

RSwing: uma biblioteca de componentes de código livre para geração de gráficos estatísticos

**Márcio de Medeiros Ribeiro, Felipe Barros Pontes,
Leandro Melo de Sales, Talita Perciano Costa Leite,
Alejandro C. Frery**

Departamento de Tecnologia da Informação – TCI
Universidade Federal de Alagoas – UFAL
57072-970 – Maceió – AL – Brasil

{mmr, fbp, lms, tpcl, frery}@tci.ufal.br

***Abstract.** The initial process of systems development consists in analyzing the requirements of the system. Many systems have the generation of graphics as a requirement, which is very important in order to analyze and visualize the data of the system. However, there are not enough open source tools that are easy to use in this context, which means that the task to implement the requirement previously defined can be difficult. This paper describes the specification and implementation of an open source library of components based on Java/Swing for statistics graphics generation. In addition, these components are showed as a case study in a multimedia environment, which is used in the context of distance education.*

***Resumo.** O processo inicial para o desenvolvimento de sistemas consiste no levantamento dos requisitos do mesmo. Nota-se que diversos sistemas têm como requisito a geração de gráficos, os quais são importantes na análise e na visualização dos dados do sistema em questão. Contudo, existem poucas ferramentas de código livre e de fácil uso para tal propósito, dificultando sobremaneira o cumprimento de tal requisito. Este artigo descreve a especificação e a implementação de uma biblioteca de componentes de código livre em Java/Swing para a geração de gráficos estatísticos. Adicionalmente, a utilização desses componentes em um ambiente multimídia para apoio a educação à distância é descrito como estudo de caso.*

1. Introdução

A criação de sistemas de software depende de diversos requisitos definidos pelos clientes e validados por seus desenvolvedores. Posteriormente ao processo de levantamento de tais requisitos, é natural que estes sejam transformados em programas de computador que visam atender as expectativas e exigências previamente definidas [Sommerville 2003].

Particularmente, alguns desses sistemas possuem como requisito a geração de gráficos, essenciais para representar dados e relatórios sob uma perspectiva mais intuitiva e interessante, pois sabe-se que grandes quantidades de dados são melhor representados em forma de diversos tipos de gráficos ao invés do uso de tabelas enormes [como discutido em Tufte 2001].

Atualmente, existem diversas abordagens para a concepção e construção de sistemas de software. No entanto, o uso de alguns métodos de desenvolvimento mostram-se inadequados, pois ocasionam altos índices de custo na etapa de desenvolvimento bem como dificuldades na evolução e manutenção do sistema. Para tentar contornar tais problemas, criou-se o desenvolvimento baseado em componentes, visando a construção de aplicações com características de sistematização de desenvolvimento, tais como escalabilidade e reuso [Szyperski 2002].

Assim, a disponibilização de componentes para geração de gráficos estatísticos pode ser de suma importância e valia para os desenvolvedores e usuários de aplicações que necessitam desta funcionalidade. Pressupondo-se que tais componentes foram desenvolvidos, testados e validados, os desenvolvedores não precisariam se preocupar com a criação dos mesmos, mas sim somente com a lógica do negócio da aplicação.

Este artigo apresenta uma biblioteca de componentes para geração de gráficos estatísticos chamada *R Swing*, que pode ajudar os desenvolvedores a gerar gráficos dentro de suas aplicações. A biblioteca é baseada na plataforma estatística *R* [R Project 2005] e foi desenvolvida para o ambiente *Java/Swing*.

O restante do artigo está organizado da seguinte maneira: a Seção 2 apresenta a contextualização do trabalho: a importância da utilização de gráficos em sistemas de software, a plataforma estatística *R*, o desenvolvimento baseado em componentes e a tecnologia *Swing* são discutidos. A Seção 3 descreve a biblioteca de componentes para geração de gráficos estatísticos *R Swing*. Na Seção 4, apresenta-se um estudo de caso: aplicação da biblioteca de componentes a um sistema multimídia. A Seção 5 discute os trabalhos relacionados e, por fim, a Seção 6 apresenta as considerações finais do artigo.

2. Contextualização

A utilização de gráficos vem se tornando uma ferramenta muito importante em diversos sistemas de software. Isto acontece porque utilizar gráficos é a melhor maneira de analisar grandes quantidades de dados e extrair informações relevantes dos mesmos. Além disso, é uma forma amigável de transmitir estas informações para os usuários.

Na grande maioria das vezes, a transmissão de informações dos sistemas para os usuários é feita através de tabelas com uma quantidade muito grande de dados, ou por outras formas de apresentação não muito atrativas. O papel dos gráficos é “enxugar” tais informações e traduzi-las para os usuários de maneira simples e intuitiva. Porém, é importante salientar que os gráficos utilizados devem ser construídos de forma adequada, pois de outra maneira estarão transmitindo informações irrelevantes ou até mesmo incorretas.

O trabalho aqui descrito propõe uma biblioteca de componentes para a tecnologia *Java/Swing* para geração de gráficos baseados no sistema *R*, uma plataforma estatística bastante utilizada para construção de gráficos de alta qualidade, como visto a seguir.

2.1. A Plataforma Estatística R

R é um ambiente para estatística computacional e produção de gráficos, sendo uma das plataformas mais disseminadas nesta área. A plataforma *R* provê uma grande variedade de técnicas estatísticas (modelagem linear e não linear, testes estatísticos clássicos,

análise de séries temporais, classificação, agrupamento) e gráficas, além de ser altamente extensível através da instalação de pacotes que provêm novas funcionalidades à plataforma.

Uma característica importante do ambiente *R* é a grande variedade de gráficos disponíveis. Além disso, os gráficos possuem listas de parâmetros que podem ser usadas para customizar os mesmos. A Figura 1 ilustra um Dendrograma (gráfico utilizado no contexto de agrupamento de dados), gerado pela plataforma *R*.

Ademais, *R* é um software livre e está disponível para uma grande variedade de ambientes computacionais (hardware e sistemas operacionais). *R* oferece ainda recursos de programação e conta com uma grande comunidade de usuários que lhe dá suporte.

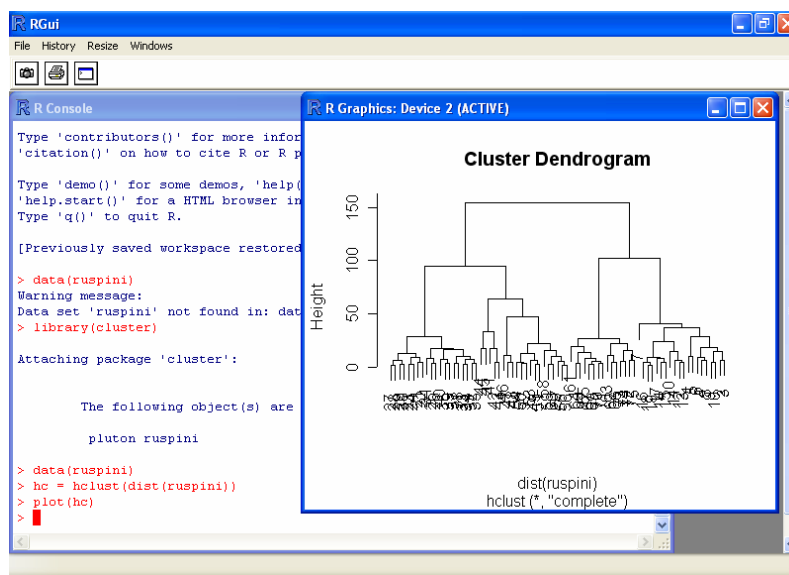


Figura 1. Exemplo de gráfico construído no *R*.

2.2. Desenvolvimento Baseado em Componentes

O desenvolvimento baseado em componentes trata da criação de componentes que possam ser reutilizados em outras aplicações. Várias vantagens são obtidas com essa abordagem: reutilização através da divisão do sistema em módulos, redução do tempo de desenvolvimento com a utilização de componentes pré-fabricados e gerenciamento das dependências dos módulos através da definição de componentes dependentes.

Nesse contexto, ao utilizar a biblioteca de componentes *RShiny*, o desenvolvedor que necessita de gráficos em sua aplicação não precisa conhecer pormenores da implementação dos mecanismos para a geração dos mesmos, preocupando-se, apenas, com a lógica do negócio de sua aplicação. Isto acontece porque cada componente encapsula suas funcionalidades específicas, abstraindo os detalhes de implementação do usuário do componente.

2.3. Swing

O *Swing* [Swing 2005] é um projeto *Java* que fornece um conjunto de componentes para criação de interfaces gráficas, tais como botões, caixas de texto e janelas. Escolheu-se implementar a biblioteca *RShiny* baseada na tecnologia *Swing* devido sua

alta portabilidade para com outras plataformas bem como sua ampla difusão em todo o mundo.

3. A biblioteca de componentes RSwing

A biblioteca de componentes *RSwing* é composta de diversos componentes visuais e não-visuais que ajudam sobremaneira o desenvolvedor a inserir gráficos em suas aplicações baseadas na tecnologia *Java/Swing*.

3.1. Arquitetura

A arquitetura da biblioteca de componentes *RSwing* é baseada em uma conexão com a plataforma *R* bem como no envio de dados para a mesma, como ilustra a Figura 2.

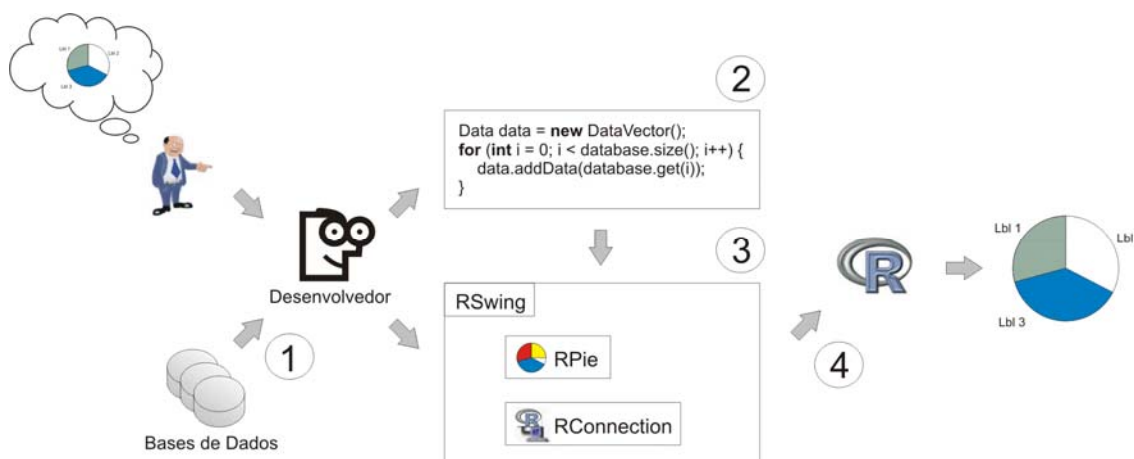


Figura 2. Arquitetura externa da biblioteca de componentes RSwing.

O processo para adicionar um gráfico a uma aplicação *Swing* utilizando a biblioteca de componentes *RSwing* é bastante simples: a partir de dados provenientes de determinada base (1), o desenvolvedor os formata de acordo com as especificações da biblioteca *RSwing* (2). Após esta etapa, basta utilizar o componente de conexão com a plataforma *R* bem como o componente escolhido para a geração do gráfico (3), fornecendo os dados necessários à construção do gráfico obtidos na etapa anteriormente descrita.

Nota-se que toda a lógica para a comunicação com a plataforma *R* está encapsulada nos componentes (4). Portanto, o desenvolvedor não precisa estar ciente de como a conexão acontece muito menos de peculiaridades sintáticas e semânticas da plataforma *R*.

3.2. Implementação da biblioteca de componentes RSwing

A biblioteca *RSwing* foi desenvolvida utilizando a linguagem de programação *Java* combinada com o modelo de especificação de componentes *JavaBeans* [JavaBeans 2005], idealizado pela *Sun* com o objetivo de resolver problemas de padronização. A idéia básica é que estes componentes, como em qualquer outro modelo, sejam altamente coesos, independentes e configuráveis, tornando possível a livre comercialização e reutilização dos mesmos.

De acordo com a especificação de *JavaBeans*, um componente (comumente denominado de *bean*) pode ser manipulado visualmente através de ambientes integrados de desenvolvimento tais como *Netbeans* e *JBuilder*. No contexto da especificação *JavaBeans*, tais ambientes são chamados de ferramentas construtoras e possuem implementações para suportar esta tecnologia. Portanto, as propriedades e comportamentos dos componentes podem ser configurados através da interface gráfica fornecida pelo próprio ambiente, facilitando sobremaneira o desenvolvimento das aplicações, além de ser um importante fator em quesitos como tempo e custos de implementação do sistema.

No contexto da implementação, cada componente *RSwing* em particular estende da classe abstrata *RGraphic*, que possui informações comuns a todos os gráficos. Adicionalmente, esta classe define métodos abstratos que precisam ser implementados por cada componente que representa um gráfico em especial, pois cada um possui uma maneira peculiar de formatar os dados e os comandos *R* para a geração de seu respectivo gráfico.

A Figura 3 ilustra o diagrama de classes da biblioteca de componentes *RSwing*.

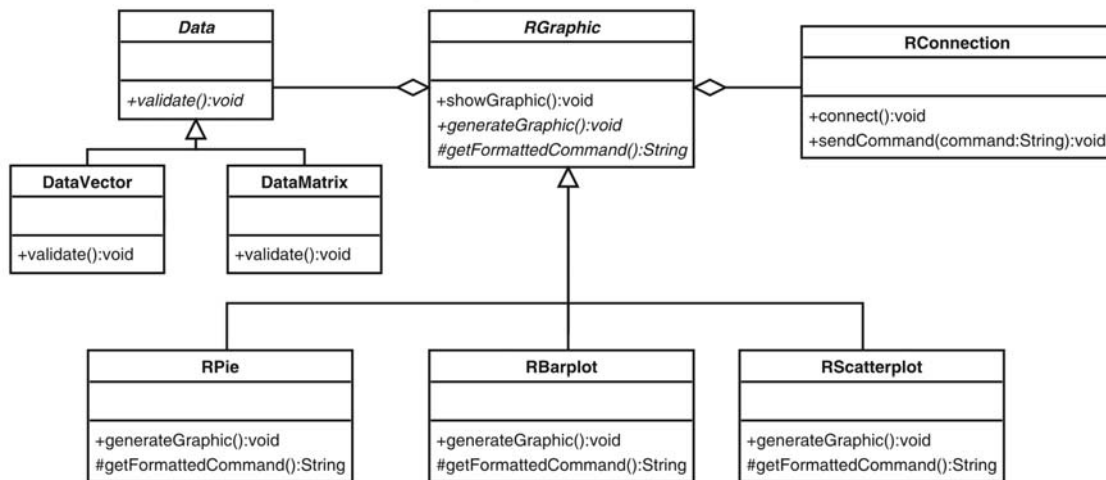


Figura 3. Diagrama de classes da biblioteca de componentes RSwing.

De modo a oferecer uma maior flexibilidade à biblioteca, criou-se a classe abstrata *Data*, onde qualquer objeto que represente um tipo de dado necessário à geração dos gráficos deve estender tal classe. Portanto, pode-se, quando oportuno, inserir um novo tipo de dado sem afetar a infra-estrutura já existente.

Os tipos de dados existentes atualmente são *DataVector* e *DataMatrix* e cada um define uma política de validação de dados. Essa validação é obrigatória, visto que a mesma encontra-se definida na classe abstrata *Data*; por outro lado, a utilização dessa validação pode ser de suma importância no quesito performance: como a validação é feita antes da conexão com a plataforma *R*, pode-se evitar o envio de seqüências de comandos para a mesma ao se perceber que os dados foram definidos erroneamente.

Para conectar os componentes à plataforma *R*, utilizou-se o acesso em modo *batch* da mesma, onde uma seqüência de comandos é enviada e posteriormente processada pela plataforma. Essa seqüência de comandos consiste no envio dos dados bem como em chamadas a funções *R* para a geração dos gráficos.

3.3. Componentes

Os componentes da biblioteca *RSwing* são baseados nos tipos de gráficos oferecidos pela plataforma *R*. Assim, foram desenvolvidos componentes para gráficos tais como Pizza, Barras, Boxplot, Scatterplot e Dendrograma. É importante salientar que cada tipo de gráfico possui seus parâmetros.

Abaixo, tem-se uma lista de alguns componentes visuais da biblioteca *RSwing*, ou seja, aqueles que são visíveis no ambiente de desenvolvimento.

- **RConnection** – componente de conexão entre a aplicação e a plataforma *R*;
- **RPie** – componente para geração de gráficos Pizza;
- **RBar** – componente para geração de gráficos em Barras;
- **RBoxplot** – componente para geração de gráficos do tipo Boxplot;
- **RScatterplot** – componente para geração de gráficos do tipo Scatterplot;
- **RDendrogram** – componente para geração de Dendrogramas.

A Figura 4 ilustra o IDE (*Integrated Development Environment*) Netbeans com os componentes visuais do *RSwing* descritos acima. O destaque através de um círculo representa os componentes da biblioteca *RSwing*, ao passo que o retângulo representa os parâmetros de cada componente. Estes parâmetros podem ser utilizados para refletir determinadas características e comportamentos no gráfico, tais como cores, títulos, dimensões, títulos de eixos, entre outros.

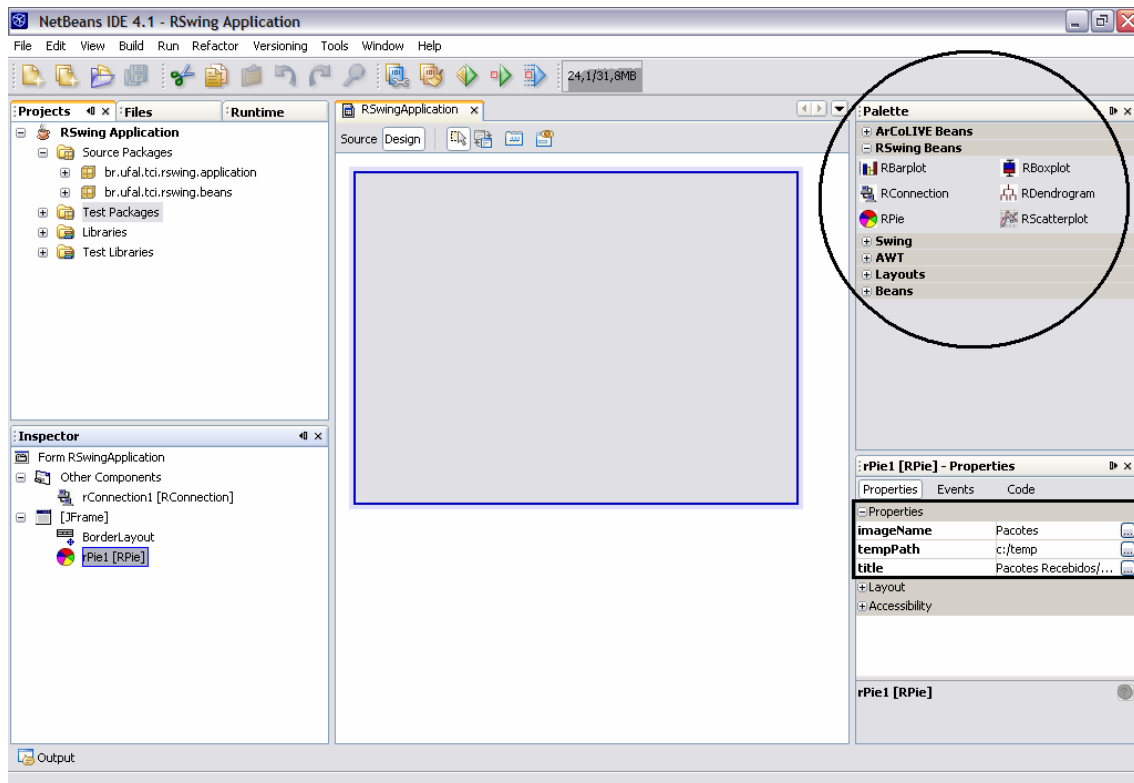


Figura 4. Biblioteca de componentes *RSwing* no IDE Netbeans.

A biblioteca dispõe ainda de componentes não-visuais, os quais servem para encapsular os dados da aplicação necessários à geração dos gráficos:

- **DataVector** – componente para a criação de vetores de dados. Estes vetores são usados na geração de gráficos tais como Pizza, Barras e Boxplot.
- **DataMatrix** – componente para criação de matrizes de dados. Estas matrizes são usadas na geração de gráficos tais como Scatterplot e Dendrogramas.

Os componentes supracitados nesta seção foram submetidos a uma bateria de testes para validar a implementação dos mesmos. Além disso, a biblioteca de componentes RSwing foi utilizada em uma aplicação para videoconferência com o objetivo de analisar os dados de conexões da rede.

4. Estudo de caso: aplicação do RSwing a um sistema de videoconferência

Os sistemas de videoconferência têm se tornado populares em ambientes empresariais, cibercafés, em residências ou em qualquer lugar onde uma conexão de banda larga esteja disponível. O principal motivo deste rápido crescimento deve-se à grande disponibilidade de ferramentas, muitas delas gratuitas. É fácil, por exemplo, utilizar sistemas de comunicação através da Internet: basta fazer o download, utilizar um microfone e uma câmera e o usuário transforma o computador em um equipamento de multimídia através de voz e imagem que supera os tradicionais meios de comunicação disponíveis.

Para este tipo de software, muitas vezes é de fundamental importância a utilização de recursos que mostrem de modo simples e intuitivo o estado da conexão. Assim sendo, informações acerca da porcentagem de perda de pacotes na rede ou quantos pacotes foram enviados e recebidos podem auxiliar sobremaneira em algumas decisões do usuário, tais como saber se é necessário aumentar o link de sua conexão ou solicitar algum reparo ao suporte técnico do provedor de acesso que está sendo utilizado. Além disso, estes recursos têm sido usados por pesquisadores como base na coleta de dados da rede e na geração de estatísticas intrínsecas da mesma, fornecendo meios de analisar e melhorar os serviços oferecidos pelo sistema.

Neste contexto, o pacote *RSwing* foi aplicado a um ambiente cujo propósito é disponibilizar serviços de transmissão de áudio/vídeo e ferramentas de colaboração tais como chat, quadro branco e compartilhamento de arquivos através da Internet, uma vez que percebeu-se a ausência recursos gráficos e estatísticos neste ambiente. O sistema em questão é o *ArCoLIVE - ArCo Live Internet Videoconference Environment* [ArCoLIVE 2005], o qual está desenvolvido na linguagem *Java*, segue a arquitetura cliente/servidor e utiliza essencialmente a pilha TCP/IP da Internet para controle de conexão e transmissão de fluxo de dados. A concepção deste ambiente foi também baseada em componentes, idêntico ao paradigma utilizado no *RSwing*.

A Figura 5 ilustra o cliente *ArCoLIVE* utilizando o componente *RPie*, projetado para gerar gráficos do tipo Pizza. Os dados são coletados da rede utilizando o componente *ArCoLIVEConnectionManager*, que fornece meios de obter dados como pacotes recebidos e perdidos através da conexão em questão. Neste caso, de acordo com a Figura 2, o componente *ArCoLIVEConnectionManager* age como a base de dados que alimenta o componente *RPie*.

Portanto, o componente RPie é utilizado para gerar um gráfico em Pizza que representa a porcentagem de pacotes recebidos e pacotes perdidos na rede. Para esta aplicação em particular, o gráfico é atualizado de 5 em 5 segundos. Este gráfico é bastante útil já que pode-se perceber problemas com a conexão da rede se a porcentagem de pacotes perdidos crescer significativamente, por exemplo.

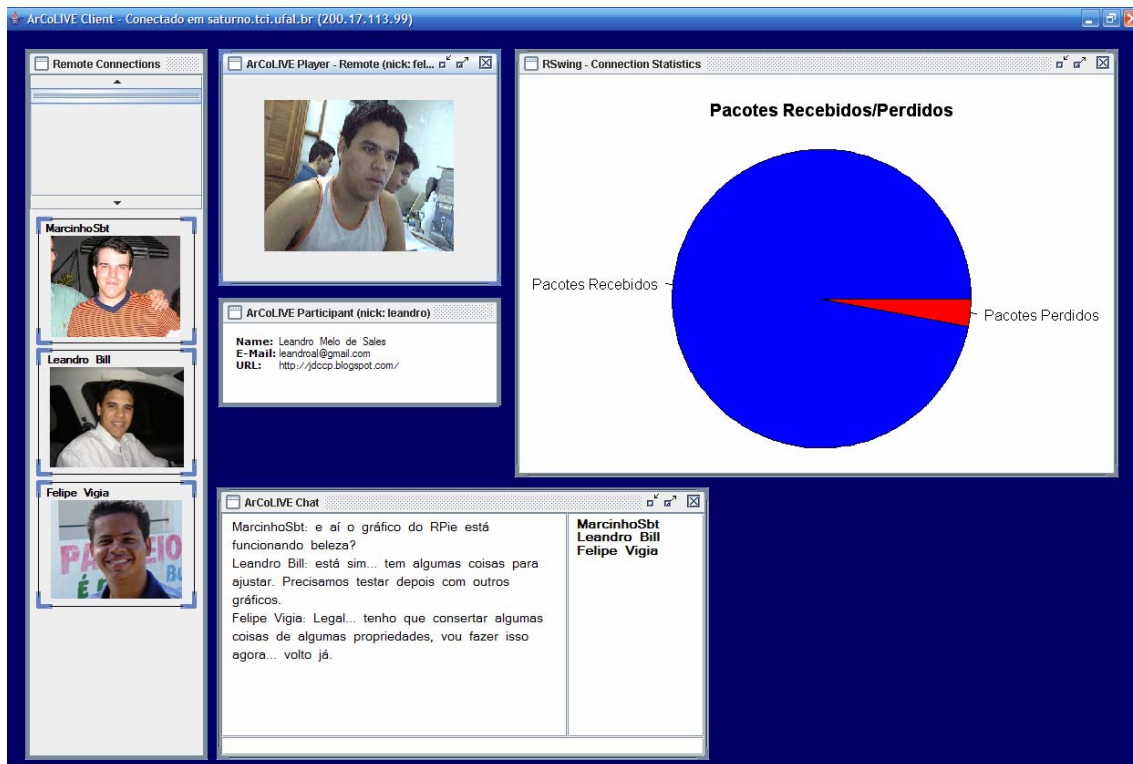


Figura 5. Cliente ArCoLIVE utilizando o componente RPie do RSwing.

5. Trabalhos Relacionados

Na literatura podem ser encontradas algumas bibliotecas que permitem a construção de gráficos para serem adicionados em sistemas. Nesta seção são abordadas duas bibliotecas e estas são comparadas com a biblioteca *RSwing*.

5.1. JFreeChart

JFreeChart é uma biblioteca de código aberto para construção de gráficos em *Java* bastante popular que gera a maioria dos tipos de gráficos conhecidos, incluindo gráfico de Pizza, de Barras, de Linhas entre outros. Ademais, esta biblioteca possui algumas ferramentas interativas e fornece a possibilidade de adicionar gráficos no *Swing*.

Apesar das boas características citadas anteriormente, a biblioteca possui algumas desvantagens. A codificação que o desenvolvedor precisa escrever para criar gráficos usando o *JFreeChart* pode ser complexa e trabalhosa, principalmente na construção de gráficos mais elaborados. Além disso, o *JFreeChart* não é baseado em componentes e não possui a alta qualidade estatística do *R*.

5.2. NetCharts Pro

NetCharts Pro é uma ferramenta para criação de componentes gráficos em *Java*. Esta ferramenta pode ser utilizada para adicionar gráficos em aplicações baseadas na Web ou aplicações *standalone*. *NetCharts Pro* cria imagens de gráficos em formatos Web populares como *PNG*, *JPG* e *BMP*, tornando a ferramenta ideal para utilização no lado servidor de uma aplicação que usa servlets ou páginas *JSP*.

NetCharts Pro é uma ferramenta bastante robusta e é utilizada principalmente por profissionais de grandes empresas, agências governamentais e outras organizações que precisam visualizar e interpretar dados pela Internet. Contudo, diferentemente da biblioteca de componentes *R Swing*, *NetCharts Pro* é paga e seu código não é livre.

6. Considerações Finais

O trabalho aqui exposto propôs uma biblioteca de componentes de software para geração de gráficos estatísticos de alta qualidade providos pela plataforma *R*, através de uma codificação simples e portátil. Cada componente da biblioteca dispõe de propriedades que podem ser utilizadas para customizar os gráficos através de características como cores, títulos, dimensões, entre outros. Adicionalmente, o uso da biblioteca pode ser facilitado através de uma interface amigável fornecida pelos ambientes integrados de desenvolvimento que suportam a tecnologia *JavaBeans*.

Mostrou-se também uma aplicação da biblioteca *R Swing* a um sistema de videoconferência, sendo de grande valia para os testes e a validação da implementação da biblioteca, além de demonstrar a viabilidade do uso da mesma.

Como trabalhos futuros, pretende-se ampliar a biblioteca de componentes com novos gráficos fornecidos através de pacotes adicionais da plataforma *R*. Além disso, pretende-se adicionar à biblioteca plataformas similares ao *R* tais como *Scilab* [Scilab 2005] e *Gnuplot* [Gnuplot 2005], o que, conseqüentemente, acarretará na mudança do nome da mesma.

Referências Bibliográficas

- ‘Gnuplot’. (2005) Última consulta em outubro de 2005.
<http://www.gnuplot.info/>
- Hornik, K. (2002), ‘The R Project for Statistical Computing’. Última consulta em outubro de 2005. <http://www.r-project.org>.
- ‘JavaBeans’. (2005) Última consulta em outubro de 2005.
<http://java.sun.com/products/javabeans/>
- ‘JFreeChart’. (2003) Última consulta em outubro de 2005.
<http://www.jfree.org/jfreechart/index.php>.
- ‘NetCharts Pro for Visual Mining’. (1999) Última consulta em outubro de 2005.
<http://www.visualmining.com/products/netcharts-pro.shtml>.
- Sales L. M., Pontes, F. B., Costa, E. B., Luna, H. P. (2005). Set your Multimedia Application Free with ArCoLIVE: an Open Source Component-based Toolkit to Support E-Learning Environments. A ser publicado no Web-based Education 2006.

'Scilab'. (2005) Última consulta em outubro de 2005.

<http://www.scilab.org/>

Sommerville, I. (2003), Engenharia de Software, 6 edn, Addison Wesley.

'Swing'. (2005) Última consulta em outubro de 2005.

<http://java.sun.com/products/jfc/>

Szyperski, C., Gruntz, D., and Murer, S. (2002). Component Software, Beyond Object-Oriented Programming. 2 edn, ACM Press.

Tufte, E. R. (2001), The Visual Display of Quantitative Information, 2 edn, Graphic Press.