

3

Uma Arquitetura Distribuída via WEB

Neste capítulo será apresentada a Arquitetura de Ambiente Distribuído no qual está implementado o Gerador VRML (VRMLGer) e o fluxo de dados que há entre as diferentes camadas da Arquitetura.

3.1. Arquitetura do Ambiente Colaborativo

Apresenta-se aqui um *framework* de um ambiente colaborativo, com uma arquitetura flexível e extensível para a construção de uma família de sistemas, aplicadas a um determinado domínio.

Um *framework* é um sistema ou subsistema de software parcialmente completo, criado com o objetivo de ser instanciado. Define uma arquitetura para uma família de sistemas e fornece blocos básicos para a sua construção. Também define as partes que devem ser adaptadas para realizar uma funcionalidade específica. Consiste de *frozen spots* e *hot spots*. Os *frozen spots* definem a arquitetura global de um sistema de software, os seus componentes básicos e o relacionamento entre ele, e devem permanecer imutáveis em qualquer instanciação do *framework*. Os *hot spots* representam as partes do *framework* que são específicas para cada instanciação.

Frameworks são gerados de aplicações que estão relacionadas com uma determinada classe de problemas e soluções, ou seja, estão diretamente relacionadas com um domínio específico. Como *frameworks* são criados para gerar aplicações para um dado domínio, devem existir, portanto, os pontos de flexibilização (*hot spots*) que são adaptados de acordo com aplicações específicas, onde cada uma soluciona um problema particular. Em virtude destes *hot spots*, os *frameworks* não são sistemas, já que não são executáveis.

Para gerar um sistema, o *framework* deve ser instanciado, ou seja, deve ser implementado um código apropriado para uma determinada aplicação, dentro do seu domínio. Na maior parte dos casos, os *hot spots* são classes abstratas, isto é, não tem implementação e, ao serem instanciadas, exigem a

implementação dos métodos abstratos, fato que pode resultar em alguma adaptação nas classes.

Foi utilizado o conceito de *framework* para criar a arquitetura do ambiente colaborativo para desenvolvimento de projetos de engenharia de grande porte. A proposta deste *framework* é mostrada na figura 5 (Felicíssimo & Salles, 2002).

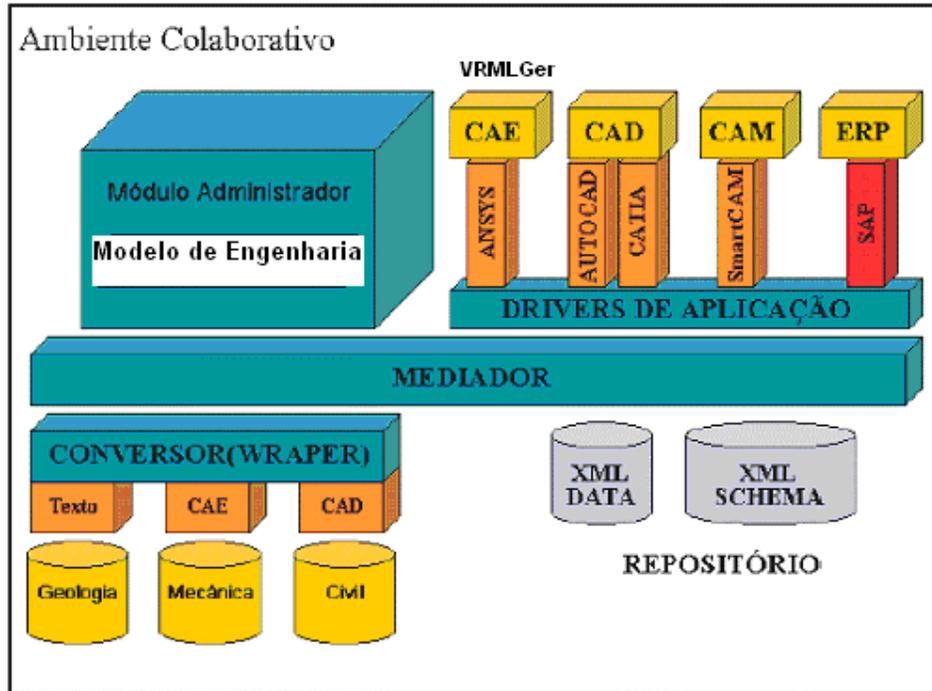


Figura 5 - Arquitetura do ambiente colaborativo.

Dentre as opções de integração de dados, a mais adequada para projetos de engenharia é um híbrido de mediadores e banco de dados. Utilizamos a flexibilidade de mediadores para integrar fontes com características heterogêneas e aliamos esta flexibilidade com o ganho de desempenho que é possibilitado através da materialização da visão integrada do mediador em um banco de dados. A seguir é apresentada uma explicação um pouco mais detalhada destes componentes.

A arquitetura é composta de um conversor para cada fonte de dados, um mediador para integração das informações, um repositório para materialização dos dados e registro de esquemas utilizados, um módulo administrador com informações dos modelos globais e um conjunto de drivers específicos para aplicações externas.

O módulo Administrador realiza a interface com os usuários, de forma a prover as seguintes funções: permitir gerenciar o projeto e sua(s) equipe(s) de desenvolvimento, controlar acesso para apenas pessoas autorizadas, solicitar a captura de esquemas e/ou de dados, permitir a formulação de consultas, trocar mensagens e guardar informações desejadas criando, assim, a memória do projeto, executar aplicações externas, visualizar seus resultados e, até fazer sua análise via ambiente WEB.

O usuário administrador pode também acionar o módulo Administrador para solicitar que um arquivo seja gerado para uma aplicação externa. Tal solicitação só pode ser efetuada se o respectivo driver tiver sido previamente desenvolvido e instanciado na arquitetura do ambiente. O módulo Administrador aciona o driver para que seja criado o arquivo. O driver por sua vez solicita os dados ao mediador e efetua a transformação necessária. Esta tarefa pode ser feita para qualquer esquema armazenado no repositório.

3.2. Localização do VRMLGer no ambiente

O VRMLGer é uma aplicação a mais na ilustração da figura 5 e poderia estar localizado no mesmo espaço físico que a aplicação CAE; mas para que o módulo Administrador tenha acesso aos arquivos VRML é necessária a criação do driver de aplicação que se comunique com o módulo Administrador. O driver deve também ser capaz de transmitir os arquivos VRML gerados para o usuário autorizado para a visualização.

Em uma tentativa de ver o Ambiente Distribuído com uma abstração para o usuário, representamos o ambiente como um sistema de 3 camadas, melhor representado na ilustração da figura 6.

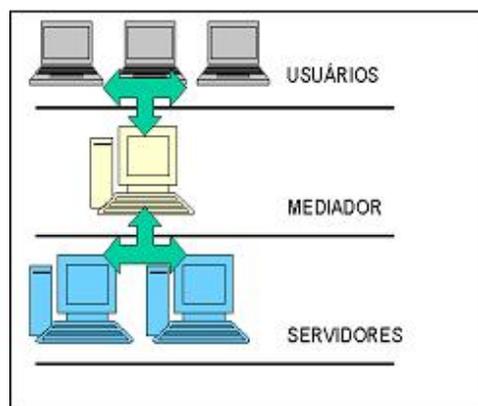


Figura 6 - Ambiente distribuído em 3 camadas.

Cada uma das três camadas tem função independente das demais e cada uma tem seu objetivo na arquitetura. A seguir será dada uma explicação das tarefas de cada camada.

A primeira camada, conhecida como camada de usuários, permite acessar, visualizar e/ou requisitar dados de forma distribuída. É nesta camada que o usuário (cliente) será capaz de interagir com o modelo tridimensional que foi gerado.

A segunda camada, conhecida como camada mediadora, é a camada encarregada de distribuir e direcionar as tarefas que são requisitadas pela camada superior e encaminhá-las para a camada seguinte a qual executará a tarefa propriamente dita, assim como identificar a qual usuário (cliente) pertence determinada tarefa.

A terceira camada, conhecida como camada de servidores, é a camada onde estão hospedadas as aplicações que irão executar as tarefas requisitadas pelo usuário (cliente).

3.3. Fluxo de dados no ambiente de 3 camadas

O fluxo de dados entre as diferentes camadas do ambiente distribuído deve ter uma interação transparente para o usuário.

Uma abstração do fluxo de dados entre as três camadas, sem entrar em detalhes, é ilustrada na figura 7. O usuário que se encontra na primeira camada deve ter a impressão de que está interagindo somente com um Servidor Web, que no caso seria a segunda camada (mediador), ficando para ele transparente a presença de um servidor de aplicações.

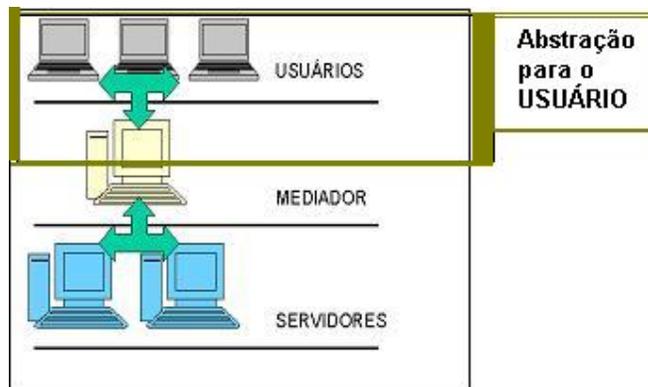


Figura 7 – Abstração do fluxo de dados em um ambiente de 3 camadas.

Na realidade o fluxo de dados é bem mais amplo do que aquele observado pelo usuário. Ilustra-se na figura 8 o fluxo de dados desde o pedido dos dados de visualização através de um browser e de um formulário até a volta dos arquivos gerados em formato VRML.

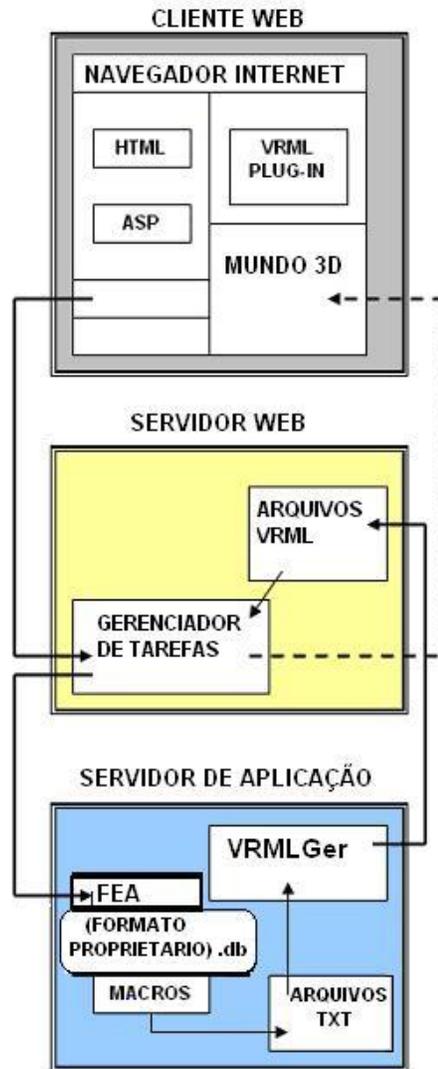


Figura 8 - Fluxo de dados entre um ambiente distribuído.

Conforme ilustra a figura 8 há um servidor de aplicações onde localiza-se o software de FEA, usualmente guardando informações em um formato proprietário. Neste mesmo servidor, há necessidade da criação de macros para obtenção dos arquivos texto que servirão como entrada para o VRMLGer.