA, 1996.

ROCHA, C. C. d. **Desenvolvimento de uma Interface Web para Integração Configuração da Rede de Sensores-Atuadores do Projeto Converge UFSM**. Trabalho de Graduação - Curso de Ciência da Computação - UFSM.

RUIZ, L. B.; CORREIA, L. H. A.; VIEIRA, L. F. M.; MACEDO, D. F.; NAKAMURA, E. F.; FIGUEIREDO, C. M. Arquiteturas para Redes de Sensores Sem Fio. In: SIMPOSIO BRASILEIRO DE REDES DE COMPUTADORES, 22., 2004. **Anais...** [S.l.: s.n.], 2004. p.167 – 218.

RYU, K. et al. **Private automatic branch exchange**. US Patent 5,400,397.

SATYANARAYANAN, M. Pervasive computing: vision and challenges. **Personal Communications, IEEE [see also IEEE Wireless Communications]**, [S.l.], v.8, n.4, p.10–17, 2001.

SCHILIT, B.; ADAMS, N.; WANT, R. et al. **Context-aware Computing Applications**. [S.l.]: Xerox Corp., Palo Alto Research Center, 1994.

SCHWARTZ, S. Are there universal aspects in the structure and contents of human values. **Journal of Social Issues**, [S.l.], v.50, n.4, p.19–45, 1994.