

AEL: Asterisk Extension Language para criação de planos de discagem

Lilian Maria Gonçalves¹, Joaquim Quinteiro Uchôa²

¹Centro Federal de Educação Tecnológica de Cuiabá (CEFET Cuiabá)
BR 364 Km 329 – CEP 78106-970 São Vicente da Serra - Santo Antonio do Leverger (MT)

²Curso de Especialização em Administração em Redes Linux (ARL)
Departamento de Ciências da Computação – Universidade Federal de Lavras (UFLA)
Cx. Postal 3037 – CEP 37.200-000 Lavras(MG)

`lilika.mary@gmail.com, joukim@ginux.ufla.br`

Resumo. *Este artigo apresenta a ferramenta AEL (Asterisk Extension Language) para construção de planos de discagem, utilizando o Asterisk como ferramenta principal e também um comparativo entre o plano de discagem padrão e o AEL, além de documentar o mesmo.*

Abstract. *Abstract. This article presents the AEL tool (Asterisk Extension Language) to the building of dial plans, using Asterisk. The article also presents a comparative between dial plan standard and AEL, and documents the AEL too.*

Palavras-Chave: AEL; Asterisk Extension Language; Planos de Discagem; VoIP.

Keywords: AEL; Asterisk Extension Language; Dial Plans; VoIP.

1. Introdução

Asterisk é um *software* livre que, além de substituir uma central telefônica (PBX), possibilita fazer o controle das chamadas originadas, criar serviços com *menus* interativos, e interligar setores de empresas, sem custo com telefonia. Existem duas formas de se configurar a central telefônica no Asterisk conforme a necessidade e padrões das corporações: através do plano de discagem padrão e através do AEL (*Asterisk Extension Language*). O Plano de Discagem Padrão, também conhecido como *dialplan*, supre as necessidades do administrador, mas não tem a praticidade e boa visualização do contexto que está sendo escrito, pois trabalha com prioridades explícitas, restrição de aplicações na montagem do plano de discagem - como por exemplo a não existência de uma função que se assemelhe ao *switch* - repetições excessivas de palavras chaves entre outros.

Já o *Asterisk Extension Language* (AEL) possibilita uma visão, baseado em estruturas de programação, de como está sendo montado o plano de discagem, além de possibilitar uma maior flexibilidade, pois permite o uso de funções de programação na sua montagem como estruturas de repetição, condição e comparação. Dessa maneira, ele é uma ferramenta mais adequada para visualização e construção de uma central telefônica através do Asterisk, além de conter algumas funções que auxiliam na montagem de serviços como URAs, como as ferramentas semelhantes aos *cases*, criação de variáveis globais e locais, facilidade de saltos entre as sub-rotinas, através da função *goto*, entre outros recursos.

É importante destacar que nenhuma bibliografia consultada sobre o Asterisk, incluindo a documentação oficial, comenta sobre o AEL, nem a título de curiosidade para o leitor poder experimentar a ferramenta e ver se esta supre suas necessidades. Nas listas de discussão sobre o assunto, o AEL também é pouco referenciado, e apenas pelos usuários dessa ferramenta. Neste artigo foram consultadas as referências sobre o AEL e sua nova versão lançada no Asterisk 1.14 no *wiki*¹. Nesse *site*, o AEL está sempre citado como experimental. É importante difundir que existem outras formas de se utilizar o Asterisk e reforçar que o AEL deixou de ser experimental e já encontra-se em funcionamento para vários usuários da ferramenta, incluindo os autores deste texto.

Dada a situação exposta, este artigo tem como objetivo a apresentação da ferramenta AEL (*Asterisk Extension Language*) e mostrar o quanto esta ferramenta facilita na construção de planos de discagem utilizando a ferramenta *Open Source* Asterisk. Para isso, o artigo faz uma breve revisão sobre convergência de redes e o Asterisk. Em seguida, são apresentados o plano de discagem padrão e o AEL, demonstrando como são construídos os contextos, prioridades, extensões e aplicativos nesse mecanismo. Por fim, são levantados os pontos diferenciais entre os dois tipos de planos de discagem, demonstrando quais as vantagens de se utilizar o AEL em vez do plano de discagem padrão.

2. Convergência de Redes

Segundo (WIRTH, 2003), a voz humana é formada não só pela sua frequência, mas da combinação de sons elementares denominados fonemas, que são representados pelas letras. A frequência da voz humana ocupa a faixa de 100 Hz a 10000 Hz. A telefonia hoje, ainda segundo (WIRTH, 2003), utiliza a faixa de 300 Hz e 3400 Hz para comunicação da voz humana, para não ter problemas de atraso, conhecido como *delay* ou eco. Caso ocorra esses problemas citados acima, tem-se uma perda de qualidade, inviável atualmente nas operadoras de telecomunicações, produtoras de *hardware* e *software* voltadas para a telefonia e desenvolvedores de aplicativos para empresas como *call centers*, provedores de serviços entre outros.

Para a transmissão, a frequência da voz, emitida pela corda vocal, é codificada em sua origem para ser transmitida na rede telefônica e decodificada no seu destino. Existem várias técnicas para a codificação como os codificadores com base em forma de onda, citando como exemplo o ITU-T² G.729. Segundo (HERSENT *et al.*, 2002), esse codificador “é executado no domínio do tempo e segue uma abordagem amostra por amostra [...] usando a correlação entre as amostras do sinal de voz e se baseiam em quantizadores adaptativos e preditores adaptativos”. Outros exemplos de padrões de codificação são o ITU-T G.711, ITU-T G.723.1, ITU-T G.728 e ITU-T G.726.

Existem 3 tipos principais de estruturas hoje utilizadas dentro da convergência de redes: o VoIP (*Voice over IP*) (ARANGO *et al.*, 1999), o VOFR (*Voice over Frame Relay*) (BRADLEY; BROWN; MALIS, 1998) e o VOATM (*Voice over ATM*) (KUMAR, 2003). O VOFR utiliza o protocolo de *Frame Relay*, utilizado em comutação de pacotes através de WANs (*World Area Network*), interconectando as LANs (*Local Area Network*) das corporações. Ele se baseia na tecnologia X.25, retirando as exigências

¹Wiki: <http://voip-info.org>

²International Telecommunication Union: <http://www.itu.int/ITU-T>

de proteção e correção de erros, dando assim maior velocidade. Segundo (SOARES; FREIRE, 2002), para transmissão de voz em *Frame Relay*, utiliza-se a codificação G.764 (*Voice Packetization-Packetized Voice Protocol*) do ITU-T e o padrão FRF.11 (*Voice over Frame Relay*) do *Frame Relay Forum*.³

O VOATM utiliza *backbones* de alta velocidade, que oferece QoS (*Quality of Services*), ideal para transmissão de voz e vídeo. Mas essa tecnologia é de custo mais alto, ficando ao alcance apenas de corporações de grande porte. Em (SOARES; FREIRE, 2002) encontram-se as recomendações para o VOATM, padronizadas pelo ATM Forum⁴, que estão especificados nas parametrizações UNI 3.x e UNI 4.0 do ATM, baseados na sinalização de rede pública do ITU-T Q.2931.

O VoIP utiliza do protocolo da internet (IP) para poder fazer a transmissão de voz e dados, barateando assim o custo para as corporações de médio e pequeno porte, juntamente com as instituições de pesquisa e ensino. Isso ocorre porque elas não precisam alterar sua estrutura física para incorporar o VoIP e sim alterar alguns equipamentos para ganho de velocidade na rede e garantir a qualidade do sinal. Ao se adotar o VoIP, deve-se levar em consideração a tecnologia que se vai utilizar como: quais protocolos estarão rodando dentro da rede ou deste equipamento adquirido, se estão utilizando SIP (*Session Initiation Protocol*) (ROACH, 2002) ou H.323 (SCHULZRINNE, 2005); qual *hardware* a ser implantado; qual protocolo que será interfaceado com a operadora de telefonia; qual método utilizado para discagem. Sabendo qual o comportamento de cada item contido dentro dessas novas tecnologias adquiridas, é possível determinar a melhor solução a ser implantada.

Um problema existente é que devido ser uma tecnologia recente, alguns autores apontam a incompatibilidade entre alguns aplicativos ou equipamentos que utilizam tecnologias diferentes ou que demandam um trabalho adicional para quem está configurando para adequar e fazer com que ambos se comuniquem. Segundo Schulzrinne e Rosenberg (1998), aprimoradores dos protocolos SIP e H.323, há ainda muita interoperabilidade entre tecnologias recentes e as primeiras implementadas e principalmente na padronização pelas instituições como na ITU-T, citando como exemplo a padronização do H.323 e a não padronização ainda do SIP. Além disso, é possível encontrar incompatibilidade entre alguns equipamentos e sistemas operacionais, principalmente em sistemas para plataformas 64 *bits*, que são pouco utilizadas, como é o caso dos *drivers* encontrados nos pacotes das placas confeccionadas pela Digium⁵.

Um dos pontos importantes na utilização do VoIP é a possibilidade de uso de plataformas livres. A maioria das ferramentas utilizadas nas convergências são proprietárias, e também com limitações a sistemas proprietários. Adotando uma tecnologia desenvolvida com sistemas livres, tem-se a liberdade de confeccionar uma nova solução para esses problemas de incompatibilidade, além da redução de custo com licenças e outras ferramentas que podem encarecer o processo de convergência da rede.

³*Frame Relay Forum* é uma associação de usuários e consultores, hoje pertencente à MFA Forum (Associação que engloba profissionais que trabalham com redes ATM e *Frame Relay*), uma das principais responsáveis pelas padronizações de protocolos de *Frame Relay*

⁴<http://www.ipmplsforum.org>

⁵Digium: <http://www.digium.com/>

A principal ferramenta livre para VoIP é, com certeza, o Asterisk, um PBX Virtual (Sistemas de Ramais Privados), totalmente gratuito e que tem sua performance superior a ferramentas semelhantes encontradas em sistemas proprietários como o CTADE (*Computer Telephony Application Development Environment*), comercializado pela Intel. O Asterisk possui as mesmas funções e até mais recursos que o sistema proprietário mencionado, como é o caso de filas de espera, alocação de agentes de atendimento, *menus*, entre outros.

3. Apresentação do Asterisk

O Asterisk foi criado por Mark Spencer da Digium, a partir de suas necessidades de suporte comercial ao Linux, via telefone. Em 2001, após contato com Jom Dixon, pai do projeto Zapata de telefonia⁶, Spencer encontrou um modelo comercial para o Asterisk e junto com Dixon desenvolveu a primeira placa PCI para uso nesse *software*. A importância de Dixon deve-se ao fato de sua visão em utilizar a velocidade das CPUs da época para efetuar DSP (*Digital Signal Processing* - Sistema Digital de Processamento) através de *software*, ao invés de utilizar componentes DSP em placas de telefonia. Mais detalhes sobre o histórico do Asterisk podem ser encontrados em (MEGGELEN; SMITH; MADSEN, 2005).

O Asterisk é uma ferramenta *open source*, disponível em vários espelhos na internet e que consiste em um PBX Virtual, em que, ao invés de se utilizar um *chip* DSP, utiliza-se a própria CPU para digitalizar a voz e transmiti-la via rede de dados. Segundo (GONÇALVES, 2005), este processo de digitalização é realizado para cada canal de voz encontrado em um circuito disponibilizado pela operadora telefônica, conhecido também como E1. Uma das recomendações colocadas por (GONÇALVES, 2005) é que seja implementado o serviço de VOIP dentro de uma máquina específica com capacidade de processamento superior ao processamento de um computador doméstico, como por exemplo, computadores com tecnologia *dual core*. Entretanto, mesmo computadores com tecnologia inferior podem conseguir processar o sinal e assim transmitir uma voz de boa qualidade. Outro ponto importante é a quantidade de memória que faz-se necessário no processamento da voz. Recomenda-se para uma rede pequena, ou de testes, no mínimo 512 MB de memória RAM.

O Asterisk suporta vários tipos de padrões de protocolos, tanto com a interface com a operadora telefônica, como em sua transmissão de voz. Podemos fazer interface com as operadoras telefônicas, utilizando hardwares específicos como os da Digium⁷ ou não. Os protocolos, também conhecidos como sinalização, com interface das operadoras telefônicas são o ISDN (*Integrated Services Digital Network*) (MORNEAULT *et al.*, 2006), MCF/R2, AT&T 5ESS, SS7 (*Signaling System 7*) (MORNEAULT; PASTOR-BALBAS, 2006) e AT&T 4ESS. Já os protocolos utilizados para a transmissão de voz via rede de dados são SIP, Digium IAX (versões 1 e 2), H.323, MGCP (*Media Gateway Control Protocol*) (ARANGO *et al.*, 1999) e SCCP (*Skinny Client Control Protocol*) (COENE; PASTOR-BALBAS, 2006).

O Asterisk também suporta diversos codificadores da voz e/ou vídeo, como: ITU-T G.711 ulaw, ITU-T G.711 alaw, ITU-T G.723.1, ITU-T G.726, ITU-T G.729, GSM (*Glo-*

⁶Zapata Telephony Organization: <http://www.zapatatelephony.org/>.

⁷Digium: www.digium.com

bal System for Mobile Communications) (HAVERINEN; SALOWEY, 2006), iLBC (ANDERSEN *et al.*, 2004), entre outros. Dentro do Asterisk, também é possível configurar o *Speech Text*, recurso para transformar um texto em voz, ou mesmo reconhecimento de voz, recursos ainda pouco utilizados no Brasil. Com isso tem-se uma grande flexibilidade e compatibilidade com os equipamentos encontrados no mercado, juntamente com que é oferecido pelas operadoras telefônicas. E ainda é possível encontrar e/ou desenvolver mais ferramentas que possam agregar mais serviços.

4. Apresentação do Plano de Discagem Padrão

O Plano de Discagem do Asterisk, conhecido também como *dialplan*, flexibiliza a forma com que serão executadas e recebidas as chamadas no PBX. É possível ter um plano de discagem simples, onde são direcionadas todas as chamadas entrantes para um único ramal ou até a montagem de uma URA (Unidade de Resposta Audível). Segundo (MEGGELEN; SMITH; MADSEN, 2005) o *dialplan* “consiste de uma lista de instruções ou passos que o Asterisk irá seguir [...] completamente personalizável”. É encontrado geralmente no diretório `/etc/asterisk`, no arquivo `extensions.conf` e seu diretório pode variar conforme for instalado o Asterisk na distribuição escolhida pelo usuário. Conforme (MEGGELEN; SMITH; MADSEN, 2005), o *dialplan* separa-se em 4 (quatro) partes principais: contextos, extensões, prioridades e aplicações. As linhas do plano de discagem, são iniciadas por `exten=>`, com exceção dos contextos e macros.

4.1. Contextos

O contexto são identificações do *dialplan*, onde são especificadas para qual fim serão destinadas as extensões que são inclusas dentro do mesmo. Nele é possível separar as extensões conforme os setores da corporação, podendo-se implementar segurança, no que tange ao controle de chamadas interurbanas, visto que o Asterisk trabalha com autenticação de usuário, restringindo assim as chamadas. Também é possível rotear chamadas para outros contextos e/ou extensões, por exemplo. Pode-se dizer também, segundo (MEGGELEN; SMITH; MADSEN, 2005), que os contextos são os grupos das extensões. O contexto tem como sintaxe o nome do contexto compreendido entre um par de colchetes([]).

Um contexto importante, o contexto `[globals]`, é onde são adicionadas as variáveis que podem ser vistas em todos os contextos do *dialplan*. Sua sintaxe é: `<nomedavariável> =><valor da variável>`. Vale salientar que o Asterisk é *case sensitive*, ou seja, difere maiúsculo de minúsculo. Neste caso, precisa-se atentar para como está sendo colocada a variável para que não haja incompatibilidade quando for executar o *dialplan*.

4.2. Extensões

Também conhecidas como *extensions*, as extensões são as linhas que serão interpretadas pelo Asterisk durante sua execução. Segundo (GONÇALVES, 2005), as extensões são numéricas e podem conter símbolos como o # e *, pois esses caracteres são encontrados no teclado telefônico. Há alguns caracteres que são restritos, pois representam extensões padrão do Asterisk como o s (*start*), t (*time-out*), h (*hangup*). Também pode-se fazer a combinação entre o _ (sublinhado) e alguns caracteres que correspondem a um conjunto de caracteres e que assim não precisam ser especificados, Tabela 1.

Tabela 1. Padrões de Extensão – (GONÇALVES, 2005, p.170)

X	corresponde à qualquer dígito de 0-9
Z	corresponde à qualquer dígito de 1-9
N	corresponde à qualquer dígito de 2-9
[1237 – 9]	corresponde à qualquer dígito ou letra entre chaves
.	coringa, corresponde à um ou mais caracteres

4.3. Prioridades

As prioridades são definidas pela quantidade de linhas encontradas na extensão ou definidas pelo usuário, principalmente em casos que existam exceções, no caso de o ramal não atender, por exemplo. No *dialplan*, elas são encontradas após o número da extensão, sempre separados os campos através de vírgulas, como por exemplo: `exten=>123,1,Answer()`. As exceções, tem como padrão não seguir a sequência da numeração, mas acrescentar 100 (cem) juntamente com o início da sequência da numeração. Por exemplo: `exten=>123,101,Playback(demo-thanks)`.

4.4. Aplicações

Segundo (GONÇALVES, 2006), “dentro do plano de discagem, há diversas funções, as quais executam ações que encontraríamos em PBX analógicos ou em aplicações que utilizam plataformas de telefonia como o CTADE que utiliza placas Dialogic da Intel”. Essas aplicações, também chamadas de funções, executam as tarefas como atender a chamada, desligar e fazer discagem, ou seja, funções que são a rotina de um PBX analógico. Desta forma pode-se trabalhar com tecnologia digital em funções analógicas. Tem-se como exemplo: `Answer()`, `Hangup()`, `Dial()`.

5. Apresentação do Plano de Discagem AEL (*Asterisk Extension Language*)

O AEL é uma forma de desenvolver o *dialplan*, utilizando os mesmos conceitos encontrados no *dialplan* padrão, mas com outras características. O módulo do AEL é compilado e instalado separadamente do módulo que contém o *dialplan* padrão e é localizado dentro do pacote principal do Asterisk. Então, quando se instala o Asterisk, o AEL é instalado automaticamente. Quando o Asterisk é carregado, o AEL também é carregado automaticamente, não precisando fazer nenhuma outra configuração para utilizá-lo. Pode-se também configurar para que os módulos do AEL e do *dialplan* padrão sejam carregados manualmente. Para isso, deve-se alterar o arquivo `/etc/asterisk/modules.conf` na propriedade *autoload*.

As características do AEL podem ser consideradas como características de programação, mas não deixando de abranger comandos que incluem a operacionalidade de um PBX. Nele são encontradas as chaves principais, como o contexto, a extensão, a prioridade e a aplicação, mas dispostas de uma forma diferenciada, onde tem-se flexibilidade de ir de um ponto a outro da extensão, além de ir a outras extensões, sem precisar fazer repetições de palavras ou determinar prioridades diferenciadas. Segue exemplo da diferenciação entre os dois planos de discagem.

Dialplan padrão

```
[default]
exten=>1000,1,Answer()
exten=>1000,2,Playback(demo-thanks)
exten=>1000,3,Hangup()
```

AEL

```
context default {
1000 => {
begin:
    Answer();
    Playback(demo-thanks);
    Hangup();
};
};
```

Pode-se observar que a estrutura do dialplan continua a mesma, mas encontramos as sub-rotinas, onde podemos identificar o código, além de servir como referência no caso de uma nova execução do código compreendido naquela sub-rotina. Não é identificado prioridades explicitamente e não faz-se necessário a especificação da palavra chave *exten*. O AEL é encontrado no diretório `/etc/asterisk` e tem o nome `extensions.ael`. Como é visto, tem o mesmo nome do arquivo utilizado no plano de discagem padrão, mas com a extensão do arquivo diferenciada. Caso se opte por utilizar o AEL, o arquivo `extensions.conf` deve ser renomeado.

5.1. Contexto

Os contextos tem a mesma finalidade de um plano de discagem padrão e detém sintaxe diferenciada. Sempre iniciados com a palavra *context*, seguido pelo nome do contexto. Seu conteúdo é compreendido entre um par de chaves (). E também dispõe da mesma finalidade que é a separação do código em partes para melhor identificação do *dialplan*.

5.2. Extensão

A extensão no AEL é mencionada somente uma única vez, ao contrário de como é feito no plano de discagem padrão. Cada contexto pode ter *n* extensões e estas são identificadas primeiramente, e seu conteúdo é restrito dentro de um par de chaves e para finalizar a sintaxe das aplicações, utiliza ponto-e-vírgula. Exemplo:

```
_X650000 {
    Playback(demo-thanks);
};
```

Com isso, tem-se noção de até onde se estende o código da extensão, onde termina e onde começa. Outro fator interessante do AEL, é que é possível identificar o código também separando em sub-rotinas dentro da extensão, dando assim mobilidade para caso se necessite executar novamente uma parte da extensão. A partir dessas sub-rotinas, consegue-se ter uma visão de como o código será montado, definir as partes que serão compostas, e ainda ter uma facilidade em acrescentar uma nova parte no programa sem precisar acrescentar macros ou novos contextos. Exemplo:

```

_X650000 {
begin:
    Answer();
    Playback(demo-thanks);

acrescentabanco:
    sqlodbc(Query Insert into ligacao values (codigo,data,horario));
    Hangup();
};

```

5.3. Prioridade

No AEL não faz-se necessidade de identificar a prioridade da linha que será interpretada pelo Asterisk. Cada linha já é interpretada sequencialmente e através das sub-rotinas, podendo ser executado novamente, através do comando `goto <contexto>|extensão|sub-rotina`.

5.4. Aplicações

As aplicações são as mesmas executadas pelo *dialplan* padrão. Sua diferença é que ficam mais visíveis e é possível identificar melhor um erro de digitação, caso o haja, além de ter maior espaço para visualizar e acrescentar parâmetros ao comando, tanto os *defaults* (padrões que vêm com a função do Asterisk) ou os criados pelo usuário.

6. Discussão

Um dos pontos principais de diferenciação entre o *dialplan* padrão e o AEL, é que o AEL possui funcionalidades de programação para montar o *dialplan*, ganhando ainda mais flexibilidade e personalização. É possível, por exemplo, utilizar condicionais (*if*), similar ao usado na linguagem C, para uso na comparação de variáveis, tanto as padrões do Asterisk, quando as criadas pelo próprio usuário. As variáveis padrões do Asterisk são variáveis que contém informações vindas da rede telefônica, como números de binas, números de ramais, números de canais de voz, além de *status* provenientes de execuções de aplicações da própria ferramenta.

Além de condicionais, o AEL também disponibiliza comandos de repetições, como *while* e *for*, possibilitando executar quantas vezes for necessária a instrução localizada dentro da sub-rotina ou até mesmo dentro de uma extensão. Existem também operadores lógicos como o “*and*”, “*or*”, similares ao da linguagem C, e que são muito úteis no caso de comparativos de variáveis utilizadas dentro do plano de discagem, representados pelos símbolos `&` para *and* e `|` para *or*.

Outra ferramenta encontrada nas linguagens de programação e que são muito utilizadas nos planos de discagem utilizando-se do AEL é os *cases*, representados pelo comando *switch*. Um exemplo a ser dado de possibilidade de uso da função *switch* são os *menus* de interação com o usuário, como os leilões de menor preço hoje existentes nos programas de televisão. O usuário que ligou precisa escolher o produto em que vai dar o menor lance. No *dialplan* não existe a função *switch*, o que dificultaria ainda mais a confecção, pois além de ter as repetições das palavras-chave, trabalharia somente com *if* ou *gotoif*. No AEL, seria utilizado:


```

MenuIVR(5|1|opcaoprincipal|${VOX_DIR}sistema/mpopcaomenu);
switch(${opcaoprincipal}) {
  case 1:
    goto leilao|00|verificadastro;
    break;
  case 2:
    goto leilao|00|consultalance;
    break;
  case 8:
    Playback(${VOX_DIR}sistema/instrucoes);
    Set(tenta=0);
    goto leilao|00|menu;
    break;
  case 9:
    Playback(${VOX_DIR}sistema/regulamento);
    Set(tenta=0);
    goto leilao|00|menu;
    break;
  default:
    Playback(${VOX_DIR}sistema/opcaoincorreta);
    goto leilao|00|menu;
    break;
};

```

A função MenuIVR toca uma mensagem de Menu (digite 1 para tal coisa, digite 2 para outra coisa) e aguarda 5 segundos a pessoa digitar e somente pega um único dígito. Depois através da variável `${opcaoprincipal}` verificada no *switch* para ver onde será encaminhada a chamada do usuário. Esta estrutura é encontrada hoje no Leilão da Pechincha apresentado na TV Gazeta em Cuiabá, estado de Mato Grosso.

Um diferencial importante dentro do AEL é a divisão entre as rotinas encontradas dentro da extensão, permitindo *loops*, ou até executar novamente a sub-rotina, sem precisar voltar ao início da extensão, ganhando assim tempo de processamento da máquina. Um exemplo para apresentar bem a facilidade das sub-rotinas é o exemplo do *switch* citado acima. Apesar de ter que citar contexto, extensão e sub-rotina, o *switch* permite ir a qualquer outro contexto, extensão ou sub-rotina do AEL, alterando esses parâmetros, sem precisar iniciá-las a partir de uma prioridade diferente ou ter que criar uma macro, como seria no *dialplan* padrão. Exemplificado:

Dialplan padrão

```

[default]
exten=>1000,1,Answer()
exten=>1000,2,Playback(demo-thanks)
exten=>1000,3,Goto(1,default)
exten=>1000,4,Hangup()

```

AEL

```

context default {
1000 => {

```

```
begin:
    Answer();
    Playback(demo-thanks);
    goto default|h|begin;
};
};
```

Além das vantagens já citadas, a visualização do *log* também é mais fácil de ser percebida, pois a ordem das linhas do *log* é exatamente como foi executado pelo AEL, misturando apenas com os demais canais existentes, mas sempre na mesma ordem. Como é diminuído a quantidade de palavras chaves na montagem do plano, o *log* também fica mais fácil de ser compreendido.

Hoje, utilizando Asterisk com o AEL como padrão de plano de discagem tem-se o Clube Montreal de Viagens em Brasília-DF e o que foi implementado pela principal autora deste artigo na empresa RBC Tecnologia e Serviços em Cuiabá-MT. Na RBC, o AEL hoje é utilizado para atender uma demanda de 20000 chamadas por dia, em período integral, em um *call center* com 30 estações VOIP, utilizando como protocolo padrão o SIP. É utilizado tanto para fazer *telemarketing* receptivo como *telemarketing* ativo (pesquisas feitas para políticos, empresas comerciais e operadoras de telefonia) e também serve como conferência para funcionários e diretores da empresa que estão em viagens pelo país, a procura de novas prestações de serviço.

Outro serviço que está sendo implementado utilizando o AEL é a conferência entre usuários de origem e destino, sem a necessidade de ambos desligarem a chamada (ficando uma chamada única). Além destes, há também o Atendimento Automático Digital, também conhecido como URA, prestado para rádios de Mato Grosso do Sul e Mato Grosso, utilizando tanto a telefonia fixa através do protocolo MCF/R2 como em telefonia celular através do protocolo SS7. Um projeto futuro que está sendo idealizado é utilizar o Asterisk com o AEL para substituição da Central Telefônica do CEFET-Cuiabá (Centro Federal de Educação Tecnológica de Cuiabá) não só para fins de utilização telefônica, mas também interligação entre as unidades de ensino que serão construídas ao longo deste ano, além de trabalhar com a bilhetagem das chamadas e divulgação de resultados de concursos vestibulares e processos seletivos totalmente automatizadas.

7. Comentários Finais

A tecnologia atualmente está cada vez mais evoluindo para a convergência dos meios em que nos comunicamos, vivemos, trabalhamos. Essa tecnologia hoje é encontrada, principalmente na forma das plataformas computacionais tanto em residências como no meio corporativo. Por outro lado, Os sistemas *open source* hoje são uma grande tendência no mercado, principalmente no meio corporativo, pois trás além de sua flexibilidade, uma grande confiabilidade nos dados, principalmente no que tange segurança.

Nesse contexto, este trabalho apresentou o uso de AEL no Asterisk, um *software* livre para implementação de um PBX utilizando VOIP. Um dos grandes problemas em utilizar a ferramenta padrão de planos de discagem do Asterisk, PBX Virtual (Sistemas de Ramais Privados), é o excesso de repetição de palavras chaves, definição obrigatória de prioridades entre outros padrões que dificultam a personalização do plano de discagem, principalmente a visualização do que está sendo feito. Uma das formas de resolver esse

problema é utilizar a AEL (*Asterisk Extension Language*), que facilita a personalização e amplia a visão do que está sendo programado e dos parâmetros utilizados.

Entretanto, como comentado anteriormente, existe pouca informação sobre o AEL, inclusive na documentação oficial do produto. Por outro lado, o AEL tem-se mostrado bastante prático, o que foi comprovado através de seu uso prático. Esperamos portanto, que este artigo possa contribuir para ampliar sua divulgação e o poder dessa ferramenta, pois isso irá proporcionar ao Administrador outras ferramentas que irão ampliar suas formas de montar seu plano de discagem e criar novos serviços para atender sua clientela.

Referências

ANDERSEN, S.; DURRICK, A.; ASTROM, H.; HAGEN, R.; KLEIJN, W.; LINDEN, J. *Internet Low Bit Rate Codec (iLBC)*. Internet Engineering Task Force (IETF), 09 2004. (Request for Comments: 3951). Disponível em: <<http://www.ietf.org/>>.

ARANGO, M.; DUGAN, A.; ELLIOTT, I.; HUITEMA, C.; PICKETT, S. *Media Gateway Control Protocol (MGCP)*. Internet Engineering Task Force (IETF), 10 1999. (Request for Comments:2705). Disponível em: <<http://www.ietf.org/>>.

BRADLEY, T.; BROWN, C.; MALIS, A. *Multiprotocol Interconnect over Frame Relay*. Internet Engineering Task Force (IETF), 09 1998. (Request for Comments: 2427, STD 55). Disponível em: <<http://www.ietf.org/>>.

COENE, L.; PASTOR-BALBAS, J. *Telephony Signalling Transport over Stream Control Transmission Protocol (SCTP) Applicability Statement*. Internet Engineering Task Force (IETF), 02 2006. (Request for Comments: 4166). Disponível em: <<http://www.ietf.org/>>.

GONÇALVES, F. E. *Asterisk - Guia de Configuração*. Santa Catarina: V. Office Networking e Informática, 2005.

GONÇALVES, L. M. *Protocolos de Sinalização: H.323 e SIP*. Monografia (Pós Graduação em Redes de Computadores e Teleprocessamento), Cuiabá, 2006.

HAVERINEN, H.; SALOWEY, J. *Extensible Authentication Protocol Method for Global System for Mobile Communications (GSM) Subscriber Identity Modules (EAP-SIM)*. Internet Engineering Task Force (IETF), 01 2006. (Request for Comments: 4186). Disponível em: <<http://www.ietf.org/>>.

HERSENT, O.; GUIDE, D.; PETIT, J.; WIRTH, A. *Telefonia IP: Comunicação multimídia baseada em pacotes*. São Paulo: Prentice Hall, 2002. Trad. de Vilela Barbosa e Hugo Bastos de Paula.

KUMAR, R. *Asynchronous Transfer Mode (ATM) Package for the Media Gateway Control Protocol (MGCP)*. Internet Engineering Task Force (IETF), 01 2003. (Request for Comments:3441). Disponível em: <<http://www.ietf.org/>>.

MEGGELEN, J. V.; SMITH, J.; MADSEN, L. *Asterisk: O Futuro da Telefonia*. Rio de Janeiro: Alta Books, 2005.

MORNEAULT, K.; PASTOR-BALBAS, J. *Signaling System 7 (SS7) Message Transfer Part 3 (MTP3) - User Adaptation Layer (M3UA)*. Internet Engineering Task Force (IETF), 09 2006. (Request for Comments: 4666). Disponível em: <<http://www.ietf.org/>>.

MORNEAULT, K.; RENGASAMI, S.; KALLA, M.; SIDEBOTTOM, G. *Integrated Services Digital Network (ISDN) Q.921-User Adaptation Layer*. Internet Engineering Task Force (IETF), 01 2006. (Request for Comments: 4233). Disponível em: <<http://www.ietf.org/>>.

ROACH, A. B. *Session Initiation Protocol (SIP) - Specific Event Notification*. Internet Engineering Task Force (IETF), 06 2002. (Request for Comments: 365). Disponível em: <<http://www.ietf.org/>>.

SCHULZRINNE, H. *Session Initiation Protocol (SIP)- H.323 Interworking Requirements*. Internet Engineering Task Force (IETF), 07 2005. (Request for Comments: 4123). Disponível em: <<http://www.ietf.org/>>.

SCHULZRINNE, H.; ROSENBERG, J. A Comparison of SIP and H.323 for Internet Telephony. In: *Network and Operating System Support for Digital Audio and Video (NOSSDAV)*. Cambridge, Inglaterra: Bell Laboratories, Lucent Technologies / Columbia University, 1998. Disponível em: <<http://www.it.uc3m.es/~diederich/arc%20-%20h323-sip.pdf>>.

SOARES, L. C.; FREIRE, V. A. *Redes Convergentes: Estratégias para transmissão de voz sobre Frame Relay, ATM, IP*. Rio de Janeiro: Alta Books, 2002.

WIRTH, A. *Formação e Aperfeiçoamento Profissional em Telecomunicações e Redes*. 1. ed. Rio de Janeiro: Axcel Books, 2003.