

2 Referencial Teórico

2.1. SHDM

A abordagem proposta neste projeto estende a metodologia SHDM (*Semantic Hypermedia Design Method*, Lima & Schwabe, 2003) possibilitando a definição de transições animadas a partir do artefato produzido pela etapa de interface abstrata, detalhada posteriormente. Neste capítulo descrevemos este método.

O SHDM (*Semantic Hypermedia Design Method*, Lima & Schwabe, 2003) é um método de desenvolvimento de Aplicações Hiperímídia para Web Semântica. Ele foi desenvolvido com base em outro método de elaboração de aplicativos hiperímídia conhecido como *Object Oriented Hypermedia Design Method* (OOHDM, Rossi, et al., 1999, Schwabe & Rossi, 1998) mantendo assim seus fundamentos e estendendo suas etapas de elaboração com mecanismos de Web Semântica. A abordagem do SHDM, assim como o OOHDM, é fundamentada em modelos, sendo composta por cinco etapas: Levantamento de Requisitos, Modelagem Conceitual, Modelagem Navegacional, Projeto de Interface Abstrata e Implementação. Cada uma destas etapas representa uma abstração de partes da aplicação, produzindo modelos que descrevem cada uma delas. Na Tabela 1, mostramos os artefatos produzidos por cada etapa da abordagem.

Tabela 1 – Etapas da abordagem SHDM e artefatos gerados (Lima & Schwabe, 2003)

Etapas	Artefatos
Levantamento de Requisitos	Cenários; Diagramas de Interação com usuário.
Modelagem Conceitual	Esquema Conceitual; Ontologia Conceitual; Instâncias Conceituais.

Modelagem Navegacional	Esquema de Classes Navegacionais; Esquema do Contexto Navegacional; Ontologia Navegacional; Instâncias Navegacionais.
Modelagem de Interface Abstrata	Definição de Interface Abstrata; Ontologia de widgets de Interface Concreta.
Implementação	Aplicação final para execução em um ambiente específico

Descrevemos melhor a seguir cada uma das etapas do método.

2.1.1. Levantamento de Requisitos

O levantamento de requisitos é uma etapa crítica para o sucesso de desenvolvimento de uma aplicação. Nela são definidos os requisitos do sistema. Para este processo, temos que analisar inicialmente os requisitos daqueles envolvidos no projeto, identificando os atores e as tarefas que eles realizarão. Em seguida especificamos cenários para cada ator identificado na forma de *user cases*, os diagramas de interação de usuários (UID). Estes diagramas representam graficamente a interação entre o usuário e o sistema durante a execução de uma tarefa.

2.1.2. Modelagem Conceitual

Nesta etapa utilizamos princípios de orientação de objetos, junto a algumas primitivas como perspectivas de atributivos para definição conceitual do domínio da aplicação e caracterização dos elementos de informação. As classes conceituais podem ser elaboradas utilizando hierarquias de agregação, generalização/especialização. Nesta fase, não consideramos os usuários ou as tarefas realizadas, focando apenas na semântica do domínio da aplicação. Ilustramos na Figura 1 abaixo um exemplo de modelo conceitual elaborado para o domínio de filmes.

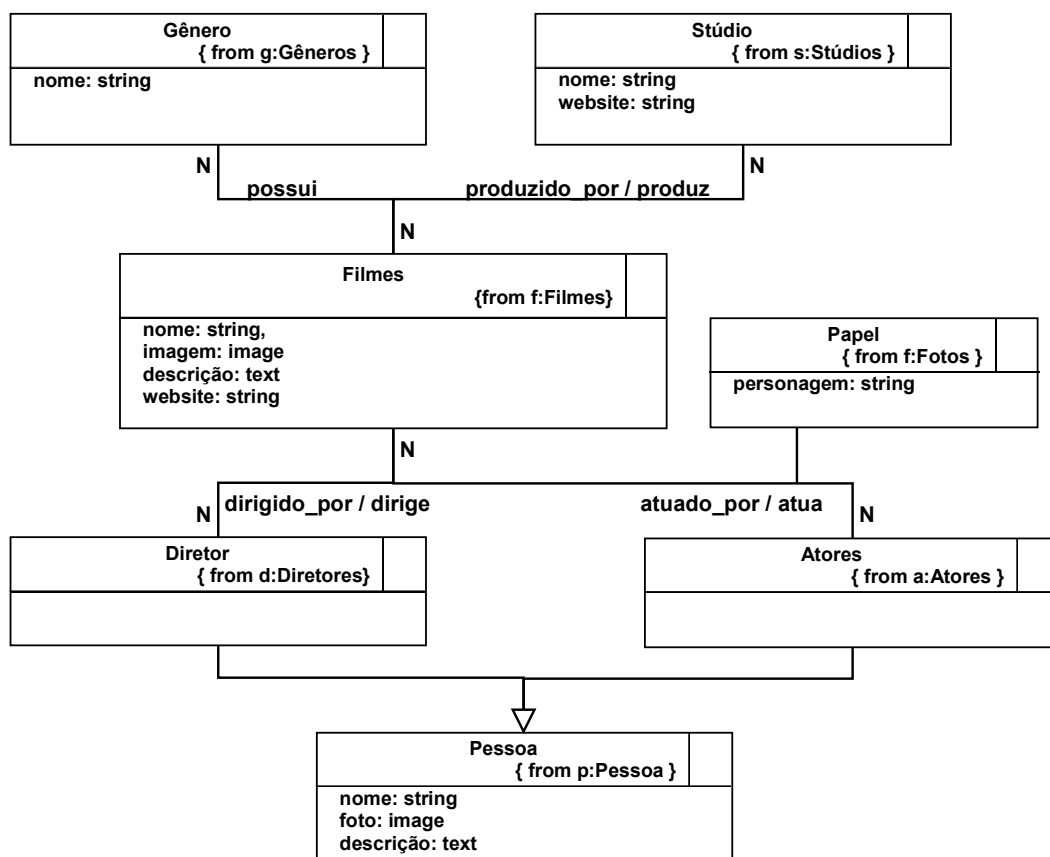


Figura 1 – Modelo conceitual para o domínio de filmes

2.1.3. Modelagem Navegacional

Durante esta etapa descrevemos a estrutura navegacional da aplicação em termos de contextos navegacionais, elaborados a partir das classes navegacionais. Os contextos navegacionais e classes navegacionais levam em consideração os usuários envolvidos e suas tarefas, especificando as informações que serão processadas e as possíveis navegações que poderão haver no sistema. Os nós do modelo representam visões lógicas da classe conceitual definidas durante a análise do domínio. Ao elaborar o modelo navegacional estamos interessados em definir os seguintes itens (Tabela 2) abaixo.

Tabela 2 – Primitivas do modelo navegacional

Estruturas navegacionais	Definição
Nós navegacionais	Os objetos que podem ser acessados pelo usuário.
Os links	As relações existentes entre os nós navegacionais.

Os contextos	Os conjuntos de objetos sobre o qual o usuário irá navegar.
As estruturas de acesso	Definição de como os conjuntos serão acessados.
As classes em contexto	Os conteúdos distintos que deverão ser apresentados ao usuário, dependendo do contexto em que ele estiver.

Ilustramos na Figura 2 um exemplo de um esquema de contextos navegacionais realizados a partir do modelo conceitual apresentado na Figura 1.

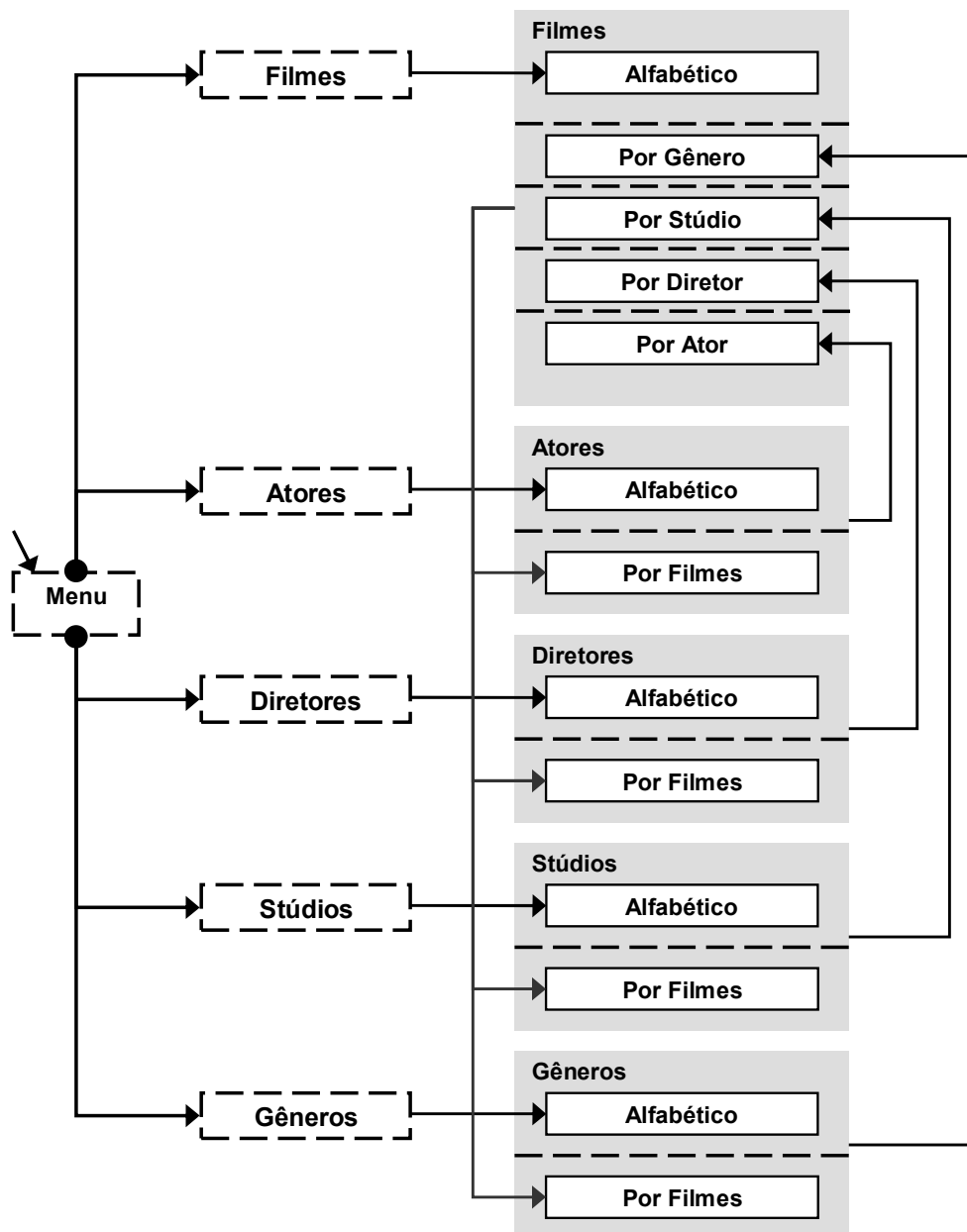


Figura 2 – Esquema de contexto navegacional para o domínio de filmes

2.1.4. Modelagem de Interface Abstrata

Como descrito anteriormente, a modelagem conceitual se foca na caracterização dos elementos de informação do domínio da aplicação, a modelagem navegacional foca no suporte ao usuário, para completar as tarefas desejadas. Já na modelagem de interfaces abstratas do método SHDM/OOHDM propostas por Moura & Schwabe (2004), focamos em fazer com que os elementos navegacionais e as funcionalidades da aplicação, sejam apresentados ao usuário. Isso só pode ser realizado na definição da interface da aplicação.

Podemos considerar de forma mais abstrata, que a funcionalidade da interface é apoiar a troca de informações entre a aplicação e o usuário, incluindo a ativação de funcionalidades. Como as tarefas oferecidas delegam esta troca de informações, esperamos que estas sejam menos sensíveis a aspectos de tempo de execução, como os padrões e dispositivos utilizados, podendo então ser realizados separadamente por um designer de interação ou engenheiro de software (Moura & Schwabe, 2004).

Em um nível mais concreto, entretanto, é necessário definir o aspecto de apresentação da aplicação, como *layout*, cores, fontes, elementos gráficos, etc. Esta etapa será realizada por designers gráficos, tendo que considerar o ambiente e as tecnologias utilizadas durante o tempo de execução. A separação entre esses níveis abstrato e concreto, permite isolar parte da modelagem da interação da inerente plataforma tecnológica associada, bem como a necessidade de dar suporte a usuários nos diversos ambientes de execução.

A especificação da interface abstrata é realizada através da Ontologia de Interfaces Abstratas, que estabelece um vocabulário para descrição dos elementos que participam de uma interface, em termos de funções que assumem durante a interação, sendo descritos então como *widgets*.

Os *widgets* de Interfaces abstratas podem ser dos seguintes tipos a seguir:

- *SimpleActivator*, aquele que reage a eventos externos (ex.: o clique do mouse), tendo uma ação associada.
- *ElementExhibitor*, aquele que simplesmente exhibe algum conteúdo como textos ou imagens.

- *VariableCapturer*, permite receber ou capturar o valor de uma ou mais variáveis como campos de texto, caixas de seleção, *check boxes*, etc. Esse tipo generaliza duas classes distintas:
- *IndefiniteVariable*, permite a entrada de valores variáveis e desconhecidos, como um texto digitado pelo usuário.
- *PredefinedVariable*, corresponde a widgets que permitem a seleção sobre um conjunto de dados predefinidos. Estes podem ainda ser definidos como *ContinuousGroups*, *DiscreetGroup*, *MultipleChoices* e *SingleChoice*.
- *CompositeInterfaceElement*, representa uma composição de quaisquer dos widgets anteriores.

Podemos ilustrar a ontologia de widgets abstratos na Figura 3. Através da ontologia, procuramos possibilitar a representação de todas as interações do usuário, sendo a interface abstrata, uma instância desta ontologia.

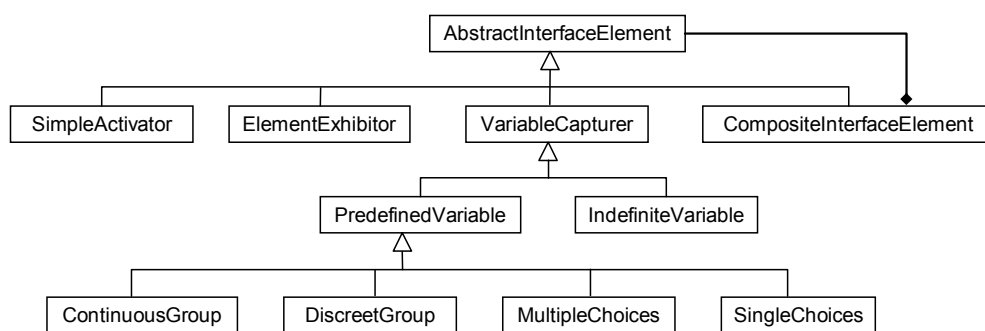


Figura 3 – Elementos que compõe a ontologia de widgets abstratos

Depois que a interface abstrata esta descrita, cada elemento deverá ser mapeado para um elemento navegacional, que proverá seu conteúdo, e a um widget de interface concreto, que define o elemento no ambiente de execução. O objetivo da ontologia de widgets concretos (Moura & Schwabe, 2004) é descrever widgets que costumam estar presentes na grande maioria das interfaces gráficas das aplicações. A ontologia de Widgets abstratos utilizará dessa ontologia para especificar sobre qual elemento concreto o elemento abstrato poderá ser mapeado. Os elementos da ontologia são apresentados abaixo, sendo que o conjunto pode

ser entendido com novos widgets que estejam disponíveis pra determinado ambiente ou aplicação.

- *Button;*
- *CheckBox;*
- *CheckBoxAction;*
- *ComboBox;*
- *ComboBoxAction;*
- *ComboBoxTarget;*
- *Composition;*
- *Form;*
- *Image;*
- *Label;*
- *Link;*
- *RadioButton;*
- *RadioButtonAction;*
- *RadioButtonTarget;*
- *TextArea;*
- *TextBox.*

Cada um destes elementos da ontologia é descrito conforme o nome indica. É importante ressaltar que esta ontologia está apresentará de forma superficial, pois não é função do modelo sintetizar a interface concreta de forma automática, mas permitir que o designer faça as escolhas durante o mapeamento. Esse mapeamento é definido através da especificação da propriedade correspondente, presente em todos widgets de interface abstratos.

2.1.5. Implementação

A implementação é a etapa final da abordagem SHDM, nela definimos a arquitetura do sistema e mapeamos os objetos de interface em objetos de implementação, elaborando o aplicativo final executável.

2.2. Animação

Existem várias definições sobre o que é a animação, porém, é comum a idéia de que a animação tem como origem um conjunto de diversas imagens estáticas, que quando exibidas em seqüência criam a ilusão de continuidade e movimento. Beacker & Small (1990), afirmam que todos os tipos de animações resultantes de quaisquer mídias, sejam originados de técnicas tradicionais, como pinturas, modelos de argila (*stop motion*), geradas por computador, por um *flipbook*, gravadas ou exibidas em uma tela de computador, são compostas por uma seqüência de imagens estáticas, que mudam com velocidade suficiente para criarem a ilusão de uma imagem que se transforma continuamente. A percepção da animação é resultante da persistência da visão ou persistência retiniana, fenômeno onde as imagens formadas na retina persistem por algum momento após serem adquiridas, permanecendo mesmo após o desaparecimento da imagem original. Assim, quando um conjunto de imagens é visto em seqüência elas parecem se mesclar criando a ilusão de continuidade.

Beacker & Small (1990), afirmam ainda que a Animação não é a arte de desenhos que se movimentam, mas a arte de movimentos que são desenhados. O que acontece entre cada quadro é mais importante do que o que existe em cada um destes quadros. Animação é então a arte de manipular os espaços invisíveis que existem entre os quadros.

O movimento sempre foi inerente à percepção humana e sendo assim, o homem sempre teve interesse em representá-lo. Williams (2002) aponta os primeiros indícios de animações criadas. Desenhos rupestres de 35.000 anos atrás apresentam animais com quatro pares de patas, quando o homem já tentava ilustrar o movimento. Ilustrações egípcias mostram figuras humanas em posições diferentes, formando uma seqüência sobre pilares e permitindo que pessoas percebessem a ilusão de movimento ao passarem rapidamente por elas. Os gregos decoravam jarros com figuras representando ações sucessivas, que ao serem girados, exibiam uma figura em movimento. Com as diversas inovações tecnológicas, as primeiras formas de captura de imagens foram elaboradas, permitindo o surgimento das primeiras películas e desenhos animados. Nos últimos anos, a animação renasceu como uma forma de entretenimento em massa. As diversas inovações em animações por computadores estão permitindo

transformar a animação com sua forma multifacetada, em uma das principais áreas do entretenimento. Porém, como Beacker & Small (1990) citam, animações não são usadas apenas para entretenimento, ou para contar uma história, ela pode ser utilizada como forma de repassar um sentimento ou estado de humor, como diversão ou drama, como forma de vender algo, para identificação, persuasão, fornecer explicação ou para o ensino.

Em nosso projeto, pretendemos fazer o uso destas animações em aplicações hipermídia, possibilitando assim, uma forma alternativa para apresentação das alterações de interface e dando ao usuário a possibilidade de identificar melhor as mudanças que ocorrem durante a interação.

2.2.1. Interfaces Dinâmicas

A animação tem se popularizado cada vez mais em sistemas computacionais, sendo bem evidenciada em aplicações que apresentam grande riqueza gráfica e interações elaboradas como jogos e simuladores. Nestes sistemas a animação se torna diversas vezes imprescindível. Porém, com o crescimento da complexidade das aplicações e sofisticação das interfaces, buscando uma melhor comunicação com o usuário, a animação tem sido cada vez mais utilizada em demais sistemas, ampliando as formas de representação de informações e demonstrando um grande potencial para elaboração de interfaces mais atrativas, agradáveis, e úteis. Temos assim hoje, um grande interesse no aprimoramento de recursos para utilização da animação em interfaces. É comum, por exemplo, observarmos seu uso extensivo em sistemas operacionais como Microsoft Windows, Apple Mac OS e Linux, que tem investido em novas tecnologias como Windows Presentation Foundation, Core Animation, e Beryl, respectivamente. Tais tecnologias permitem ampliar as possibilidades de uso da animação e auxiliam no seu desenvolvimento, identificando o crescente interesse e a relevância do uso de interfaces dinâmicas. Diversos outros sistemas para *desktop* também propõe o uso de animação com o propósito de enriquecer a interação e fornecer transições suaves (Drucker et al. 2004; Huynh et al. 2004; Regan et al. 2002).

Sistemas hipermídia também fazem vasto uso de interfaces dinâmicas. É fácil perceber que nas aplicações web, esta cada vez mais comum o uso de

elementos multimídias na forma de vídeos, sons e das animações, possibilitando interfaces com grande riqueza áudio-visual e dinamismo.

De acordo com Chen et al. (1996) existem diversas formas de inserção de animações na web. Temos animações *Inline*, que são aquelas que são apresentadas em uma página diretamente, sendo renderizadas pelo navegador, e temos também as animações executadas no cliente através da utilização de aplicações adicionais. Sendo assim, quando as animações são apresentadas o navegador deverá oferecer suporte nativo ao formato acessado, ou o usuário deverá possuir uma aplicação adicional ou um *plugin* específico para a execução.

Existem diversas tecnologias disponíveis que permitem a inserção de animações na web. Essas tecnologias estão em constante crescimento e aprimoramento, favorecendo o aumento de aplicativos com interfaces dinâmicas. Uma das primeiras tecnologias bem difundidas que possibilitou animações na web foi o tipo de arquivo de imagens .GIF (Graphics Interchange Format) que são exibidos nativamente pelos *browsers*. Elas permitiam a exibição de uma seqüência de imagens em uma única fonte, produzindo os chamados *gifs* animados. Uma tecnologia de grande destaque é o *Flash* amplamente utilizada na web para criação de aplicações com animações interativas. O *Flash* é conhecido como uma poderosa ferramenta para criação de interfaces gráficas e exige um *plugin* (*Flash player*) para visualização de seus arquivos. Ele tem sido adotado como uma das principais tecnologias para elaboração de aplicativos multimídia, sites com alto nível de interatividade, propagandas e para o desenvolvimento de *Rich Internet Applications* (RIA), aplicativos web, que como aplicativos tradicionais de *desktops*, possuem alto nível de funcionalidades. Outras duas tecnologias recentes que competem com *Flash*, apresentando funcionalidades similares são, o *Silverlight*, derivado do Windows Presentation Foundation e o *JavaFX*, uma linguagem script desenvolvida pela Sun Microsystems sobre a plataforma Java.

Outra tecnologia que se destaca, ampliando significativamente a quantidade de interfaces dinâmicas existentes é o AJAX (*Asynchronous Javascript And XML*) um conjunto de tecnologias que tem como objetivo, melhorar a interatividade, rapidez e funcionalidade da aplicação. A principal característica do AJAX é a comunicação assíncrona, através da qual é possível realizar alterações parciais das informações de interface e juntamente com o DHTML (*Dynamic HTML*), fornecer uma interação e visualização dinâmica. Existem *frameworks* tais como

script.aculo.us, que oferecem um conjunto de bibliotecas para facilitar o uso de AJAX, disponibilizando efeitos visuais para elaboração das interfaces. Uma vantagem do uso dessas tecnologias é que os navegadores têm oferecido suporte nativo para a sua execução. Problemas de padronização e compatibilidade entre diferentes navegadores, entretanto persistem.

Propomos em nosso trabalho o uso da tecnologia AJAX para execução das transições animadas, permitindo assim, que as animações sejam renderizadas diretamente pelo navegador, pois estes suportam a interpretação de comandos *javascript*, que compõem a tecnologia.

2.2.2. Vantagens no uso de Animações em Interfaces

Acredita-se que o uso de interfaces dinâmicas pode trazer diversas vantagens em relação ao uso de interfaces tradicionais estáticas. Sobretudo, as animações podem ajudar a identificar as mudanças que ocorrem durante a interação, auxiliando na interpretação das ações realizadas pelo usuário. Através dela, conseguimos representar transformações de uma forma contínua e seqüencial, utilizando movimentos e efeitos para realização de alterações graduais sem saltos entre os estados visuais.

Stafford & Webb (2004) apontam que é mais fácil para o cérebro receber diversas leituras e depois interpolar a resposta do que passar um grande período analisando uma única cena. Acreditamos então que é mais fácil interpretar mudanças realizadas na forma de animação, com estágios intermediários de informação entre dois estados distintos, do que interpretar alterações estáticas, que saltam diretamente para o estado visual final da transformação. Tidwell (2006) descreve que a mudança contínua entre estados é positiva, ou seja animar transições de forma suave e sem descontinuidade ajuda na orientação do usuário, porque assemelha a realidade do mundo físico onde não ocorre mudanças estáticas. Uma transição animada dá ao usuário a oportunidade de rastrear a mudança visual, evitando que ele tente localizar as alterações após uma mudança abrupta.

Existem estudos que revelam que a animação auxilia o usuário durante o processo de interação, melhorando a compreensão, aumentando sua eficiência e tornando-a mais prazerosa. Bederson & Boltman (1999) elaboraram testes sobre

como a animação do ponto de visão em um sistema de informação espacial afeta o usuário na construção de um mapa mental da informação no espaço, mostrando que a animação ajuda manter consistência e aprimorar as habilidades do usuário durante a reconstrução da informação espacial, sem perda de desempenho.

Gonzalez (1996) afirma, entretanto, que podem existir inconsistências nos resultados de diferentes estudos devido a vários fatores. Inicialmente, a maioria dos estudos compara o uso de informação textual com a informação gráfica dinâmica, sendo assim, diferenças de desempenho podem ser resultantes dos diferentes meios de representações e não pelo fato da animação estar presente. Outro fato é que em alguns estudos as animações são utilizadas para demonstrar tarefas óbvias e conhecidas que não oferecem qualquer desafio mental no processamento da tarefa pelo usuário. Sendo assim, os resultados podem ser afetados pelas escolhas das tarefas a serem animadas. Por fim, a maioria dos estudos é feito através da apresentação ao invés da interação, tais ambientes passivos podem gerar conclusões incorretas em relação à interação.

Como forma de testar a hipótese da melhoria do processo de decisão pela animação Gonzalez (1996) realizou um experimento para investigar os efeitos de imagens, transições e estilos de interação utilizados em interfaces animadas sobre duas visões: uma realista e outra abstrata. Quanto aos resultados, foi observado inicialmente que as imagens realistas em interfaces animadas melhoram as tomadas de decisões e tornam a interação mais prazerosa, porém estes efeitos variam conforme o tipo da tarefa a ser realizada. Foi visto também que o uso de animações suaves, apresenta vantagens sobre as animações abruptas, aumentando a eficiência das decisões. Por fim, as animações paralelas agradaram mais os usuários e proporcionaram também melhor eficiência nas decisões quando comparadas às animações sequenciais.

É importante ressaltar, conforme citado por Golnessa & Mostafa (2006), que uma grande quantidade de pesquisas surpreenderam e frustraram aqueles que promovem o uso de animações, uma vez que é raro identificar benefícios mensuráveis aos usuários com o uso de animações (Morrison et al. 2000). Outro aspecto importante, é que ao utilizarem animações, os usuários geralmente apresentam alta satisfação, porém, as animações podem distrair a sua atenção, atrapalhando a sua concentração para identificação das informações principais (Weiss et al. 2000). Sendo assim, é importante realizar as animações de forma

coerente tentando evitar um desvio errôneo de atenção do usuário. Como forma de apresentar as animações de forma coerente, nos basearemos em práticas e técnicas existentes, bem como no uso da retórica para estruturação seqüencial e definição dos estilos das animações que compõem uma transição.

2.2.3. Classificação

Existem diversas formas de classificar os tipos de animação existentes. As classificações apresentadas adiante servem como base para determinação das possíveis opções de animações que estarão disponíveis, estando estas associadas ao comportamento dos elementos durante uma transição.

A classificação é geralmente derivada da função ou significado da animação. Em ferramentas de apresentação multimídias é comum apresentar uma classificação que indica o comportamento que a animação possui durante a apresentação. Tomando como base, por exemplo, a ferramenta Microsoft Office Power Point, podemos utilizar a seguinte interpretação: Os efeitos de animação podem ser realizados sobre elementos da interface ou então sobre o próprio slide (ou *canvas*, tela). Quando realizado sobre o slide, será identificado como transição, quando realizado sobre os elementos, poderá pertencer a uma das três categorias: Entrada, Ênfase ou Saída. Todos estes efeitos são acionados a partir de uma ação ou evento podendo ter um intervalo de tempo associado. Estas três categorias estão descritas melhor a seguir:

- Efeitos de Entrada: Definem como um elemento irá se tornar visível para o usuário, podendo, por exemplo, mover para dentro da cena, crescer a partir de um ponto, etc. É importante ressaltar que quando um efeito de entrada é adicionado o elemento não é inicialmente visível, aparecendo apenas após um evento ou intervalo definido.
- Efeitos de Ênfase: Como o nome indica é um efeito de destaque que possibilita atrair a atenção do usuário a um elemento visível. Este efeito pode ser a alteração do tamanho do elemento, uma rotação, mudança de cor, etc.
- Efeitos de Saída: Ao contrário do efeito de entrada ele define como um elemento de interface desaparece da tela. O elemento pode então esmaecer (dissolver), encolher-se, mover para fora da tela, etc.

Existem ainda dois outros tipos de efeitos que podem não pertencer às três categorias citadas, o *Motion Path* e o *Object Action*. O primeiro define a movimentação de um elemento através de um caminho invisível, podendo auxiliar na apresentação do funcionamento de algum processo ou simplesmente reposicionar um elemento na cena. O segundo se refere a efeitos que o próprio elemento realiza como, por exemplo, a execução de uma mídia associada.

Beacker & Small (1990) separam as animações em três classes, de acordo com seu papel na interface: as animações de estrutura, de processo e de função.

As animações de estrutura são realizadas sobre ambientes tridimensionais com o objetivo de simular, prever e explorar como se comportam diferentes estruturas e ambientes quando animados. Nessas, podemos realizar animações sobre diferentes pontos de visão a fim de explorar as relações entre um elemento e o ambiente em que ele está inserido. Verificamos assim, como esse elemento se comporta e interage com o ambiente quando observado a partir de diferentes ângulos. Podemos também animar a aparência de um objeto ou ambiente, modificando, por exemplo, as propriedades de iluminação e permitindo então visualizar como uma estrutura se apresenta sobre diferentes condições de luz e sombra. Por fim, há também a animação de futuros alternativos, que nos permite examinar como seriam ambientes hipotéticos ou simular atividades perigosas de difícil acesso.

As animações de processo são utilizadas para visualizar o processamento de alguma função, tarefa ou aplicativo, sendo muito utilizada para visualização de algoritmos ou programas. Com ela, podemos simular a execução de uma classe de algoritmos representando através de um modelo visual elaborado, como é seu funcionamento. É possível identificar então quais as características do algoritmo, como o resultado é atingido, como está implementada a recursão ou *loop*, qual estrutura de dados utilizada e como ele se compara com outros algoritmos da mesma classe.

A terceira e última classe de animação definida por Beacker & Small (1990) é a animação de função. Animações desta classe têm o papel de facilitar compreensão da interface, minimizando sua complexidade. Ela poderá:

- Rever o que foi realizado;

- Mostrar o que pode ser realizado;
- Mostrar o que não pode ser realizado;
- Guiar o usuário, indicando o que deverá ser feito;
- Guiar o usuário, indicando o que não deverá ser feito.

A animação de função nos ajuda assim analisar o passado, entender o presente e descrever o futuro. Podemos descrever os usos deste recurso na Tabela 3 a seguir, que identifica o tipo de função, as perguntas que correspondem a o que as animações poderiam auxiliar a solucionar e um exemplo (Beacker & Small, 1990).

Tabela 3 – Funções das animações associadas a perguntas que auxiliam responder

Função	Pergunta	Exemplo
Identificação	O que é isto?	Ícones animados, que demonstram melhor suas funcionalidades.
Transição	De onde eu vim e para onde fui?	Ilustra mudanças de estado do sistema que modificam a interface como ao fechar uma janela ela encolhe até desaparecer.
Escolha	O que posso fazer agora?	Menus animados que apresentam informações de forma mais eficiente, mostrando itens de forma incremental.
Demonstração	O que posso fazer com isso?	Símbolos ou Ícones animados, que exemplificam o método de sua utilização.
Explicação	Como faço isto?	Demonstrar os passos necessários para realizar uma tarefa, como um Guia animado.
Feedback	O que esta acontecendo?	Barras de progressão ou outros indicadores animados que mostram o quanto falta para completar um processo.
Histórico	O que eu fiz?	Exibir uma animação das ações do usuário mostrando os passos que ele executou para chegar a determinado estado no sistema.
Guia	O que devo fazer agora?	Animações que mostram quais seriam os resultados de possíveis ações do usuário.

A definição das animações de função refletem melhor os objetivos do uso das animações presentes nas transições definidas pela abordagem defendida na dissertação.

Podemos ainda estender as funções das animações conforme um estudo do uso de animações para o aprendizado por computadores (Weiss et al. 2000). Para

o autor, é importante que o designer entenda quais os possíveis propósitos de aplicação de animações antes de utilizá-los. Dividimos as animações em cinco tipos de funções, estendendo seu uso para interfaces:

- **Função cosmética** – Como o nome indica, animações deste tipo tem função decorativa, podendo ser utilizadas para tornar uma informação mais atrativa, redirecionando a atenção do usuário. Esse tipo de animação pode ainda distrair o usuário das informações mais importantes, devendo então ser utilizadas com cautela.
- **Função de atração da atenção** – As animações podem também ser utilizadas para atrair atenção do usuário, podendo ser representada através de movimentos ou por efeitos sobre elementos, redirecionando a atenção do usuário para aquele elemento e realçando sua presença.
- **Função motivacional** – Animações chamativas que apresentam efeitos cativantes como forma de *feedback*, incentivando os usuários a tomarem uma decisão específica. É importante que a motivação indique as decisões corretas, para evitar que o usuário opte por ações contrárias ao objetivo desejado.
- **Função de apresentação** – A forma mais direta para aplicação da animação é utilizá-la para apresentação. Servindo como uma referência concreta ou um como um contexto visual de idéias e informações, auxiliando assim, na retenção de uma informação. Essa função é comumente utilizada para exemplificar uma ação ou processo.
- **Função de clarificação** – Animações com esta função têm alta relação com as que possuem função de apresentação. Elas oferecem uma compreensão conceitual sem adicionar entretanto, novas informações. Isto é, a animação identifica relações através de transformações visuais. Podemos assim, enfatizar abstrações de relações existentes que poderiam ser difíceis de identificar sem a animação.

2.2.4. Efeitos de Animação

Existem diversos tipos de animações que podem ser utilizadas como efeitos de transição, sendo que sua combinação pode resultar ainda em outros efeitos

mais complexos. Tais efeitos foram extraídos de diversas áreas do conhecimento em que a animação é utilizada e podem ser observados cotidianamente sendo grande parte derivada de abstrações de movimentos e efeitos que são presenciados no mundo físico.

A escolha dos efeitos irá auxiliar na definição de transições de forma a apresentar melhor coerência, possibilitando associar cada tipo de efeito específico a uma animação ou conjuntos de animações acordo com seu papel na transição. Este processo está associado à retórica da animação, definida adiante.

Beacker & Small (1990) afirmam que podemos tomar como base modelos de linguagem cinematográfica, que permitem diversas opções de comportamento de uma interface. Podemos utilizar recursos globais na tela, tais como *Cut, Fade in (out), dissolve, wipe, overlay, multiple exposure*. E localmente, em uma região definida na tela, *pop on (off), pull down(up), flip, spin*. E por fim, tanto globalmente quanto localmente, *reverse video, color changes, scrolling, panning, zooming e close-ups*, tornando as interfaces, mais vivas, atraentes, e prazerosas.

Representaremos alguns efeitos de animações, em sua maioria, associados ao *framework* de animação do script.aculo.us. Esta abstração de efeitos representa bem os tipos de animações disponíveis para utilização durante em transições de interface.

Todo elemento gráfico possui características específicas como tamanho, posição, cor, opacidade, etc. Estes parâmetros podem ser alterados de forma incremental definindo um efeito de animação núcleo, ou seja, uma transformação básica realizada sobre um elemento. Descrevemos a seguir alguns destes tipos de animações.

- **Fade** – O efeito dissolver ou fade, onde ocorre a alteração do parâmetro de opacidade de um elemento gráfico, fazendo com que ele desapareça ou apareça gradualmente .
- **Move** – O movimento ou translação é o efeito de deslocamento, realizado através da alteração dos parâmetros de posicionamento horizontal e/ou vertical do elemento de forma consecutiva.
- **Resize** – O redimensionamento, como o nome indica, realiza a alteração seqüencial dos parâmetros de tamanho do elemento gráfico.

- **Alterações de Cor** – Temos também animações sobre os parâmetros de cor, que podem alterar um elemento gráfico, atingindo diversos efeitos diferentes, como: realce, escurecimento, clareamento, etc.

Podemos ainda realizar todos estes efeitos núcleos ou básicos simultaneamente sobre um único elemento gráfico a fim de igualar um conjunto de parâmetros determinado. Se realizarmos essa combinação para igualar os parâmetros de dois elementos distintos, podemos ter um efeito de metamorfose, onde um elemento se transforma em outro.

Existem outros efeitos que são obtidos a partir de uma combinação de efeitos básicos. Estes efeitos são realizados sobre os elementos gráficos para identificar sua entrada na cena, sua saída e ou transformações como reforçar sua presença, ou apresentar a alteração de alguma característica. Descrevemos alguns destes efeitos a seguir, sendo que a maioria destes efeitos pode ser visualizado no site www.script.aculo.us

- ***Fade-in (out)*** – Realizam transformações fade para representar o efeito de entrada ou saída de um elemento gráfico.
- ***Shrink/Grow*** – São efeitos de entrada e saída de um elemento realizados a partir do seu redimensionamento, encolhendo ou crescendo a partir de um ponto.
- ***Puff*** – É um efeito realizado para identificar a saída de um elemento, obtido pela combinação dos efeitos de redimensionamento e fade. Nele um elemento cresce e desaparece gradualmente, como uma nuvem de fumaça que se desfaz.
- ***Drop-Out*** – Uma transição de saída sobre um elemento que combina uma animação de movimentação vertical e um fade simultaneamente. Simula assim, a queda deste elemento enquanto o mesmo desaparece.
- ***Shake*** – É um efeito realizado como forma de destacar um elemento. Nela ocorre uma combinação de movimentações horizontais sucessivas, sacudindo assim o elemento de um lado para o outro.
- ***Pulsate*** – Realiza uma combinação dos efeitos *fade-in* e *fade-out* sobre um elemento gráfico para destacá-lo. O resultado é que o elemento pulsa, aparecendo e desaparecendo sucessivamente.

- **Blind** – São efeitos de entrada e saída de um elemento, onde o elemento é revelado ou escondido a partir de uma de suas bordas. Atingido através de uma transformação de redimensionamento sobre tais bordas, ou canvas que envolve o elemento.
- **Slide** – Identifica também a entrada ou saída de um elemento realizado a partir da animação de movimento no elemento gráfico, mostrando-o a partir de uma borda que delimita sua área visível.
- **Wipe** – No domínio de edição de filmes e na cinematografia definimos ainda este efeito, que ocorre quando uma imagem esta sobreposta à outra e desejamos realizar uma transição entre estas imagens, ocultando a imagem inicial e revelando a imagem final. Este efeito é comumente realizado através de transformações geométricas onde ocorre a contração (fechamento) ou dilatação (abertura) de uma forma geométrica específica (círculo, quadrado, estrela, etc.) a partir de um ponto inicial da imagem, revelando uma segunda imagem.

Podemos também realizar qualquer combinação dos efeitos citados anteriormente para transição entre dois elementos.

2.2.5. Retórica em Animações

Como descrito anteriormente, as animações podem trazer benefícios para os usuários, porém, estas animações devem ser elaboradas de forma coerente para que as transformações sejam interpretadas corretamente. Sendo assim, temos que definir quais melhores efeitos deverão ser empregados, qual a duração das animações realizadas e até qual a melhor seqüência de ocorrência. Todas estas escolhas deverão ser realizadas de acordo com a intenção que se deseja passar, como por exemplo: indicar alterações do estado, relações entre elementos, destacar uma informação, realizar uma mudança de forma mais atrativa, etc. Descrevemos nesta seção como a retórica pode auxiliar na definição destas propriedades e, sobretudo, como um conjunto de animações pode ser estruturado.

A retórica surgiu como uma técnica ou arte de persuasão, enfatizando a dimensão pragmática do discurso oral ou da escrita. Podemos referir a retórica como o estudo do discurso, da interpretação e da comunicação eficiente.

Embora o uso de retórica esteja frequentemente associado a textos e discursos, ela pode ser empregada em diversos domínios. Dormann (1994) defende o uso de uma perspectiva retórica para desenvolvimento de animações, especificamente para signos animados (ícones) e ajuda animada. Para o autor, existem poucos guias ou regras claras para auxiliar na criação de animações, que considerem tamanho, conteúdo, ordenação e representação visual de forma a garantir uma comunicação visual efetiva. As comunicações podem ser expressas como transferências de mensagens, a semiótica explica os princípios que definem a estrutura de signos e sua utilização como mensagens, e a retórica sugere modos para construção e apresentação destas.

O uso de retórica clássica já foi adaptado para a retórica visual, mostrando sua aplicação na comunicação visual. Marcus (1979) apresenta uma metalinguagem para retórica visual, elaborada com base no seqüenciamento de imagens e de parâmetros sintáticos, como tamanho, orientação e detalhamento. Dormann (1994) destaca alguns princípios de retórica visual relacionados, que podem ser utilizados em animações de interfaces, mas que são frequentemente associados ao domínio de desenhos animados:

- Metáfora – Uma comparação entre dois itens de natureza distinta
- Personificação – Associação de características humanas a objetos inanimados.
- Ironia – Um símbolo que apresenta um significado oposto ao seu significado real.
- Hipérbole – Representação exagerada de um objeto com relação a suas características naturais.
- Trocadilho – Imagens similares, que apresentam uma mesma forma, porém com significados diferentes.

Muitos aspectos no estudo de retórica para utilização em documentos e demais apresentações foram elaborados. Mann e Thompson (1988) especificaram uma coleção de relações retóricas denominadas Teoria de Estrutura Retórica (*Rhetorical Structure Theory*, RST), uma abordagem de descrição lingüística para a determinação da organização de discursos. O RST oferece uma explicação sobre

a coerência de textos, definindo para toda parte do texto, uma função e uma razão plausível e evidente para o leitor, para sua presença na estrutura.

RST é focada na descrição de textos, podendo representar diversas possibilidades de estrutura, através da identificação dos “blocos de construção” contidos nos textos. Ele se refere à organização do texto através das relações existentes entre as partes distintas desse texto e apresenta coerência postulando uma estrutura de conexão hierárquica, onde cada parte possui um papel ou função a exercer com relação as outras partes do texto (Taboada e Mann, 2006).

A abordagem permite a análise através de uma notação que interconecta todas as partes do texto em um único conjunto. A análise é geralmente elaborada interpretando-se o documento e construindo um diagrama conforme ilustrado na Figura 4.

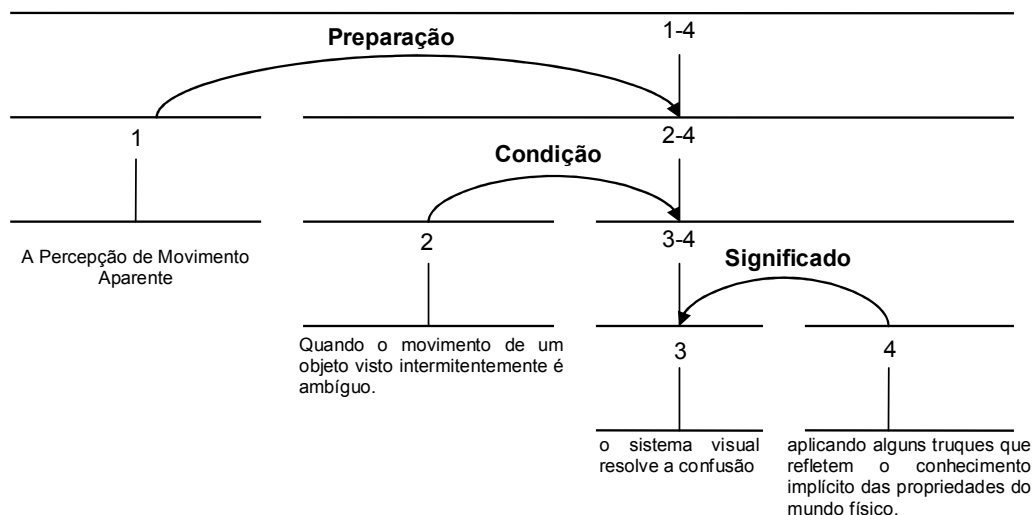


Figura 4 – Análise retórica de um texto utilizando RST

Existem dois tipos de unidades que podem ser definidas no RST. O núcleo, considerado a parte mais importante do texto e os satélites, unidades secundárias que contribuem com o núcleo. O núcleo é geralmente essencial para apresentar o propósito do autor com relação aos satélites, que geralmente não são compreendidos sem a presença do núcleo. Podem ocorrer ainda relações com múltiplos núcleos (multinuclear), onde nenhuma unidade apresenta maior importância em relação à outra. As relações podem ser definidas de acordo com quatro campos:

1. Restrições sobre o núcleo;
2. Restrições sobre o satélite;

3. Restrições sobre a combinação de núcleo e satélite;
4. Efeito.

Para especificar cada campo para qualquer instância de uma relação específica o analista deverá realizar decisões baseado na intenção do autor. Todas as relações são definidas conforme os quatro campos e as definições são realizadas com base no critério semântico e funcional, desconsiderando sinais sintáticos ou morfológicos (Taboada e Mann, 2006).

A análise retórica da comunicação consiste em dividi-la em atos comunicativos, que são as unidades onde as informações são apresentadas. Dispositivos de comunicação são padrões de atos comunicativos que expressam uma retórica específica. Assim, podemos quebrar qualquer forma de comunicação, como por exemplo, documentos hipertexto, em uma coleção de atos (Maybury, 1993). As mídias para comunicação incluem sons, textos, vídeos, desenhos, documentos hiperímídia, etc. Assim, a análise retórica, utilizando RST, não se restringe aos textos, ela tem ampliado seu uso a novos tipos variados mídias (Taboada e Mann 2006, 2).

A utilização de RST para auxílio na elaboração de animações pode ser visto no estudo de Kennedy e Mercer (2002). Os autores desenvolveram um sistema que utiliza o conhecimento de técnicas de cinematografia para aprimorar a expressividade de animações. O sistema demonstra a habilidade de alterar as animações, comunicando informações de emoção e intenção, de forma a alterar a percepção do espectador. A aplicação dos conhecimentos é realizada através de um planejador de atos comunicativos que se apóia em técnicas de RST.

Como um exemplo de como os conceitos de retórica podem ser utilizados em uma cena, considere a relação denominada Elaboração, que é definida na forma:

- Núcleo: Informação básica;
- Satélite: Informação adicional.

A Elaboração ocorre quando uma ou mais partes de um texto adicionam informação sobre outra parte do texto. No planejador, essa abordagem seria utilizada ao querer apresentar um conceito adicional e não óbvio. Por exemplo, quando requisitado a mostrar que um personagem está infeliz, o planejador

poderia elaborar a animação utilizando efeitos de iluminação fraca e um esquema de cores sem brilho sobre o personagem.

Quando utilizamos RST no domínio visual, é necessário adaptar algumas relações retóricas ou até mesmo criar novas relações específicas. Para a utilização do RST na área de animação cinematográfica Kennedy e Mercer (2002) definiram três correspondências entre as características da retórica em textos e em animações:

1. Autor \Rightarrow Animador;
2. Texto \Rightarrow Imagens;
3. Frases ou Sentenças \Rightarrow Apresentação e estrutura da cena (técnicas cinematográficas).

Estas analogias permitem aplicar os princípios da abordagem nos conceitos de comunicação visual. Assim, é possível utilizar uma estrutura para elaborar sobre altos níveis de atos de comunicação, produzindo uma representação coerente para tal.

O planejamento realizado no sistema é similar ao planejamento exigido para produção de um parágrafo de texto. O planejador reúne os passos de planos necessários (relações retóricas) em um *framework* lógico, combinando essas relações em uma árvore RST que atinge os objetivos dos atos comunicativos do animador. O plano é então uma estrutura retórica que implementa um plano de ações para comunicar algo ao espectador. Uma das principais tarefas deste planejador é criar um plano onde não haja contradições, evitando conflitos que podem ocorrer ao interpretar a árvore RST definida (Kennedy e Mercer, 2002)

Dada tais funcionalidades do uso da retórica, propomos seu uso para definição das transições, identificando os conjuntos de animações que pertencem a transição e especificando então, parâmetros como ordem, efeitos e duração, conforme apresentado na seção 3.2.2.3. Essa identificação é realizada da mesma forma em que definiríamos palavras que compõem uma sentença. Para definição retórica da transição, propomos inicialmente a divisão das animações que fazem parte da transição em estruturas retóricas de acordo com sua função ou papel na transição, como por exemplo, um conjunto de animações que determinam a inserção de novos elementos ou animações que identificam a principal alteração

durante a transição. A definição das estruturas é um processo iterativo e deverá ser definido segundo as escolhas do designer, sendo assim, uma etapa resultante de suas opiniões e preferências. Propomos conforme descrito na seção 6.3 , um estudo mais detalhado sobre diferentes formas de determinação de tais estruturas, estipulando guias para auxiliar durante sua definição.

Uma vez que as estruturas estão definidas podemos então utilizar de regras de RST como base para a definição da seqüência de execução dos conjuntos. Neste trabalho não foi realizado um estudo sobre quais seriam as melhores estruturas de conexões hierárquicas existentes, bem como das relações retóricas específicas às transições, sendo assim, destacamos o uso da estrutura retórica como uma possível alternativa para organização hierárquica de uma transição, reconhecendo entretanto a necessidade de um estudo mais detalhado para identificar qual o melhor processo de obtenção da estrutura.

Após definirmos a retórica da animação quanto a ordem, podemos também definir quais os efeitos que serão apresentados sobre cada animação, assim como qual a sua duração, conforme apresentado na seção 3.2.2.4. Tais características podem ser definidas para cada estrutura como forma de apresentar um significado para as animações dos conjuntos definidos conforme sua função na transição. Para este processo, nos apoiamos nas técnicas, práticas e padrões de animações descritas nas próximas seções, que auxiliam na definição da animação quanto à intenção que se deseja apresentar durante a alteração realizada.

2.2.6. Padrões de Transição

A idéia de padrões ou *patterns* foi introduzida pelo arquiteto Alexander (1979), como forma de capturar a essência das soluções encontradas para problemas recorrentes de design na arquitetura urbana. Os padrões definidos na arquitetura urbana descrevem aspectos do mundo físico onde as pessoas vivem e trabalham. O artefato que o arquiteto desenvolve é algo que os habitantes interagem diretamente e onde vivem. No design de Interação Humano-Computador (IHC) ocorre algo similar: A interface é o artefato produzido pelo designer, sendo algo em que os usuários atuam ou interagem (Borchers, 2000).

Padrões descrevem um conjunto das melhores práticas para utilização em um domínio de design específico. Eles capturam as soluções comuns para casos

elaborados e recorrentes e não apresentam novidades de solução no domínio. Quando temos um conjunto abrangente e complexo de padrões específicos, podemos definir uma linguagem de padrões, que cobrem todo vocabulário de elementos utilizados durante o design.

Existem diversos padrões disponíveis para design de interfaces, o seu uso em transições, entretanto é recente, sendo fortemente associado a o uso aplicações da tecnologia AJAX. Destacamos a seguir o padrão para transições (*Yahoo Design Patterns Library*) elaborado pelo centro de design e experiências de usuário do Yahoo! As soluções apresentadas pelo grupo são testadas e iteradas em um processo de pesquisa realizado com auxílio de protótipos e testes de usabilidade, sendo então avaliadas, refinadas e apresentadas como possíveis *patterns*.

Nos padrões de transições definidos, descrevemos o foco do problema da seguinte forma: O usuário está alterando ou direcionando o foco para uma tarefa realizada ou necessita perceber que ocorreu uma mudança de estado na aplicação. O designer pretende então, manter o usuário atento a uma mudança de relação entre elementos, de um estado da aplicação, desacelerar a interação ou fazê-la parecer mais acelerada.

Seus padrões se baseiam na lógica de que, as transições, são uma poderosa forma de comunicar alterações de relações entre elementos em uma página, ou uma mudança do estado de um elemento. Adicionalmente, quando as interações são complexas ou envolvem transformações importantes, as transições podem ser usadas para desacelerar o processo de interação, possibilitando ao usuário perceber o que ocorreu. Durante processos demorados, as transições podem também fazer o tempo parecer passar mais depressa, distraindo o usuário durante o tempo de espera. Descrevemos alguns exemplos levantados que apontam como as transições podem comunicar informações durante a interação:

- Se um objeto desvanecer, sabemos que ele mudou do estado visível para o invisível mesmo que não estejamos olhando para ele diretamente.
- Se um objeto aparece lentamente em uma *view*, sabemos que este objeto chegou, passando a compor a interface.
- Se um objeto um objeto desvanecer rapidamente ele é visto com maior importância, se for lentamente, apresenta importância menor.

- Se um objeto está vindo em nossa direção (expandindo seu tamanho como se fosse passar por nós) consideramos que seja algo importante (ou perigoso).
- Se um objeto encolhe seu tamanho rapidamente até desaparecer, ele irá capturar a atenção do usuário imediatamente. Porém, se o objeto não estava no foco imediato do usuário (o usuário não estava manipulando esse objeto diretamente) ele poderá perceber a transformação, mas poderá nem mesmo saber qual objeto desapareceu.

De acordo com os estudos, é importante que o objetivo da comunicação seja mantido durante qualquer efeito de transição. Em sites mais atrativos ou de entretenimento, as transições podem ser importantes, exigindo transições mais longas ou mais efeitos. Já em sites orientados à comunicação, a transição é secundária, sendo utilizada apenas para comunicar alterações na interface. Com estas considerações é possível extrair um conjunto de regras gerais, descritas a seguir:

- Quanto mais rápida a mudança mais importante o evento;
- Movimentos rápidos apresentam mais importância do que alterações rápidas de cor;
- Movimentos em direção ao usuário são visto com maior importância do que movimentos que se distanciam do usuário;
- Alterações muito lentas são processadas sem interromper a atenção do usuário;
- Movimentos podem ser utilizados para comunicar onde é a localização de um objeto. Ao ver um objeto se mover de um lugar ao outro nós compreendemos para onde ele foi sendo possível localizá-lo no futuro.

2.2.7. Práticas para Transição

De acordo com Weir e Heeps (2003), com a utilização de técnicas e formas de expressões apropriadas, animações podem apresentar informações de maneira mais eficientes. Considerando diversas fontes potenciais que definem meios mais efetivos para auxiliar animadores, eles reuniram diversas práticas e guias para

utilização prática na elaboração de animações. A pesquisa dos autores resultou em uma coleção de dez princípios ou “mandamentos” para utilização pelos animadores:

1. Não distraia a atenção de informações salientes
2. Evite desordem (amontoamento)
3. Permita um tempo de exposição apropriado
4. Administre o posicionamento e organização dos Objetos
5. Adere a convenções de cores
6. Suporte as animações com informações textuais ou sonoras.
7. Aprecie as visões da semiótica
8. Concorde com os máximos Co-operativos
9. Aprenda com a Disney
10. Evite a miopia de design

Um estudo sobre o impacto do uso de animações web (Golnessa & Mostafa, 2006) aponta algumas práticas para sua utilização, bem como algumas características observadas. Podemos descrever através deste estudo os seguintes pontos observados:

- Animações devem ser direcionadas a aspectos essenciais da animação, para que os usuários foquem nos aspectos relevantes da animação. A realização de animações sobre informações adicionais desnecessárias não apresenta vantagem.
- É necessário proporcionar um tempo de exposição apropriado para a animação. O olho humano necessita de tempo para processar alterações e mudanças rápidas poderão apenas dividir a atenção no cérebro entre as imagens antes e depois a animação.
- A alteração de cor deverá ser utilizada primariamente para identificar elementos ou aspectos relevantes. Alterações desnecessárias de cor podem distrair ou confundir os usuários; a utilização de cores deverá fornecer informações de representação adicionais para um conceito de animação específico.

- Animações devem ser utilizadas junto a textos, sons ou narrações, para facilitar a interpretação das transformações. Animações isoladas correm o risco de gerar ambigüidade, diferenciando as interpretações do usuário.
- As animações devem manter níveis apropriados de abstração. Elas não devem ser realizadas sobre muitos objetos ou realizar ações de formas muito detalhada, causando excesso de informações.

Destacamos ainda alguns princípios descritos por Zongker e Salesin (2003) para utilização em apresentações multimídia. Os princípios foram elaborados para obter apresentações com animações coerentes evitando sua utilização desnecessária ou incorreta. Embora estas observações tenham sido feitas para apresentações, não exigindo interação direta do usuário, podemos citar alguns princípios que podem ser aplicados em animações de interface.

- Tornar todo movimento significativo – Técnicas tradicionais de animação utilizadas durante um movimento tais como exagero e distorção tendem a dar vida a um desenho, porém ele é considerado, uma distração quando a intenção é informacional. Tais efeitos instigam a curiosidade sobre qual será o próximo movimento. Assim, devemos ter cautela para evitar que o destaque seja mais chamativo do que outra informação prioritária.
- Evitar mudanças bruscas – O uso de transições suaves é recomendado sempre quando algo será acrescentado ou retirado de cena. Cortes ou mudanças bruscas entre estados podem fazer com que o expectador perca o foco da atenção.
- Reforçar a estrutura através de Transições – As animações podem reforçar a estrutura do que esta sendo apresentado servindo como um tipo de pontuação. Assim, animações sutis e simples indicam mudanças de mesma magnitude, e animações mais dramáticas, destacam uma quebra visual indicando maior distância semântica durante a alteração.
- Criar uma grande área virtual – Podemos utilizar de recursos como *panning* e *zooming* para facilitar a representação de informações que não cabem em uma mesma tela, mas possuem o mesmo contexto. Uma informação que desliza para fora da tela permanece mais "visível" na

mente de um usuário do que uma informação que simplesmente desaparece.

- Expandir e contrair detalhes suavemente – Semelhante à propriedade anterior, podemos representar elementos gráficos relacionados, como por exemplo, figuras de diferentes escalas, através de uma transformação sequencial, sem substituí-la. Assim, podemos representar uma grande quantidade de informações e ampliar de forma análoga a uma lente de aumento, as informações que desejamos mostrar. Exigindo menos esforço cognitivo do usuário para interpretar diferentes visões.
- Administrar complexidade através de sobreposições – Manter as informações em Camadas (*Layers*) evita a quebra (espacial) de um diagrama. Podemos assim manipular as camadas, deixando visível aquilo que necessitamos mostrar em determinado momento, utilizando transições simples (dissolver, ou deslizar) para diferenciação das camadas.
- Realizar uma coisa de cada vez – Uma grande quantidade de animações pode dificultar a interpretação do espectador, impossibilitando-o de se concentrar em parte da animação. Sendo assim, uma animação complexa deve ser feita de forma mais lenta e em etapas.
- Reforçar animação através de narração – É importante apresentar outras informações junto às animações mantendo a consistência entre as informações descritas (os textos, imagens, etc.) e a animação, evitando assim, dois focos distintos.

2.2.8. Técnicas de Animação

Existem inúmeras técnicas e teorias provenientes de diversos domínios utilizados para elaborar animações que representam da melhor forma o significado que se deseja expressar. Podemos aproveitar diversas destas para desenvolver interfaces dinâmicas da melhor maneira, possibilitando animações mais agradáveis e um melhor entendimento das transformações pelo usuário. Um primeiro conjunto de técnicas que podem ser aproveitados é proveniente dos métodos tradicionais de animação nos desenhos animados (Chang & Ungar, 1993). Essas técnicas prometem auxiliar a contornar as dificuldades na assimilação durante mudanças na interface.

Os desenhos animados apresentam um elevado nível de informações para o espectador sem, entretanto confundir ou exigir um maior esforço para sua compreensão. Neles os personagens e objetos não desaparecem ou mudam de posicionamento instantaneamente, pois utilizam princípios de animação que permitem oferecer dicas e estabelecer ligações das ações que ocorrem, ou irão ocorrer. Podemos então aplicar essas técnicas para substituir mudanças bruscas por transições suaves eliminando assim um grande esforço cognitivo para interpretação das mudanças no sistema, onde o usuário poderá se concentrar na tarefa que pretende atingir sem um desvio do foco de atenção (Chang & Ungar, 1993).

As técnicas utilizadas em desenhos animados foram desenvolvidas e aperfeiçoadas durante anos, resultando em 3 grupos de princípios de animação: Solidez, Exagero e Reforço (Johnston & Thomas, 1981). Descrevemos a seguir tais princípios:

Solidez - Neste princípio, tratamos os personagens e objetos dos desenhos como elementos sólidos, tendo peso e massa associados, como teriam os objetos reais. Para aplicar este princípio as características de peso e equilíbrio devem ser representadas durante a animação, reforçando assim o realismo do elemento e de sua ação. Temos então duas técnicas que representam esse princípio:

- ***Motion Blur***: Quando um objeto move mais do que a metade de seu tamanho em um intervalo de poucos quadros, este efeito é adicionado para representar um rastro de movimento, preenchendo o vazio que aparece entre a antiga e a nova posição e representando então, um movimento contínuo;
- ***Chegadas e Saídas***: Objetos possuem maior realismo quando entram ou deixam a cena com movimentações suaves ao invés de simplesmente aparecerem e desaparecerem instantaneamente.

Exagero – Nos desenhos animados não existe apenas a intenção de imitar a realidade, mas de dramatizar as ações como forma de ressaltar seu significado e atingir o efeito desejado. Podemos então intensificar características significantes facilitando assim a compreensão da natureza dos objetos e das situações que ocorrem durante a interação. Descrevemos uma técnica que ilustra esse princípio.

- **Antecipação:** É uma técnica que tem como objetivo exagerar uma ação preliminar chamando atenção do espectador para a próxima ação que ocorrerá.

Reforço – Neste princípio reforçamos o realismo de uma ação de forma subliminar, mantendo o espectador atento às cenas, mas sem deixá-lo perceber os efeitos utilizados. Existem três técnicas que demonstram esse princípio:

- **Slow in e Slow out:** Algumas animações devem ser realizadas com diferenças de velocidade a fim de manter realismo e apresentar idéia de peso ao elemento;
- **Arcos:** Ao invés de realizar movimentos que obedecem a uma linha reta como trajetória, é interessante realiza-los por trajetórias com uma suave curvatura, assim, o movimento ganha vida, realismo e maior atenção do usuário;
- **Follow through:** Imitando o modo como os objetos tendem a parar na realidade devido a inércia, podemos adicionar um efeito de forma a manter a continuidade de um movimento.

Outra área de onde podemos aproveitar técnicas para elaboração de interfaces é a cinematografia (May et al. 2003). Os cinematografistas utilizam de técnicas de filmagem e edição gerando resultados que possibilitam a interpretação de uma cena sem dificuldades. Durante a mudança de cenas, eles enfatizam a necessidade de apresentar a informação visual de modo a permitir que a tarefa primária em termos de importância seja interpretada sem interrupções. Para representar os processos visuais, eles utilizam diversas técnicas tais como: a aproximação (*close-up*) para representar a atenção perceptual, *flashbacks* como memória visual e a seqüência de tomadas como atenção seqüencial direcionada em uma cena real. O autor afirma que os cinematografistas aprenderam quais formatos de cenas dinâmicas são facilmente assimiladas pelo espectador, e quais não são. Desta forma tal conhecimento oferece uma fonte de princípios que poderiam ser aplicadas a IHC. No seu estudo o autor aponta um método para identificar a prática de se colocar elementos de informação em uma mesma posição, antes e após um corte ou transição de modo a permitir uma relação entre informações durante as mudanças de interface. Como resultado, observaram que é

possível descrever efeitos a fim de manter a compressão do espectador durante a transição, adotando regras para construção da interfaces (May et al. 2003).

2.2.9. Tempo de Animação

Stasko (1993) aponta que a duração de uma animação é um fator crucial no design da animação. Tal característica afeta de forma direta a forma como a transição será representada. Animações podem expressar através de tempos diferentes, diferentes significados e, sendo assim, devemos sempre determinar qual será o tempo adequado de acordo com a transição representada.

Determinar a duração de uma animação, entretanto não é uma tarefa trivial. Devemos ter animações rápidas de forma a minimizar o retardo e não prejudicar o tempo de realização de uma tarefa, porém se esta for realizada muito rapidamente a animação pode fugir da percepção do usuário (Stasko, 1993). Acreditamos que se o usuário se perder durante a interação ele poderá necessitar então de um esforço cognitivo adicional para identificar a mudança.

De acordo com Bederson & Boltman (1999), a relação temporal entre a interação do usuário com o sistema é geralmente descrita pelo intervalo de tempo de resposta do sistema, representando a partir do instante que o usuário realiza uma ação até o instante em que o sistema inicia a exibição de resultados. Na animação entretanto, ela deverá ser executada geralmente até o final para representar uma informação útil para interpretação do usuário. Neste caso, o tempo de interação com animação é descrito através do tempo total de resposta do sistema, exibido no modelo de tempo abaixo (Figura 5).

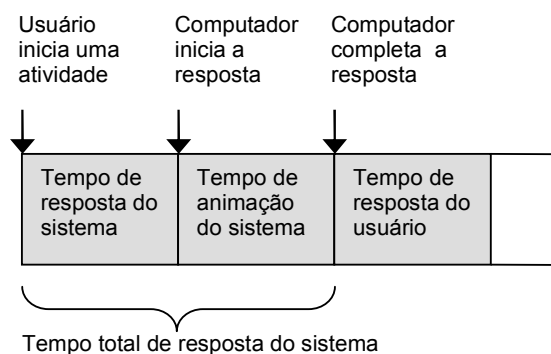


Figura 5 – Tempo de resposta total (Bederson & Boltman 1999)

No estudo realizado pelo autor para determinar o tempo de resposta total ideal, o tempo de animação apropriado de modo que não ocorra um atraso nas tarefas ou perda de desempenho, permitindo ainda a interpretação das mudanças pelo usuário, varia entre 0,5 a 1 segundo. É importante ressaltar, entretanto que o tempo de animação dependerá ainda da quantidade de alterações feitas e da complexidade das animações realizadas.