

## Ferramenta de auxílio no ensino de Organização e Arquitetura de Computadores: extensão Ptolemy para fins educacionais

André L. L. Torres, Alisson V. Brito

*Universidade Federal da Paraíba*

*Centro de Informática (CI)*

*Programa de Pós-Graduação em Informática (PPGI)*

*{atorres, alissonbrito}@ci.ufpb.br*

### Resumo

*A utilização de simuladores na educação se faz cada vez mais presente nas atividades pedagógicas. Nesta nova realidade, várias aplicações visam facilitar a construção do conhecimento por parte dos professores em relação aos alunos, através de métodos dinâmicos e expondo aulas para múltiplos ramos sem haver grande esforço ou repetições desnecessárias. Pois, se tem observado que os ensinamentos de alguns conceitos introdutórios em disciplinas essenciais costumam apresentar um nível de abstração que prejudica o aprendizado dos alunos de cursos de informática que já possuem uma grande dificuldade em lidar com disciplinas da área de hardware. Neste sentido, este trabalho apresenta os resultados alcançados com a aplicação de uma extensão desenvolvida numa ferramenta de modelagem e simulação de sistemas concorrente baseada em atores, denominada Ptolemy. A extensão foi criada para contribuir com o processo de ensino-aprendizagem da disciplina de Organização e Arquitetura de Computadores com alunos da graduação.*

### 1. Introdução

A análise dos sistemas computacionais atuais, seu entendimento é uma tarefa difícil, principalmente embasado apenas na teoria. Logo, o estudo da disciplina Organização e Arquitetura de Computadores podem ser facilitados com a utilização de simuladores [3], sendo uma abordagem benéfica para que os alunos entendam seu comportamento e possam visualizar a execução de programas para montagem real de uma arquitetura.

Existem muitas técnicas de ensino utilizando simuladores para representar arquiteturas reais ou hipotéticas [2]. Estas técnicas, porém, não permitem uma abordagem construtivista onde os alunos possam criar suas próprias arquiteturas.

Neste sentido, esse trabalho apresenta os resultados obtidos com a aplicação da extensão desenvolvida no trabalho [9] e sua contribuição com o processo de aprendizagem na disciplina de Organização e Arquitetura de Computadores com alunos de graduação. Para tal, o trabalho utiliza a modelagem e simulação de sistemas concorrentes baseada em atores, utilizando um Framework denominado Ptolemy [6]. A metodologia abordada possui um grande potencial educacional ainda pouco explorado. Onde um modelo de atores é definido como um modelo matemático de computação concorrente que trata “atores” como sendo os elementos primitivos da computação concorrente digital [6].

A inclusão do experimento e suas atividades práticas na disciplina de Arquitetura de Computadores vêm suprir uma necessidade dos alunos em fixar melhor o aprendizado, simulando situações reais em que elas acontecem. Além disso, estas atividades práticas podem servir como motivadores iniciais dos tópicos a serem estudados dando um estímulo aos alunos que os levará a entender melhor os assuntos.

A Ferramenta receberá mais funcionalidades e espera-se incorporá-lo em atividades práticas em tópicos curriculares na ementa da disciplina de Arquitetura de Computadores. Esse novo ambiente proporcionará ao aluno a um ambiente que permita a eles a criação dos seus próprios modelos computacionais, além da possibilidade de combinarem novos elementos, tornando-os elementos ativos na formação do conhecimento [2]. Logo, os alunos podem criar seus próprios modelos de arquitetura de maneira simples, interativa e didática.

### 2. Trabalhos Relacionados

Os simuladores computacionais são recursos tecnológicos que podem auxiliar em diversas áreas. No contexto abordado neste trabalho, os simuladores auxiliam os professores a provocar, favorecer e

orientar situações controladas de ensino e aprendizagem, pedagogicamente interessantes para o nível de ensino que deseja. Do ponto de vista pedagógico, simuladores computacionais são softwares úteis e simples, uma vez que levam os alunos a interagirem com os mesmos, o que lhes facilita a compreensão de conceitos abstratos, leis e características do mundo físico.

Segundo Felix [4], em tal disciplina é notável a necessidade de ferramentas gráficas e didáticas para auxiliar o aprendizado. Especificamente para explicar a arquitetura de processadores, pois são necessárias muitas figuras para ilustrar os caminhos de dados e de controle quando uma instrução é executada.

Para a disciplina de Organização e Arquitetura de Computadores, podemos citar algumas ferramentas voltadas para o campo da informática educativa, especificamente em uma das áreas correlatas (processadores) ao tema da pesquisa. Dentre os vários processadores, dos mais simples aos mais complexos, destacamos o processador MIPS (*Microprocessor without Interlocked Pipe Stages*). O MIPS é uma arquitetura de microprocessador projetada com objetivo de oferecer alto desempenho na execução de código compilado. Suas instruções são todas de 32 bits, mesmo nos processadores de 64 bits, como forma de manter a compatibilidade. Todas as operações aritméticas no MIPS utilizam valores contidos nos registradores e os resultados também são colocados em um registrador.

O WebMIPS [1] possui uma plataforma gráfica visualmente agradável para simular a arquitetura MIPS com pipeline. Ele possui a vantagem de ser uma plataforma disponível na Internet, proporcionando uma interface dinâmica, a qual o estudante pode ter acesso sem ter que instalar módulos do simulador.

O DIMIPSS [4] (*Didact Interactive MIPS Simulator*) é um software multiplataforma de simulação da execução (caminho de dados e de controle) das instruções do MIPS Monociclo. Ele recebe um programa em linguagem de montagem (assembly), o converte para a linguagem de máquina e representa graficamente o comportamento do caminho de dados e de controle durante a sua execução.

O MARS (*Mips Assembly and Runtime Simulator*) desenvolvido por Vollmar e Sanderson [10] é uma ferramenta em Java com a IDE gráfica implementada usando Swing que simula um conjunto de instruções do MIPS32. A versão atual é a 3.8 e implementa 98 instruções, 36 pseudo-instruções e 17 chamadas de sistemas.

Outra ferramenta é o WinMIPS64 [8] que simula a arquitetura pipeline do microprocessador MIPS64, apresentando um conjunto de 32 registradores para

números inteiros, e 32 para números representados em pontos flutuantes.

O WinMIPS64 [8] possibilita a visualização da instrução passando pelos pipelines, e ainda estatísticas relativas à execução, como, por exemplo, quantidade de ciclos que determinada instrução consumiu para passar por todos os estágios do pipeline.

E por último descrevemos o MipsIt [3] que é um ambiente de desenvolvimento integrado e que também realiza a simulação do processador MIPS, possibilitando a programação em alto nível e permitindo relacionar conceitos de programação com organização e arquitetura de computadores.

A Tabela 1 descreve uma análise comparativa entre ferramentas de simulação. Os tópicos abordados baseiam-se nos estudos de [7] para avaliar o potencial educacional dos simuladores.

Tabela 1: Análise comparativa entre as ferramentas no ensino de Arquitetura de Computadores

Ferramenta	A	B	C	D	E	F
WebMIPS	AA	N	S	IL	S	S
DIMIPSS	BA	N	S	IL	N	N
MARS	BA	N	S	PL	N	S
WinMIPS	BA	N	N	IL	S	S
MipsIt	AA	N	S	IL/PL	N	S
Ptolemy	BA/AA	S	S	CL/IL	N	S

Legenda:

- (A) Ambiente e Complexidade
- (B) Suporte ao Projeto
- (C) Apresentação Visual
- (D) Nível de Simulação
- (E) Ensino a Distância
- (F) Detalhamento da Implementação

Onde (A) avalia se o simulador apresenta noções básicas da introdução à Organização e Arquitetura de Computadores. Os níveis de avaliação são Arquitetura Básica (BA) e Arquitetura Avançada (AA). O critério (B) avalia se o simulador inclui ferramentas de apoio à elaboração de módulos informatizados reutilizáveis ou apenas simula sistemas pré-definidos.

Para o critério (C) existem aqueles simuladores que detalham visualmente o funcionamento interno de um sistema computacional seguindo um fluxo de dados, ou aqueles onde os resultados são apresentados de forma textual, com a possibilidade de seu pós processamento. Ainda sobre os critérios utilizados, o (D) diz respeito ao nível de detalhamento do comportamento de um objeto exposto, com base no nível mais baixo de granularidade suportado durante uma seção de simulação. Podendo ser: Nível de Clock (CL), Nível de Instrução (IL) ou Nível de Programa (PL). Como exemplo, o nível (IL) tem o objetivo primário de apresentar os detalhes da implementação e tem a característica de visualização textual e do fluxo

de simulação. Para o critério (E), com base na disponibilidade de suporte para ensino à distância, os simuladores podem ser divididos entre aqueles com e sem suporte para o critério. E finalizando o (F) avalia se os simuladores fornecem detalhes da execução visíveis durante a simulação.

Após a análise das ferramentas verificou-se que o Ptolemy é o único que possibilita suporte ao projeto (B), pois permite que o aluno crie seus próprios elementos e possa reutilizá-los em diferentes níveis de abstração, como também em outros projetos e arquiteturas. Além disso, o Ptolemy possui algumas vantagens em relação às outras ferramentas, como por exemplo: i) no WebMIPS [1] as modificações que ocorrem nos caminhos de dados e controle durante a execução das instruções não são apresentadas; ii) O DIMIPSS [4] não é uma ferramenta web, dificultando o ensino a distância. Além de só implementar um pequeno número de instruções do MIPS monociclo; iii) o MARS não possui suporte a depuração e não permite que o usuário desfaça etapas de execução; iv) o WinMIPS64 [8] não possui uma área para edição do código de execução pelo usuário; v) já o sistema MipsIt [3] a arquitetura do processador MIPS, ilustrada no MipsIt, possui complexidade acima do desejável para ilustração de conceitos aos alunos que estão ingressando em cursos de Computação.

Podemos concluir que apesar do framework Ptolemy não ser um simulador didático (não possuindo um procedimento para o ensino a distância e de ter uma interface gráfica complexa), ele oferece a funcionalidade de criação de um repositório de atores, possibilitando ao aluno criar seus próprios cenários, como também seus atores, além de ser uma ferramenta gratuita e com o código fonte aberto, possibilitando que novas funcionalidades e mudanças na sua interface sejam realizadas para torná-la uma ferramenta apropriada para fins educacionais.

### 3. Framework Ptolemy

Este trabalho se baseia no uso do Ptolemy [6], criado e mantido na Universidade de Berkeley, como uma ferramenta de ensino de Organização e Arquitetura de Computadores.

O Ptolemy não é uma ferramenta preparada para o auxílio ao ensino de Organização e Arquitetura de Computadores. Entretanto, como ele é uma ferramenta de código aberto, as modificações sugeridas buscam torná-lo uma ferramenta educacional.

Para verificar se uma ferramenta atende os requisitos para ser caracterizada como um software educacional, algumas características de qualidade técnica e aspectos educacionais devem ser levantados,

como: viabilidade de sua utilização em situações educacionais, adaptabilidade a metodologia de ensino e adequação a uma proposta de educação mais construtivista. Mas para conseguir êxito em tal objetivo, foram verificadas algumas barreiras para a utilização da ferramenta: i) Ptolemy é um software para fins científicos e industriais com interface pouco amigável para usuários inexperientes. ii) Ptolemy não oferece nenhum tipo de ajuda sobre como o aluno deverá montar seus ambientes. iii) Caso um professor configure um ambiente para que seja trabalhado por seus alunos, ele não terá controle sobre o que os alunos fizeram com o ambiente ou com os atores.

De forma a se obter um bom retorno do potencial da ferramenta e procurar sanar tais problemas, foram verificadas para o projeto as seguintes necessidades: (a) conhecimento sobre o sistema, (b) Java e XML e (c) domínios e atores;

Com relação ao item *a*, sabe-se que a operação correta de qualquer software computacional, passa pelo conhecimento de sua documentação (manuais, ajudas, descrição de funcionalidades, etc.). Entretanto, em virtude do público alvo do Ptolemy, as funcionalidades são muito variadas e em constante ampliação, tornando essa necessidade ainda maior.

Tratando-se do item *b*, o conhecimento de Java e XML serão primordiais para a modelagem de novas tecnologias ou sistemas, ou ainda para aplicação da ferramenta em situações específicas, como no ensino de arquitetura de computadores, pois haverá a necessidade de criação de novos atores.

E por fim, para efetuar simulações e modelagens, utilizando-se da grande biblioteca de atores já criados, precisa-se compreender os domínios, que implementam os modelos computacionais, definindo a semântica das relações entre os atores, e compreender propriamente os atores, para utilização correta de suas portas e parâmetros.

De forma a tratar tais questões, buscou-se através de uma análise de usabilidade no framework Ptolemy, buscar compreender e apontar as necessidades do software para torná-lo uma ferramenta de ensino eficaz, para auxílio na disciplina de Arquitetura de Computadores, de forma mais simples e mais amigável.

Na realização da avaliação de usabilidade, foi utilizada a técnica diagnóstica de inspeção via *checklist*, com o auxílio da ferramenta *Ergolist*<sup>1</sup>, desenvolvida pelo Laboratório de Utilizabilidade da UFSC/SENAI (LabiUtil), coordenado pelo Prof. Dr. Walter de Abreu Cybis. Essa ferramenta possui dezoito *checklists*, totalizando 193 questões.

---

<sup>1</sup> Para acessar o Ergolist, visite:  
<http://www.labiutil.inf.ufsc.br/ergolist/check.htm>

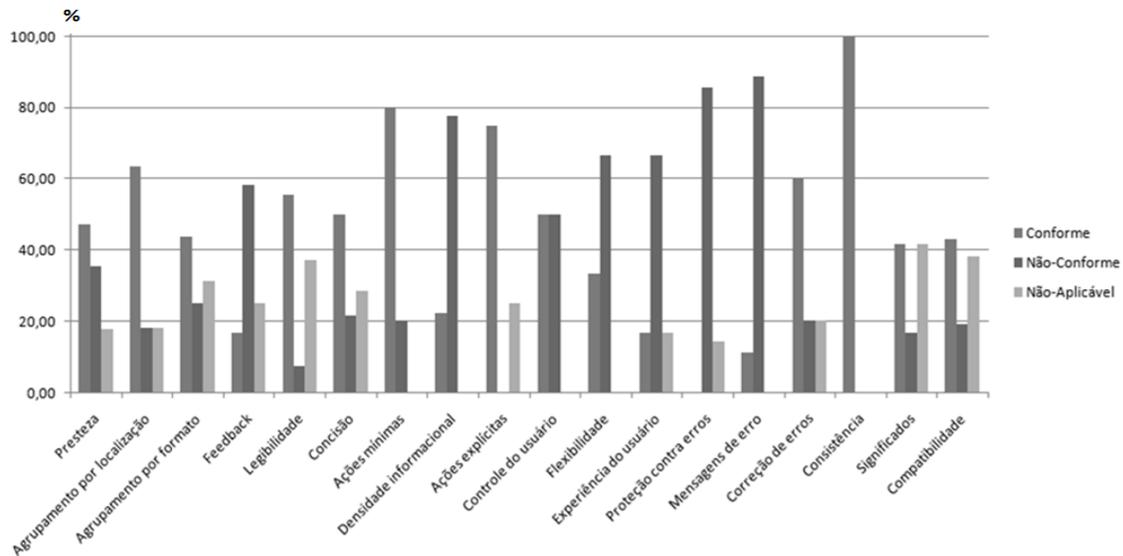


Figura 1: Avaliação da usabilidade do Ptolemy

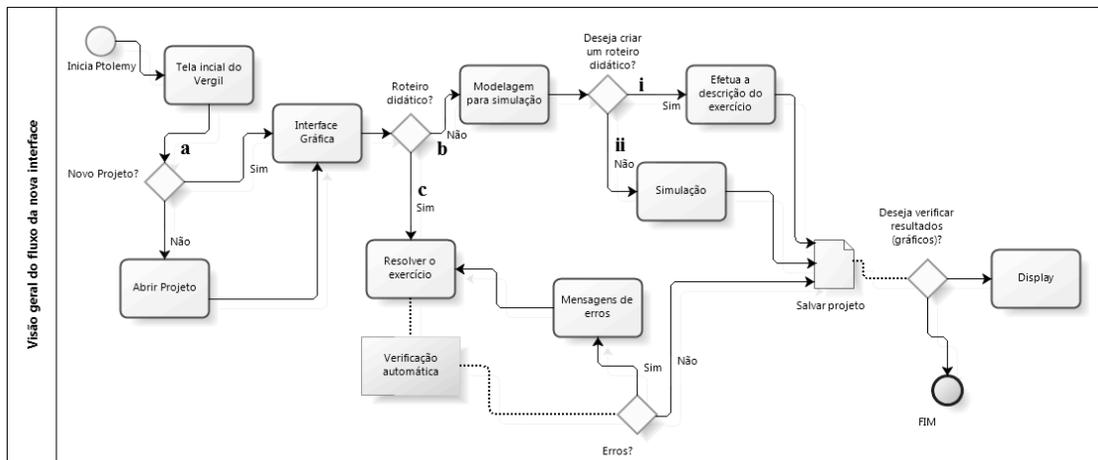


Figura 2: Visão geral da estrutura da nova interface gráfica

Na Figura 1 são apresentadas às estatísticas da avaliação de usabilidade do Ptolemy II na ferramenta Ergolist. O eixo horizontal apresenta os critérios de avaliação e o eixo vertical apresenta o percentual de questões conformes, não-conformes e não-aplicáveis em cada critério.

Após a avaliação, podem-se destacar problemas de usabilidade nos seguintes critérios, cujo percentual de questões não-conformes se sobressaíram:

- *Feedback*: o Ptolemy II apresentou quase 60% de questões não-conformes, tendo em vista que, por existir uma documentação muito extensa, muitas tarefas executadas não fornecem um *feedback*

apropriado para o usuário ficar mais seguro do que está fazendo;

- *Flexibilidade*: o sistema deve disponibilizar diversas formas de se realizar uma tarefa e possibilitar efetuar modificações. Com quase 67% de questões não-conformes, no Ptolemy II só é possível essa flexibilidade se o usuário conhecer Java e XML;
- *Proteção contra erros*: esse critério diz respeito aos mecanismos de detecção e prevenção de erros de entrada de dados ou comandos e ações não recuperáveis. O Ptolemy II obteve mais de 85% de questões não-conformes, pois a maioria dos

erros é detectada na validação dos arquivos XML e classes Java;

Buscando atender às necessidades do software para torná-lo uma ferramenta apropriada para o ensino de Organização e Arquitetura de Computadores, o trabalho descreve duas melhorias: i) tornar sua interface mais amigável, com instruções em português, e para que instruções passo a passo sejam dadas aos alunos na montagem de seus ambientes na forma de atividades; ii) Verificar automaticamente os exercícios feitos pelos alunos e dar *feedback* sobre o que pode ser melhorado.

#### 4. Extensão Ptolemy para fins educacionais

A extensão desenvolvida permite que o usuário modele suas próprias arquiteturas utilizando o repositório de atores específicos para a disciplina de Arquitetura de Computadores.

A nova interface contribuirá para que o professor demonstre na prática aos seus alunos tópicos abordados na ementa da disciplina, como: hierarquia de memória, registradores, barramentos, dispositivos de entrada/saída e a abordagem estrutural no estudo de computadores. De forma a especificar a nova arquitetura do Ptolemy para fins educacionais, foram utilizados padrões de projeto de interface [5], estudo de algumas ferramentas educacionais (seção 2) e abordagens para software educativo.

Inicialmente foram desenvolvidos atores para representação dos componentes da arquitetura do processador MIPS como estudo de caso. Com a aplicação da ferramenta almeja-se estender sua aplicação para outras arquiteturas de processadores, memórias, dispositivos de Entrada e Saída etc.

##### 4.1. Interface gráfica Vergil

A nova interface (Figura 2) foi implementada na versão 8.0.1 do Ptolemy. Seu desenvolvimento foi na linguagem de programação Java, utilizando o compilador *Eclipse Helios Service Release 2*.

Conforme [9], ao acessar a interface gráfica [6], o usuário decidirá se deseja abrir um ambiente de modelagem para: (a) criar um novo projeto ou abrir um projeto já desenvolvido, podendo este ser (b) um projeto salvo no computador ou (c) um roteiro didático também salvo.

No caso (b) o usuário utilizará a interface para modelagem e simulação, onde terá duas opções: i) criar um roteiro didático (Figura 2), utilizando os atores da biblioteca, selecionando os obrigatórios, suas propriedades e as conexões entre eles, além de salvá-lo

e/ou simulá-lo ou ii) criar seu próprio modelo utilizando a biblioteca, salvá-lo e/ou simulá-lo.

Em relação ao item (c) o usuário poderá abrir os roteiros didáticos existentes, resolvendo os exercícios propostos, verifique possíveis erros através do *Checker* e receba um *checklist* da avaliação.

#### 5. Roteiro didático

Para ilustrar o roteiro didático aplicado aos alunos, nesta seção é detalhado um experimento onde foram abordados os conhecimentos descritos no tutorial desenvolvido na disciplina Arquitetura de Computadores no Mestrado em Informática do DI – UFPB, intitulado: “Modelagem e Simulação do Processador MIPS na Ferramenta Ptolemy”. O material serve de base para os conceitos básicos de utilização do framework e definição dos componentes estudados.

O objetivo deste experimento foi o de estimular os alunos a criarem seus próprios modelos de arquitetura de maneira simples, interativa e didática, baseados num roteiro didático exposto pelo professor, ao mesmo tempo, possibilitando que eles verifiquem de uma forma automática os erros no seu cenário.

Na aplicação do experimento, os alunos deveriam criar um modelo que simulasse um dos estágios do processador MIPS, que foi a Busca da Instrução - *Instruction Fetch* (IF) que deveria ter a seguinte premissa: que o modelo buscasse a próxima instrução na memória utilizando o endereço contido no registrador PC (contador de programa) e armazenasse na Memória de Instruções. A Figura 3 mostra a parte do caminho de dados usado na busca de uma instrução e no incremento do PC.

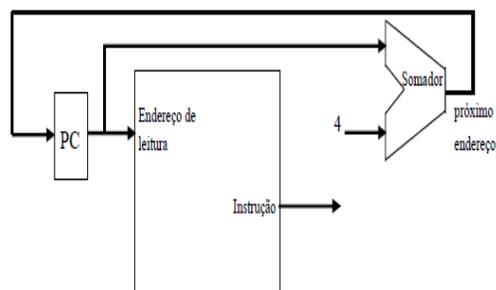


Figura 3: Envio da Instrução do PC para memória

Os alunos seguiram as orientações do roteiro didático e prepararam o modelo com os atores informados pelo professor (Figura 5).

Um ponto importante do exercício foi o aprendizado do funcionamento e da importância da temporização de um sistema.

Ainda sobre o exercício, podemos verificar que o ator *InstructionMemoryAndDecorder* consiste em receber na sua porta de entrada o valor de 32 bits originados do *PC* e usá-lo com index para selecionar uma instrução na memória de instrução, enviando para as suas saídas segmentos da instrução de entrada, de forma que cada segmento seguirá para um destino diferente, seguindo a ordem abaixo:

- Instrução 25 -0: Esses 26 bits seguem para o ator *Shift Left2* que tem o papel de efetuar o deslocamento dos elementos em dois bits à esquerda.
- Instrução 31 -26: A Unidade de Controle receberá 6 bits da instrução;
- Instrução 25 – 21: Esses 5 bits da instrução irão direto para o banco de registradores;
- Instrução 20 – 16: Esses 5 bits irão para o Multiplexador, como também para o banco de registradores;
- Instrução 15 -11: Esses 5 bits irão para o Multiplexador, antes de chegar no banco de registradores;
- Instrução 15 – 0: Esses 16 bits seguirão para a Extensão de Sinal, sendo que 6 bits desses irão para a *ALU Control*.

Um sinal de clock foi adicionado (ator mais a esquerda na Figura 4), e atores especiais (Time Delay) foram utilizados para gerar um atraso inerente ao processo entre enviar a instrução e esperar seu resultado ficar pronto na saída.

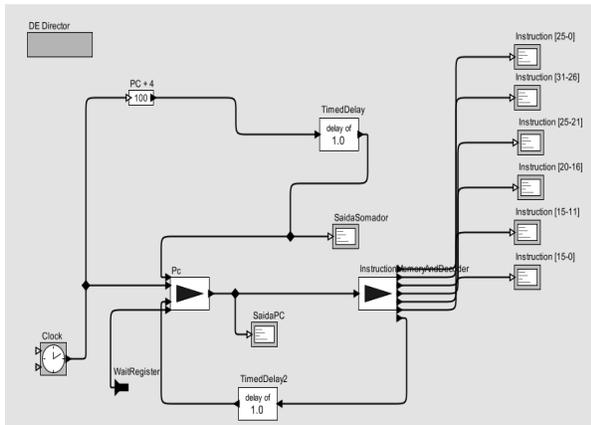


Figura 4: Modelo criado em sala de aula

Na resolução do exercício, são informadas as três instruções de 32 bits que correspondem ao index originado do *PC* (Figura 5). Logo, o aluno ao receber o feedback das tarefas que faltavam ser realizadas pelos mesmos e não encontrar mais erros no seu cenário,

poderá analisar que na saída do ator *InstructionMemoryAndDecorder* irão aparecer os segmentos de instruções, que seguirão para os outros componentes.

## 6. Experimentos e resultados

A inclusão do experimento e suas atividades práticas na disciplina de Arquitetura de Computadores vem suprir uma necessidade dos alunos em fixar melhor o aprendizado, simulando situações reais em que elas acontecem. Além disso, estas atividades práticas podem servir como motivadores iniciais dos tópicos a serem estudados dando um estímulo aos alunos que os levará a entender melhor os assuntos.

Os alunos que participaram do experimento estão no 4º Período do Curso de Ciência da Computação, dos turnos vespertino e noturno. Pela análise da ementa conclui-se que a disciplina de arquitetura e organização de computadores tem o foco para os aspectos básicos, tanto de baixo quanto de alto nível, relacionados aos recursos computacionais.

Durante o experimento ficou claro que os mesmos possuem uma grande dificuldade em relação a tais aspectos, como também, as características dos componentes e do caminho de dados entre eles.

Apenas a extensão do Ptolemy foi avaliada para verificar sua aplicação e usabilidade para fins educacionais. Os alunos preencheram um formulário online (<http://twixar.com/YEWuLspLAT6h>) onde puderam informar quais foram às dificuldades encontradas na resolução do experimento.

Como se tratava de turmas pequenas, apenas 16 alunos participaram das aulas e responderam à avaliação aplicada. A seguir, são apresentados alguns destes resultados.

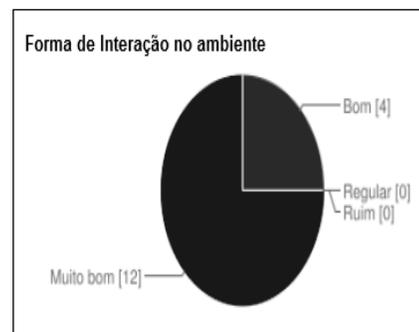


Figura 6: Forma de interação no ambiente

### 6.1. Forma de interação no ambiente

Na Figura 6 é informado o resultado da questão respondida pelos alunos referente a forma de interação no ambiente.



Figura 5: Resultado do experimento

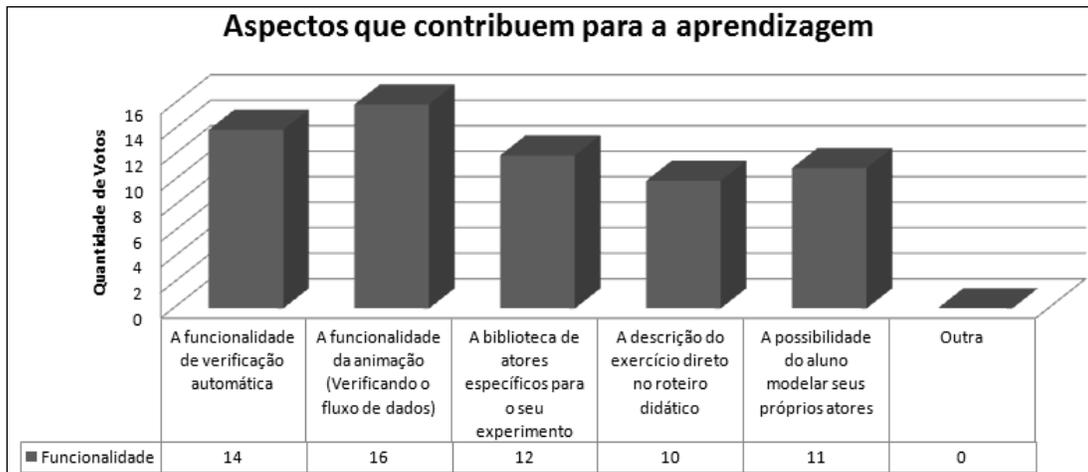


Figura 7: Características da extensão Ptolemy

Na questão, 12 alunos consideraram como “muito bom” os aspectos de simulação e de acompanhamento do fluxo de dados.

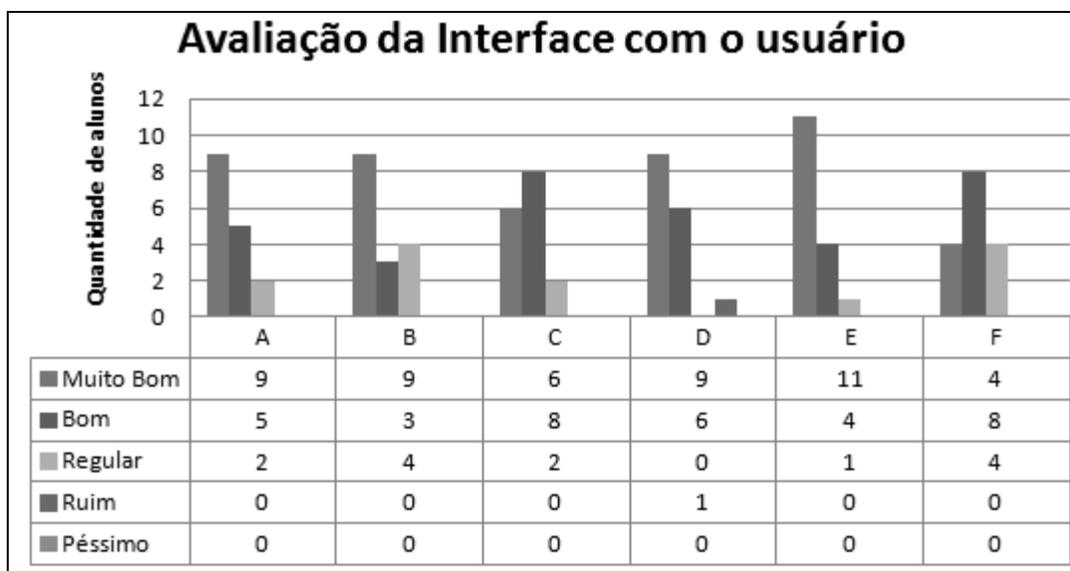
## 6.2. Aspectos que contribuem para a aprendizagem

Na questão sobre os aspetos que contribuem para a aprendizagem, o aluno deveria assinalar as características da extensão que mais contribuíram para a aprendizagem segundo sua interpretação. Na Figura 7 duas características se destacaram: a funcionalidade de animação (94%) e a funcionalidade de verificação automática (88%). A característica de animação possibilita aos alunos acompanharem o fluxo de dados

e das instruções envolvidas no experimento passo-a-passo, facilitando o entendimento pela passagem das instruções pelos atores. Já a verificação automática, possibilita o feedback ao aluno.

## 6.3. Avaliação da Interface com o usuário

A Figura 8 ilustra os resultados para as questões que tratavam da avaliação geral da interface com o usuário. Nestas questões foram avaliados alguns itens. São eles: (A) Disponibilidade das tarefas; (B) Interface do programa; (C) Apresentação dos menus; (D) Tamanho e tipo de fontes utilizadas; (E) Forma de despertar o interesse do aluno; (F) Navegação pelo sistema.



**Figura 8: Avaliação da interface com o usuário**

Em relação aos itens (A) e (B), mais de 56% da turma os consideraram como “muito bom”. Nestas questões, os alunos foram questionados pela forma que as tarefas ficavam disponíveis na área de trabalho da Extensão do Ptolemy e como os mesmos analisaram inicialmente as mudanças feitas na interface do programa.

Em relação ao item (C), 88% dos alunos consideram como “muito bom” ou “bom” a localização dos menus, atalhos e ícones.

Para o item (D), o percentual de 56% ficou satisfeito com o tamanho e tipo de fontes utilizadas, classificando como “muito bom”.

O item (E) teve o maior percentual de aprovação pelos alunos, com 69%. Nesta questão, os alunos avaliaram como “muito bom” a forma que a Extensão do Ptolemy possibilita despertar interesse. O fato da criação de atores específicos para a disciplina possibilita a eles entender melhor seu funcionamento, suas características e como cada ator se relaciona com outros no cenário.

Outra característica que colabora para um alto conceito da questão é a verificação automática, onde é possível passar para o professor como o aluno está realizando (ou realizou) as atividades escritas pelo mesmo, como também colaborar para o aluno resolva o experimento sem a presença do professor. Finalizando as questões referentes à avaliação da interface com o usuário, o item (F) teve 75%, quando se considera as respostas como: “muito bom” ou “bom”.

O formulário de avaliação aplicado com os alunos aborda tópicos levantados anteriormente para avaliar a usabilidade do Ptolemy. Logo, podemos concluir que

os critérios que apresentaram não conformidade no Ptolemy obtiveram um bom percentual de conformidade com a extensão criada para fins educacionais.

Para as próximas avaliações, será verificado se a ferramenta atende os requisitos para ser caracterizada como um software educacional, como também será realizado um novo experimento no qual os alunos serão avaliados.

Como requisito de um software educacional serão levadas em consideração, algumas características pedagógicas, como: viabilidade de utilização do software em situações educacionais, adaptabilidade a metodologia de ensino e adequação a uma proposta de educação mais construtivista.

Para avaliar os alunos, serão abordadas atividades com a concepção do propósito de sua utilização. Onde se procura dar exemplos e fazer experiências com o software em situações educacionais.

## 7. Conclusão

Como requisito de um software educacional, a extensão do Ptolemy possui agora algumas características pedagógicas, tais como: viabilidade de utilização do software em situações educacionais, adaptabilidade a metodologia de ensino e adequação a uma proposta de educação mais construtivista.

Para avaliar a eficácia da ferramenta junto aos alunos, foi realizado um experimento prático que demonstrou uma boa avaliação por parte dos mesmos.

Como trabalhos futuros novos componentes serão desenvolvidos para o ensino de outros aspectos da Arquitetura de Computadores, tais como os barramentos, entrada e saída e memória, assim como novos experimentos envolvendo diferentes alunos.

Outro aspecto importante que pode ser avaliado é a possibilidade de aplicação da extensão do Ptolemy para o ensino de outras disciplinas. Teoricamente, qualquer sistema que possa ser baseado em atores, pode ser ensinado através do Ptolemy modificado. Como disciplinas em potencial que podem ser exploradas são Redes de Computadores, Sistemas Operacionais e Algoritmos, isso sem falar de disciplinas de outras áreas, como Processamento de Sinais, Cálculo, Física, entre outras.

## Referências

- [1] Branovic, I, Giorgi, R e Matinelli, E., 2004. WebMIPS: A New Web-Based MIPS Simulation Environment for Computer Architecture Education, *Proceedings of the 31<sup>st</sup> Annual International Symposium on Computer Architecture*, Munchen, Germany.
- [2] Brito, A. V. Simulação Baseada em Atores para o Ensino de Arquitetura de Computadores. Apresentado no Workshop sobre Educação em Arquitetura de Computadores (WEAC) – Simpósio Brasileiro de Arquitetura de Computadores. SBAC, São Paulo, 2009.
- [3] Brorsson, M. MipsIt: a simulation and development environment using animation for computer architecture education. In *Proceedings of 2002 Workshop on Computer Architecture Education: Held in Conjunction with the 29<sup>th</sup> international Symposium on Computer Architecture (Anchorage, ACM, New York, NY, 12. p. WCAE'02.Alaska)*, p. 1 – 8.
- [4] Felix, A. F; Pousa, C. V. e Carvalho, M. B. DIMIPSS: Um simulador didático e interativo do MIPS . Apresentado no Workshop sobre Educação em Arquitetura de Computadores (WEAC) – Simpósio Brasileiro de Arquitetura de Computadores. SBAC, Ouro Preto - MG, 2006.
- [5] Koscianski, A., Soares, M. S. Qualidade de Software: Aprenda as metodologias e técnicas modernas para o desenvolvimento de software. Editora: Novatec. São Paulo – SP. 2007.
- [6] Lee, A, 2007. Neuendorffer, S. Tutorial: Building Ptolemy II Models Graphically. Technical Report No. UCB/EECS-2007-129.
- [7] Nikolic, B., Radivojevic, Z., Djordjevic, J and Milutinovic V, 2009. A Survey and Evaluation of Simulators Suitable for Teaching Courses in Computer Architecture and Organization. *IEEE Transactions on Education*, Vol. 52, No. 4.
- [8] Scott, M. WinMips64, version 1.5, School of Computing, Dublin City University, Ireland, 2006.
- [9] Torres, A. L. L, Brito, A. V. Extensão do Ptolemy para o ensino de Organização e Arquitetura de Computadores. Workshop sobre Educação em Arquitetura de Computadores (WEAC) - Simpósio Brasileiro de Arquitetura de Computadores. SBAC, Vitória - ES, 2011.
- [10] Vollmar, K. and Sanderson, P., A MIPS Assembly Language Simulator Designed For Education. *The Journal of Computing Sciences in Colleges*, Vol. 21, No. 1, 2005.