

6 Conclusões

O objetivo principal deste trabalho foi desenvolver uma ferramenta para a especificação de arquiteturas de sistemas baseadas em grafos compostos, em especial grafos com composicionalidade. A ferramenta desenvolvida pode ser aplicada diretamente no domínio de autoria de documentos hipermídia, mas pode também ser utilizada em outros domínios, como na definição de arquiteturas de sistemas de software, ferramentas de especificação formal, projetos de *Workflows*, entre outros.

No Capítulo 2, a definição de grafos compostos e o conceito de composicionalidade foi apresentado. Em seguida, algumas arquiteturas de sistemas baseadas em grafos compostos foram analisadas, destacando-se os elementos de cada arquitetura que são passíveis de serem modelados em elementos de grafos compostos.

As visões espacial, estrutural e declarativa disponíveis no ambiente de autoria do sistema apresentado nesta dissertação trabalham de forma sincronizada, de maneira que as alterações realizadas em uma das visões sejam refletidas nas outras visões. No Capítulo 3, as visões do sistema desenvolvido foram analisadas destacando as suas funcionalidades disponíveis em cada visão. Em seguida, o funcionamento do mecanismo de sincronização entre as visões foi detalhado. Por fim, um estudo sobre os compiladores no sistema foi discutido levantando as questões que devem ser analisadas no desenvolvimento de novos compiladores para diferentes arquiteturas.

A principal técnica de filtragem aplicada sobre as visões estrutural, declarativa e na especificação de leiaute da visão espacial foi a estratégia olho-de-peixe. A técnica foi estendida para trabalhar com grafos compostos de modo que, em uma única visão, os detalhes do vértice focado pelo usuário sempre sejam mostrados sem perder-se a noção da estrutura global do grafo. O Capítulo 4 analisou a técnica olho-de-peixe na ferramenta com exemplos de uso da técnica sobre as visões, assim como os cálculos realizados em cada uma das visões.

As comparações entre o sistema HyperProp (na versão atual) com diferentes ferramentas de autoria destacando os pontos positivos e negativos de cada ferramenta foi abordado no Capítulo 5.

6.1. Trabalhos Futuros

Alguns pontos dos trabalhos realizados nesta dissertação podem ser explorados como trabalhos futuros.

A utilização da ferramenta desenvolvida pode ser estendida através do desenvolvimento de novos compiladores para diferentes arquiteturas. Atualmente, apenas o compilador para documentos NCL está integrado ao sistema. Com novos compiladores implementados, o sistema ficará mais portátil, pois diferentes arquiteturas poderão ser importadas e editadas no sistema.

Muitas ferramentas para autoria de grafos exportam seus documentos no formato GXL - *Graph eXchange Language* (Winter et al., 2002) (Apêndice A). É interessante o desenvolvimento de um compilador que possibilite aos usuários do sistema HyperProp exportar e importar documentos no formato GXL. Com esse compilador integrado ao sistema, a interoperabilidade na edição em diferentes sistemas de edição de grafos torna-se possível.

A versão atual da visão estrutural do sistema apresenta alguns problemas com relação à exibição de vértices. Por exemplo, o tamanho do quadrado desenhado para uma composição expandida fica demasiadamente grande em alguns casos, causando transtornos para o usuário, que acaba tendo que redimensionar o quadrado manualmente. Além disso, a visão estrutural disponibiliza apenas o algoritmo de *Spring* (Kamada & Kawai, 1989) para desenho de grafos.

Os algoritmos de *Spring*, Hierárquico (Sugiyama et al., 1981) e Circular (Wills, 1997) já estão implementados na nova versão da visão estrutural. No entanto, alguns pontos permanecem pendentes e ficam como trabalhos futuros, tais como: ausência da técnica olho-de-peixe, a possibilidade de edição de vértices e arestas e expandir e colapsar vértices compostos.

Com relação à técnica de filtragem olho-de-peixe, um trabalho futuro é a sua extensão de forma a possibilitar a definição de dois vértices como foco na visão estrutural, facilitando assim a criação de relacionamentos. Nesse caso, uma alternativa seria a definição de uma função grau de interesse conjunta, composta pela soma algébrica de duas parcelas, onde cada uma é a aplicação da função grau de interesse considerando-se cada um dos focos separadamente.

A visão espacial para edição de leiaute pode ser aperfeiçoada acrescentando-se uma visão em árvore interna ao editor, semelhante à visão espacial do sistema GRiNS (Subseção 5.4.3). Com a árvore integrada à visão espacial, o usuário tem uma visão de hierarquia dos vértices de recursos do grafo. Muitas vezes, essa hierarquia não fica explicitada apenas através da janela em que o desenho do grafo é representado devido à sobreposição de vértices atômicos em vértices compostos.

No caso específico de documentos hipermídia, a implementação da árvore no editor espacial apresentado nesta dissertação deve mostrar, para cada região de apresentação do documento, o conjunto de folhas diretamente ligadas a ela, sendo cada folha um nó de mídia pertencente à região. Assim, o usuário teria uma visão imediata da relação entre nós de mídias e regiões de apresentação do documento.

Ainda com relação à visão espacial, outro trabalho futuro é a possibilidade de edição de relacionamentos espaço-temporais, pois a versão atual não disponibiliza tal funcionalidade.

A nova versão da visão temporal apresenta algumas pendências, tais como: ausência de representação de vértices compostos (apenas os vértices atômicos estão sendo representados na visão), inexistência de arestas relacionando os vértices do grafo, implementação dos formulários para edição de vértices e arestas e, por fim, a técnica de filtragem olho-de-peixe não se encontra implementada na visão.

Atualmente, o sistema apresentado nesta dissertação trabalha apenas com um documento de arquitetura por vez. Seria interessante a possibilidade de se trabalhar com diferentes documentos ao mesmo tempo no ambiente. Assim, o sistema disponibilizaria ao usuário um ambiente de comparações para diferentes arquiteturas através das visões oferecidas pelo sistema.