

- A Figura 3.12 elucida o mapa conceitual do experimento de Lançamento Oblíquo.

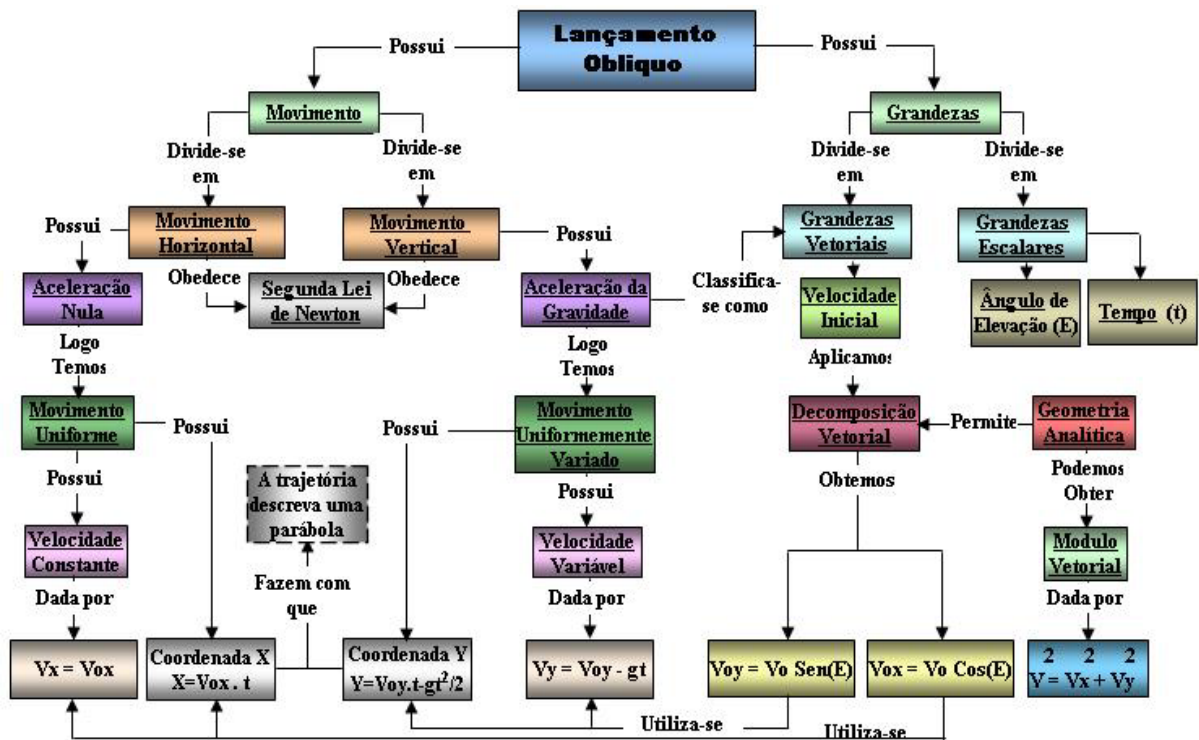


Figura 3.12: Mapa conceitual do experimento de Lançamento Oblíquo

- A Figura 3.13 apresenta o mapa conceitual do experimento de Empuxo da Hidrostática.

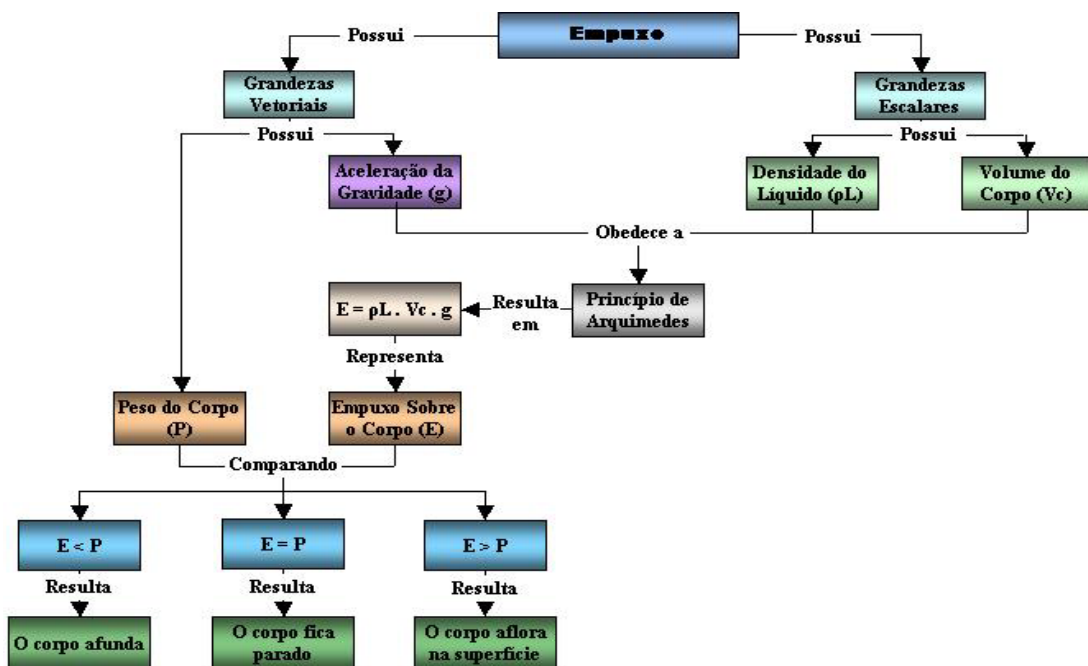


Figura 3.13: Mapa conceitual do experimento de Empuxo da Hidrostática

3.5 Considerações Finais

Neste capítulo foram analisadas algumas teorias construtivistas que podem auxiliar no aprendizado dos alunos. Assim como, ressaltados pontos importantes destas teorias que contribuem para um melhor desempenho de *softwares* educacionais e, apresentado os desenhos pedagógicos desenvolvidos para o SEFIRV.

No próximo capítulo apresenta-se a arquitetura do SEFIRV, levando-se em consideração os *softwares* analisados no Capítulo 2 e os processos pedagógicos estudados neste capítulo.

Capítulo 4

Arquitetura do Sistema

4.1 Introdução

A partir do estudo descrito nos capítulos anteriores, elaborou-se uma arquitetura de um sistema voltado para o ensino que permita ao usuário estabelecer uma conexão conceitual com o mundo real e realizar uma aprendizagem significativa dos conceitos físicos da grade curricular do Ensino Médio.

4.2 Tecnologias de Apoio

Antes de apresentar a referida arquitetura do sistema em detalhes, será realizada uma breve exposição sobre as tecnologias de apoio necessárias para o funcionamento deste sistema, bem como a importância da utilização de cada uma delas.

4.2.1 VRML e X3D

Existem várias abordagens sobre a Realidade Virtual, sendo que algumas requerem o uso de tecnologias avançadas como plataformas de *hardware* e *software* sofisticadas. No entanto, a linguagem VRML possibilita a criação de mundos virtuais tridimensionais, com alta qualidade, aplicáveis à Internet, utilizando apenas um *browser* e um *plug-in* para realizar a visualização. O *plug-in* é um programa que permite visualizar informações dentro da janela do navegador de Internet. O surgimento dos *plug-ins* ocorreu para evitar as constantes modificações nos navegadores para se adaptarem aos novos formatos de mídia. Optou-se por manter os navegadores e desenvolver programas específicos, os *plug-ins*, associados a cada tipo de arquivos que, de acordo com a necessidade do usuário, seriam utilizados pelo navegador.

A lista de *plug-ins* para visualização 3D cresceu rapidamente, existindo atualmente uma grande variedade à venda no mercado. Entre os *plug-ins* mais utilizados estão o Cortona criado pela ParallelGraphics [ParallelGraphics, 2005], e o Cosmo Player produzido pela Silicon Graphics [Silicon Graphics, 2005], ambos estão disponíveis, gratuitamente, na Internet para *download*. Neste trabalho optou-se por utilizar o Cortona devido a sua alta capacidade de renderização, oferecendo uma visualização dos objetos virtuais com maior qualidade e a sua maior portabilidade, sendo compatível a diversos navegadores. A seguir, são apresentadas as ferramentas utilizadas na arquitetura deste sistema.

A sigla VRML é a abreviação de Virtual Reality Modeling Language, cuja tradução para Português é Linguagem para Modelagem de Realidade Virtual. VRML é uma linguagem criada por Mark Pesce e Tony Parisi, que independe de plataforma e permite a criação de cenários 3D. A linguagem foi concebida para descrever simulações interativas, em mundos virtuais disponibilizados na Internet e ligados com o WWW (World Wide Web), mas a primeira versão da linguagem, versão 1.0, não possibilitou grandes interações do usuário com o ambiente virtual. Nas versões atuais foram acrescentadas características como animação, movimentos de corpos, som e etc. A última versão é a 2.0 draft #3, chamada Moving Worlds VRML 2.0. [Hartman and Wernecke, 1996].

Tudo que se precisa para escrever um código VRML é um editor de textos. Uma vez editados, os arquivos são gravados em formato ASCII (American Standard Code for Information Interchange) com a extensão *.wrl*, sendo necessário um *browser* e um *plug-in* para visualizar os objetos virtuais criados, como já dito.

A versão de VRML 1.0 tem como principais características a criação de mundos 3D estáticos e a criação de âncoras para outros ambientes. Já na versão de VRML 2.0, os objetos do mundo virtual podem mover-se e responder a eventos baseados no tempo ou em iniciativas do usuário. Além disso, esta nova versão permite a utilização de objetos multimídia, como sons e filmes em uma cena 3D. Em resumo, as características do VRML 2.0 podem ser agrupadas em quatro áreas principais:

melhoria dos mundos estáticos, interação, animação e comportamento baseado em scripts, e prototipação de novos objetos VRML. [Fox and Shaddock, 1996]

A estrutura de um arquivo VRML 2.0 pode conter quatro tipos principais de componentes:

- Cabeçalho (obrigatório em qualquer arquivo VRML);
- *Prototypes*;
- Formas, *Interpolators*, Sensores, *Scripts*;
- Rotas;

A modelagem dos Ambientes Virtuais é feita basicamente através de nós, que descrevem os objetos da cena e suas propriedades, utilizando parâmetros chamados de *fields*. Os *nodes* podem ser hierarquicamente agrupados de modo a viabilizar a representação audio-visual dos objetos, além de participarem do processo de geração e transmissão de eventos.

Abaixo é apresentado um pequeno trecho de código VRML, que exemplifica alguns de seus componentes e a sua simplicidade em termos de codificação desta linguagem. A Figura 4.1 mostra como ocorre a visualização deste código no *browser*.

```
#VRML V2.0 utf8           #Cabeçalho
Shape {
  geometry
    Sphere {               #Nodo
      radius 2.0           #Campo
    }
}
```

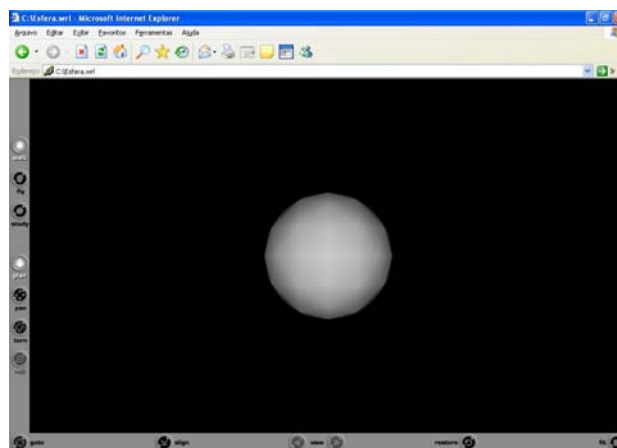


Figura 4.1: Exemplo de um arquivo VRML

A modelagem de um Ambiente Virtual em VRML pode ser feita apenas em um editor de texto padrão, porém isto se torna extremamente difícil quando os objetos virtuais possuem geometrias complexas. Para amenizar tal problema existem diversas ferramentas para auxiliar a modelagem tridimensional como, por exemplo, o 3D Studio [Discreet, 2002], o Cosmo World [Cosmo worlds, 1996], etc. Contudo, ao término da modelagem faz-se necessária a sua conversão para o formato VRML, o que pode acarretar alguns problemas, como a geração de um código VRML excessivamente grande e até mesmo a geração de um código com erros em sua estrutura.

Atualmente surge um novo padrão para criação de ambientes tridimensionais via Web o X3D, uma extensão do XML - Extensible Markup Language. Ele não é uma API (Application Programming Interface) de programação, nem um formato de arquivo para troca de geometrias. O X3D combina ambos, geometria e descrições de comportamentos instantâneos em apenas um arquivo. [X3D, 2005]

4.2.2 JavaScript

A evolução do potencial da linguagem VRML em termos de interação e animação de objetos virtuais deve-se, em grande parte, à criação do nó Script. Ele provê meios de incluir formas mais complexas de interação e pode ser usado para realizar uma grande variedade de tarefas, mas é tipicamente utilizado para desempenhar funções como:

- agir como o estágio da "máquina de execução" para controlar uma animação;
- processar entradas para agir como o estado de Lógica em uma animação.

Como outros nós, um Script recebe eventos de entrada e gera eventos de saída. Ao contrário de outros nós, entretanto, ele pode executar algum tipo de processamento antes de produzir esses eventos de saída. A parte do nó que realiza o processamento é um programa escrito em uma linguagem de programação suportada pelo *browser*, e as linguagens geralmente utilizadas são Java e JavaScript.

JavaScript é uma linguagem desenvolvida especificamente para a programação de scripts diferentemente do Java. O JavaScript não oferece todos os recursos computacionais de Java e, principalmente, não inclui nenhum suporte às funções de acesso às redes. Por esta razão, para criar mundos virtuais que se baseiam em uma arquitetura cliente/servidor (como em um ambiente multi-usuário, por exemplo) os desenvolvedores geralmente terão de optar por Java.

A combinação VRML/Java através dos nós Script é bastante poderosa, pois o programa associado ao nó pode controlar toda a interação e o comportamento dos elementos do mundo virtual. [Tamiosso and Raposo, 1997].

Cada vez que um nó Script recebe um conjunto de eventos de entrada, o browser entrega-os para o programa especificado. Através de vários métodos (dependendo da linguagem utilizada para escrever o programa), cada evento é tratado. A abordagem mais comum para o tratamento de eventos de entrada é a utilizada em JavaScript: definir uma função no programa para cada evento de entrada listado no nó Script. Quando um evento de entrada chega ao nó, o *browser* chama a função de mesmo nome. A função recebe como parâmetros o valor do evento e um timestamp que indica, geralmente, quando o evento foi enviado.

Para cada campo e evento de saída definido em um nó Script, o *browser* cria uma variável global com o mesmo nome e o mesmo tipo, para ser usada pelo programa. Para inserir o valor de um evento de saída, basta atribuir um valor à variável correspondente.

Abaixo se encontra um trecho de código VRML, contendo um JavaScript que atribui uma forma espiral a um objeto virtual de acordo com equações matemáticas inseridas no código de programação JavaScript. A Figura 4.2 ilustra este objeto.

```
url "javascript: // Início do código de programação JavaScript
function generateSpine( ){
    y = -6; angle = 0.0; delta = 0.19625;
    for( i = 0; i <= (192); i++){
        spine_changed[i][0] = 2 * Math.cos(angle);
        spine_changed[i][1] = y;
        spine_changed[i][2] = 2 * Math.sin(angle);
        angle += -delta; y = y + 0.04;
    } } "
```

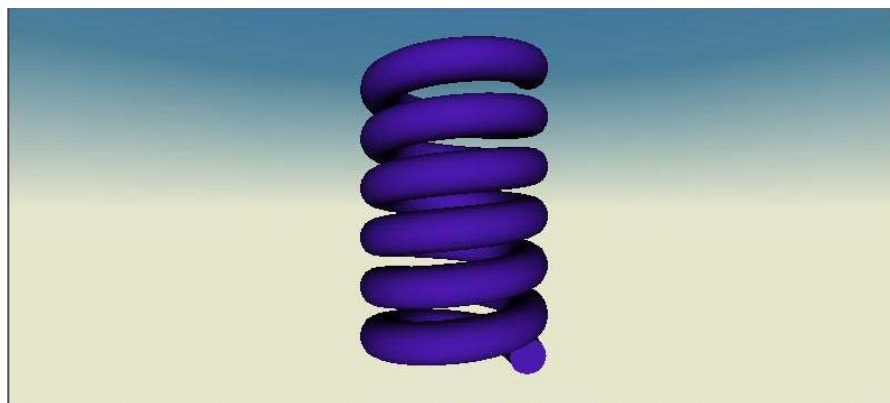


Figura 4.2: Objeto virtual com o código JavaScript

4.3 Diagrama da Arquitetura do Sistema

O sistema proposto possui uma interface simples, permitindo a professores e alunos a manipulação fácil dos ambientes virtuais. Não há necessidade de conhecimentos técnicos relacionados à Informática e/ou sobre os conceitos de Física, porque foram criadas estratégias de navegação que orientam os aprendizes no processo de interação nos ambientes de aprendizagem.

A primeira estratégia é a criação de um conjunto de organizadores gráficos conceituais da grade curricular do ensino de Física, para facilitar o processo de navegação conceitual e estimular a aprendizagem significativa. A segunda estratégia é criar o acesso às experiências virtuais de fenômenos físicos por meio dos nós conceituais dos organizadores gráficos. A terceira estratégia é o acesso aos mapas conceituais tutoriais nos ambientes de simulações de experiências virtuais.

Os Ambientes Virtuais do sistema são construídos com a linguagem VRML e JavaScript, o que contribui para a criação de arquivos pequenos, que são compartilhados via Web a um baixo custo de transmissão e armazenamento. O acesso a Web é obtido por meio de um programa navegador na Internet. A visualização dos Ambientes Virtuais é realizada por meio de um *plug-in*, como o Cortona, que é executado sob supervisão do navegador da Internet.

Os Ambientes Virtuais dos experimentos são, em grande maioria, modelados de acordo com o mundo real, facilitando deste modo uma melhor associação do conteúdo com o experimento. Para a montagem e definição dos Ambientes Virtuais (cenários de

aprendizagem) foram elaborados conjuntos de objetos de aprendizagem, cada qual tentando representar o conceito ao qual a experiência se refere.

O usuário pode acessar a GUI (Interface Gráfica do Usuário) do sistema que oferece suporte a todos processos de criação e visualização dos resultados dos experimentos, presencialmente ou a distância, via Internet. A Figura 4.3 mostra o diagrama básico da arquitetura do sistema. Tal figura representa a visão geral da comunicação e as partes que compõem este sistema.

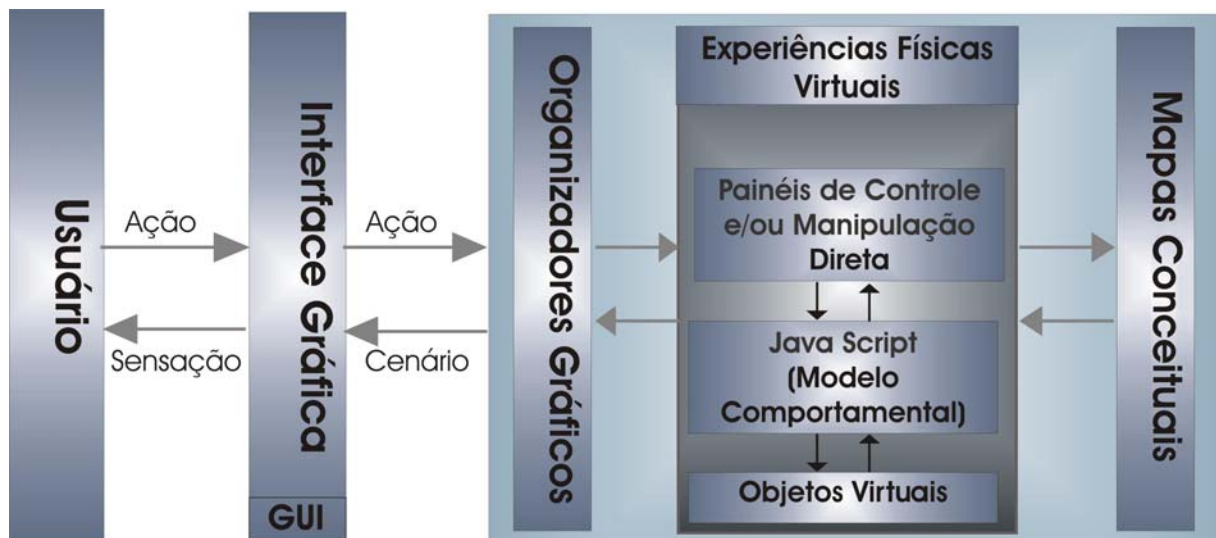


Figura 4.3: Arquitetura do Sistema

A arquitetura proposta é formada pelos módulos e sub-módulos a seguir:

- Interface Gráfica do Usuário;
- Bloco dos Organizadores Gráficos;
- Bloco das Experiências Físicas Virtuais:
 - ✓ Bloco dos Painéis de Controle e/ou Manipulação Direta;
 - ✓ Bloco Modelo Comportamental;
 - ✓ Bloco dos Objetos Virtuais;
- Bloco dos Mapas Conceituais.

4.3.1 Interface Gráfica com o Usuário – GUI

Este bloco permite ao usuário visualizar graficamente e de modo interativo o resultado de suas ações no sistema. O usuário realiza uma ação, entrada de dados, por exemplo, e recebe da GUI a saída de informações, através de atualizações, proporcionando uma maior sensação de interação com o meio virtual.

A GUI permite o acesso aos organizadores gráficos que ativam os módulos das experiências que, por sua vez, possibilitam o acesso ao conteúdo dos mapas conceituais. Assim, a GUI pode apresentar:

- um organizador gráfico, modelado a partir de primitivas geométricas, que permite a visualização de conceitos, suas definições, o acesso a outros organizadores e a simulações virtuais;
- uma experiência de Física a ser desenvolvida, composta por um conjunto de objetos virtuais de aprendizagem que viabilizam interações e a reações do ambiente virtual;
- o mapa conceitual da experiência, desenvolvido em 2D para evitar confusões visuais.

4.3.2 Bloco dos Organizadores Gráficos

Este bloco corresponde a um conjunto de organizadores gráficos conceituais construídos em três dimensões, usando a linguagem VRML. Os organizadores gráficos conceituais foram desenhados em estruturas de árvores. Este modelo teve como objetivo eliminar a organização linear e unidimensional dos conteúdos curriculares formados pelos modelos tradicionais de ensino, e criar um sistema multidimensional e interdisciplinar para a organização e distribuição dos conceitos científicos. Foram utilizados organizadores gráficos para grade curricular porque os mapas conceituais podem confundir visualmente os usuários com a sua estrutura de reconciliações integrativas e estruturas semânticas.

O uso de VRML para organização e distribuição dos conceitos teve como objetivo facilitar a visualização rápida de profundidade do todo da rede conceitual, fato que não é possível com o uso de sistemas bidimensionais (que necessitam de navegação por meio de sub-mapas para a visualização da rede conceitual). Ao colocar o mouse sobre cada conceito o usuário pode visualizar os conceitos que virão, se a progressão ocorrer. Deste modo, o estudante pode investigar minuciosamente os conceitos e verificar se possui conhecimento *a priori* necessário (Subsunçores) para assimilar a nova informação, como apresentado na Figura 4.4, que demonstra que, se o usuário colocar o mouse sobre o nó conceitual “Mecânica”, ele visualizará os conceitos que virão.

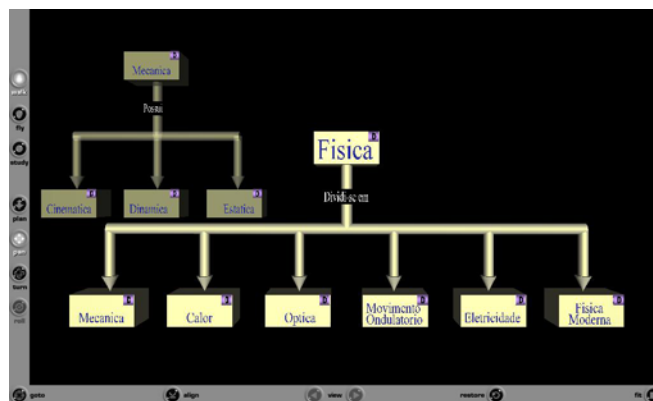


Figura 4.4: A visualização de conceitos anteriores

Estes ambientes também apresentam um ícone que permite ao usuário visualizar e retornar à ramificação conceitual anterior, como exemplifica a Figura 4.5. Permitir a visualização de conceitos anteriores e posteriores em um mesmo ambiente é uma estratégia pedagógica que tem como objetivo localizar conceitualmente o usuário nos organizadores gráficos virtuais.

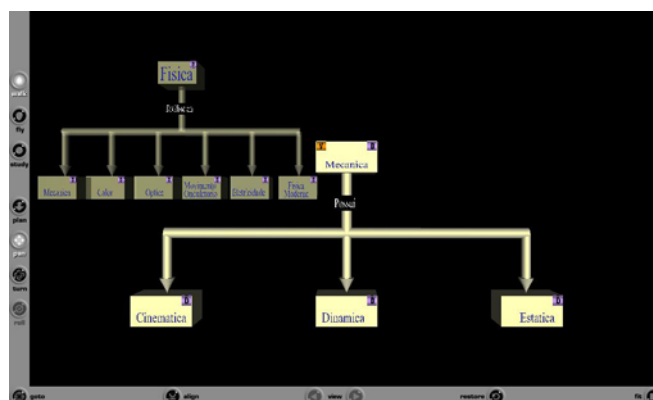


Figura 4.5: A visualização de conceitos posteriores

A progressão dos organizadores gráficos ocorre através de trocas de ambientes virtuais, que apresentam um sistema de redes conceituais com poucos nós na interface, com objetivo de atrair a imagem visual e focalizar a atenção. Isso pode ser verificado na Figura 4.6, quando o usuário acessa a “Mecânica”, ele passa para um novo Ambiente Virtual com conceitos de maior especificidade que os conceitos anteriores.

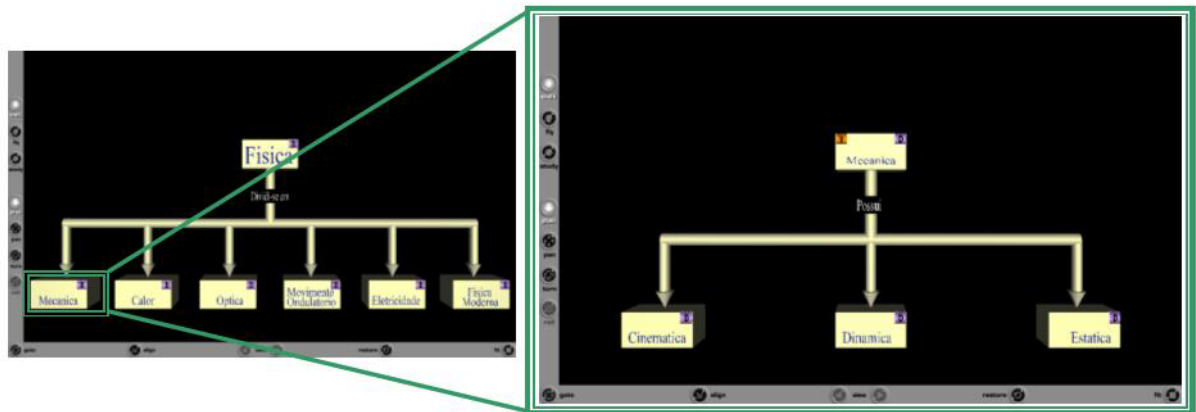


Figura 4.6: Progressão de conceitos

O usuário pode também acessar as definições conceituais de cada nó. Esse procedimento possibilita o gerenciamento da aprendizagem significativa dos conceitos envolvidos nos fenômenos físicos. Na Figura 4.7, por exemplo, pode-se verificar a definição de cinemática e, depois disto, o usuário pode fechá-la, evitando confusão visual no ambiente virtual.

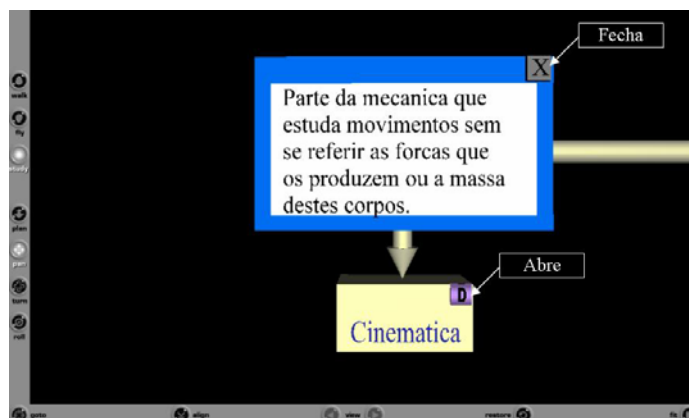


Figura 4.7: Conceitos e definições

Os conceitos mais específicos apresentam ambientes de aprendizagem que possibilitam a interação do aprendiz com simulações de fenômenos físicos em Realidade Virtual, envolvendo toda a rede conceitual de um determinado fenômeno. A

Figura 4.8 mostra que se o usuário quiser requisitar o conceito de Espelhos Esféricos, ele é transferido para um novo Ambiente Virtual que é interativo, permitindo-lhe a observação, exploração e descoberta de fenômenos físicos, podendo também inserir e receber dados.

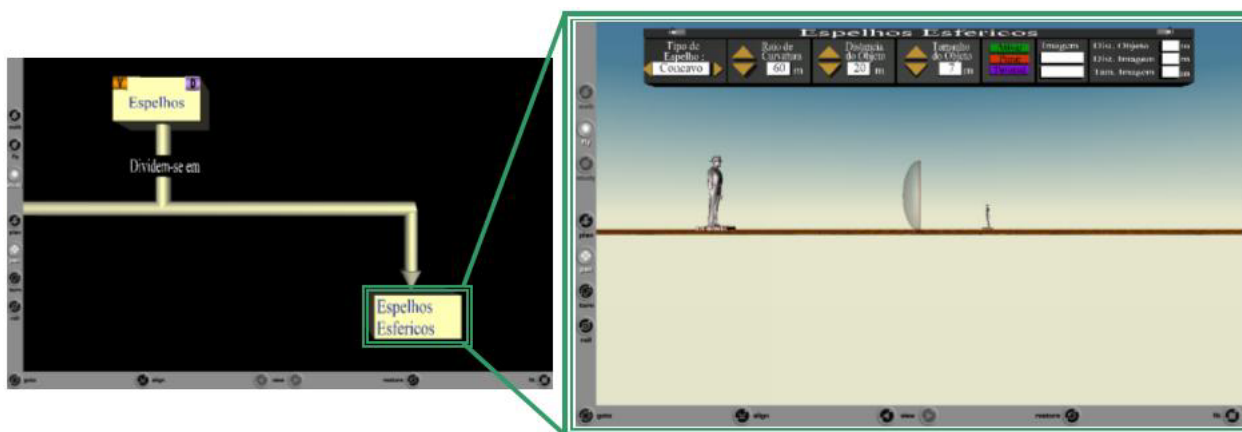


Figura 4.8: Conceitos - Experiência ativada

4.3.3 Bloco das Experiências Físicas Virtuais

Este bloco foi produzido para auxiliar a compreensão de definições e conceitos *a priori*, associando-os a objetos virtuais e seus comportamentos. As experiências físicas foram construídas de acordo com o mundo real do usuário, por exemplo, carros percorrendo uma rodovia ou uma avenida no meio urbano, aviões sobrevoando cidades, circuitos elétricos usados para acender uma lâmpada, etc. Tais situações fazem parte do cotidiano do aprendiz, o que o leva a associar o Ambiente Virtual ao mundo real e ao conteúdo estudado. Esta associação é de grande importância, pois mostra ao usuário qual é a aplicação e o porquê de se estudar Física no Ensino Médio.

4.3.3.1 Experiências de Física Desenvolvidas

Foram construídas vinte e seis experiências físicas virtuais distribuídas em diferentes áreas da Física e organizadas conceitualmente pela ferramenta cognitiva organizador gráfico. Tais experiências são:

- i. O experimento de Movimento Retilíneo Uniforme (MRU), ilustrado na Figura 4.9.

Esta simulação possui um cenário virtual com dois carros de cores diferentes, cor azul e cinza, localizados em uma rodovia sinalizada por placas verticais situada entre montanhas. O usuário pode alterar a velocidade de ambos os carros, a distância entre eles e decidir se o movimento destes carros deve acontecer no mesmo sentido ou em sentidos opostos.

Após a seleção destes dados e ativada a experiência, o usuário pode visualizar a posição de cada carro e o tempo do experimento. Com estes dados o usuário observa a posição de encontro dos carros, quando houver, e o tempo gasto para tal encontro.



Figura 4.9: Experiência do Movimento Retilíneo Uniforme

ii. O experimento de Movimento Retilíneo Uniformemente Variado (MRUV).

A criação do cenário de aprendizagem do MRUV consistiu na construção de uma área urbana virtual que se baseia em detalhes do mundo real, contendo casa, pessoas, animais domésticos, etc. Este Ambiente Virtual possui um conjunto de objetos educacionais necessário para a aprendizagem dos conceitos envolvidos no experimento: seis bairros, cinco semáforos e uma caminhonete situada em uma avenida principal. Cada bairro mede duzentos metros e todos possuem placas verticais de posição, Figura 4.10 (a), que permitem ao usuário calcular este valor. Outra informação, transmitida por placas de sinalização pelo

Ambiente Virtual, é o limite máximo de velocidade, ou seja, vinte e cinco metros por segundo, Figura 4.10 (b).



Figura 4.10: Placas de sinalização de Trânsito.

Os semáforos estão localizados em todos os cruzamentos da avenida principal e possuem três tipos de sinais luminosos diferentes e pré-cronometrados: a luz verde, com duração de vinte e cinco segundos, indica a livre passagem de automóveis, a luz amarela, com duração de cinco segundos, adverte que ocorrerá uma transição para a luz vermelha, e a luz vermelha, com duração de trinta segundos, indica que está proibida a passagem de automóveis. Os cinco semáforos estão sincronizados, com o tempo de dez segundos, para executarem o fenômeno de trânsito conhecido como “Onda Verde”.

O objetivo do usuário, nesta simulação, é passar por todos os semáforos abertos sem ultrapassar o limite de velocidade, podendo acelerar ou frear a caminhonete a qualquer momento do experimento. Caso não cumpra seu objetivo, ele é multado a cada infração.

A primeira ação realizada no Ambiente Virtual da simulação é escolher valores para a aceleração do motor e de frenagem da caminhonete. Esta calibragem do veículo possibilita a interação com o cenário virtual e a recepção de diferentes sensações de acordo com suas escolhas no ambiente. A entrada destes dados é feita por meio de um painel de controle que permite a seleção de valores inteiros entre um e nove para ambas as acelerações.

Além disso, o usuário tem a opção de realizar a experiência de fora da caminhonete por meio de uma câmera externa ou aérea. A forma externa possibilita utilizar um ponto de visão que acompanha a caminhonete durante o experimento, como mostra a Figura 4.11. A forma aérea permite a visualização

do cenário virtual de cima para baixo como um todo; em ambas as opções deve-se utilizar o painel de controle para observar os dados de saída.



Figura 4.11: Câmera externa.

O *layout* deste cenário virtual foi desenvolvido pela equipe da Artes, composta pela aluna de iniciação científica Stephanie Nagamine e pela professora Nikoleta Kerinska da UFU. A equipe contribuiu com a criação de alguns objetos virtuais como casas, prédios, etc. E a geração de algumas imagens como o fundo do painel de controle.

Realizando o experimento de forma aérea, pode-se visualizar um rastro deixado pela caminhonete que muda de cor dependendo das ações do usuário, que podem ser: (1^a) acelerar, sendo a velocidade variável, o rastro deixado será de cor verde; (2^a) acelerar, sendo a velocidade máxima (50 m/s), o rastro deixado será de cor amarela, neste caso como a velocidade foi saturada, ela permanece constante, com o veículo movendo-se em MRU (Movimento Retilíneo Uniforme); (3^a) frear, o rastro deixado será de cor vermelha; (4^a) se não houver ação do usuário o rastro deixado será de cor azul. Esse procedimento do cenário virtual oferecer rastro tem como objetivo servir de suporte para análises gráficas do usuário. A Figura 4.12 ilustra esta situação.



Figura 4.12: Câmera aérea.

O usuário pode ainda realizar o experimento dentro da caminhonete, utilizando seu painel para visualizar os resultados da experiência, como é apresentado na Figura 4.13.



Figura 4.13: Câmera interna.

Existem três câmeras localizadas em pontos pré-definidos no cenário virtual que permanecem fixas e acompanham o movimento da caminhonete durante seu trajeto. Esse artifício orienta a reflexão do usuário sobre os conceitos de referencial e movimento relativo. A Figura 4.14 exemplifica a situação em que a câmera está em um destes pontos referenciais.