

## 5 Conclusões

Esta dissertação apresentou uma proposta de toolkit para agilizar o desenvolvimento de aplicações X3D. Foram abordados alguns pontos fracos detectados no X3D, especialmente na comparação direta com outras tecnologias 3D na web. Os aspectos abordados foram: colaboração, persistência, interação com outras aplicações e interação com o GUI web.

Para a colaboração foi desenvolvido um dispositivo de cliente / servidor que possibilita a troca de mensagens e sincronização da cena. Para persistência foi criado um *server control* que gerencia os cookies e os usuários, possibilitando a persistência da posição da câmera entre diversos acessos dos usuários. Ainda em persistência disponibiliza-se um mecanismo para persistência em disco da cena. Para a interação com outras aplicações reutiliza-se o mecanismo de troca de mensagens de forma que seja possível atribuir respostas programáticas a eventos da cena. Finalmente, para interação com o GUI web, foram utilizados métodos de programação Ajax do próprio .NET, que permitem acesso ao grafo da cena.

O desenvolvimento do toolkit foi motivado pela ideia de que ao facilitar a vida dos programadores, descomplicando o processo de desenvolvimento, pode-se acelerar a aceitação de uma tecnologia. Porém, não é possível afirmar que um toolkit como o DWeb3D vai mudar sozinho a realidade do X3D, e este trabalho não se propõe a isto. De fato seria interessante ter resposta de outros desenvolvedores que pudessem testar e avaliar a ferramenta, o que não foi possível dentro do escopo deste estudo. Outro ponto importante na avaliação do DWeb3D seria o desenvolvimento de uma aplicação mais complexa, de uso real.

Um ponto que poderia ser melhorado no futuro é adicionar ao DWeb3D a possibilidade de comunicação síncrona. Atualmente o toolkit só trabalha com comunicação assíncrona, o que obriga o cliente fazer pedidos de sincronismos periódicos, mesmo que ele não tenha alterado nada. Ainda no âmbito da comunicação talvez seja interessante um mecanismo encapsulado no toolkit que permita a transferência de objetos completos via rede. Hoje ele sincroniza propriedades de objetos, mas é necessário haver uma cópia dele no cliente

e no servidor. Para atendermos esta demanda com as funcionalidades hoje implementadas foi criado um processo onde um objeto gerente recebe uma mensagem que indica qual objeto ele deve criar. Isto funciona, mas pressupõe que o gerente conhece todos os tipos de objetos necessários.

Ainda como trabalho futuro, pretende-se desenvolver um site para publicar o toolkit. Se ele deve ser útil para a comunidade, ele deve estar disponível de forma gratuita na internet e preferencialmente ser OpenSource. Ainda dentro desse espírito deve-se desenvolver e publicar uma documentação de uso completa que permita a outros desenvolvedores entender o DWeb3D em detalhes. Também pretende-se estender a aplicação de demonstração criando um pequeno mundo virtual visitável que possa servir de exemplo para outros desenvolvedores.

Dentro desta linha de pesquisa seria interessante um estudo sobre as aplicabilidades de aplicações 3D interativas e seu potencial de impacto na forma como a interação ocorre na web. Outro estudo interessante seriam formas de codificar informação de cenas 3D e interações de forma a otimizar a experiência online que de acordo com todas as suas limitações ainda é bem única.

Dentre as lições aprendidas com este trabalho, notei o quão difícil é achar dados confiáveis no mundo que é a internet e estatísticas consisas de utilização de qualquer coisa na web são raras. Nota-se também o porquê de muitas decisões de arquitetura quando se tem de lidar com o mundo tão diversificado como os browsers, visualizadores e computadores clientes. É que alguns tipos de interação são complicados, como por exemplo coisas que precisem de um tempo de resposta muito curto, pois as diversas camadas de aplicação existentes criam uma sobrecarga considerável.

O tipo de abordagem utilizada tem limitações claras ao definir a necessidade do plugin da Unity3D, pois outros visualizadores não serão capazes de utilizar o DWeb3D. Esta decisão se deu para simplificar o processo e permitir trabalhar com um ambiente um pouco mais controlado, pois os recursos de um projeto como este são bastante limitados.

Outro aspecto a ser explorado é que o DWeb3D se propõe a instrumentar o X3D para fazer funções que muitos defendem não ser sua vocação. Estes defendem ter o X3D um aspecto muito mais voltado para cenas não multi-usuário, para visualização de produtos, definição de cenas com interações simples; outras ferramentas seriam naturalmente mais interativas como a Unity3D e outros motores gráficos que aparecem no mercado com opções de exportar para web. É fato que ferramentas mais novas vêm melhor preparadas para as demandas correntes do mercado, porém não se deve ignorar as

qualidades de um padrão bem estabelecido como X3D e nem subestimar o que pode ser alcançado com ele. Além disso o DWeb3D ao se utilizar da Unity3D, agrega ao X3D muitas das qualidades da Unity3D.

Uma forma interessante de verificar as qualidades do toolkit seria compará-lo a outros browsers. Tendo em vista essa comparação escolhi um dos browsers mais conhecidos no mercado atualmente, o BSContact. Trata-se de um browser X3D completo que implementa a SAI e é utilizado em várias aplicações existentes. Apesar dele possuir uma versão de um servidor multi-usuário, a programação ainda tem de ser feita basicamente utilizando ECMAScript e o controle sobre o servidor é limitado, ele basicamente sincroniza os diversos clientes sobre uma cena porém esta pode ser muito pouco modificada. A grande vantagem da abordagem do DWeb3D é exatamente permitir a modificação da cena 3D.