**Capítulo 4**

**Trabalhos Relacionados**

Neste capítulo será apresentado trabalhos acadêmicos relacionados à sistemas distribuídos modularizados. Em sua descrição será abordada uma breve apresentação dos trabalhos junto com as características propostas neles. Além disso, será abordada na descrição uma breve comparação com o InteropFrame, destacando as principais diferenças e semelhanças.

4.1 HighFrame

Segundo {SantosSaulo2014} (HighFrame: Uma Ferramenta para o desenvolvimento em alto nível e deployment automático de sistemas distribuídos baseados em componentes), a HighFrame tem como objetivo simplificar o desenvolvimento de sistemas distribuídos baseados em componentes, permitindo ao desenvolvedor manter o foco no negócio do sistema. Ela propõe uma solução integrada que suporta ao desenvolvimento de componentes baseado em implementações genéricas focada no negócio do sistema, a definição da arquitetura do sistema baseado em um modelo gráfico de alto nível, como também ao *deployment* automático da arquitetura nos nós distribuídos disponibilizando o sistema na sua forma funcional.

A HighFrame foi projetada com o objetivo de manter o foco do desenvolvedor no negócio. Para isso foi criada uma camada de abstração para o desenvolvimento de componentes. Nela, anotações do modelo Fraclet representam preocupações do desenvolvimento baseado em componentes independentes da tecnologia de componentes. Dessa forma o código fonte com as anotações necessárias podem representar um componente e passa a ser inserido em uma base de componentes genéricos. Nessa base de componentes é centralizado todos os componentes genéricos em um servidor para que sejam disponibilizados em qualquer arquitetura independente de um dado modelo de componente específico, assim como mostra na figura 4.1.

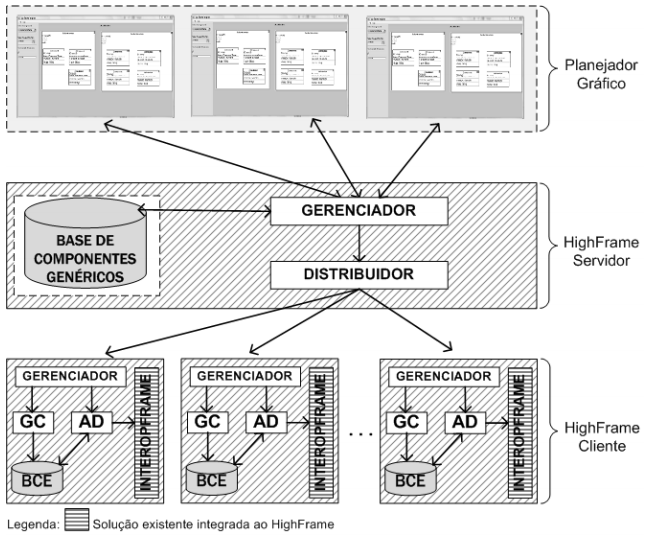


Figura 4.1 – Arquitetura da HighFrame. Fonte: {SantosSaulo2014}

Em sua arquitetura o InteropFrame, que é a ferramenta utilizada no trabalho proposto, é utilizada em uma parte da sua estrutura. Ele se localiza no módulo HighFrame Cliente e é utilizado para prover a interconexão automática entre componentes distribuídos possivelmente heterogêneos. Sendo assim, baseado no suporte aos modelos de componentes que o InteropFrame provê, o HighFrame também segue a mesma linha, pois ele se utiliza do InteropFrame para a interconexão desses componentes heterogêneos. Ou seja, o HighFrame se limita aos modelos de componentes que o InteropFrame provê. Entretanto o InteropFrame foi implementado de forma a ser extensível para novos modelos de componentes através de *templates*.

4.2 Vienna Component Framework

Segundo {OBERLEITNER2003} o VFC (Vienna Component Framework) suporta a interoperabilidade e modularidade dos componentes através de modelos de componentes diferentes. Dessa forma, o VFC apresenta um sistema unificado de modelo de componente onde a abordagem é baseada na construção de componentes de fachadas para cada modelo de componente. Baseado nessa criação das fachadas para cada modelo, o VFC consegue abstrair as diferenças entre os modelos suportados.

O VFC foi desenvolvido em JAVA com suporte a quatro modelos de componentes, assim como mostra a Figura 4.2: JavaBeans, EJB (Enterprise JavaBeans), CORBA (Common Object Request Broker Architecture) e COM (Microsoft’s Component Object Model). Esses modelos são suportados por meio de plugins. Logo novos modelos podem ser adicionados com a implementação de novos plugins. Para isso o VFC fornece um modelo de programaçMicrosoft'omponente espec programaçodelos podem ser adicionados com a implementaç e COM (Microsoft'omponente especão generico que possibilita a construção de novos plugins.

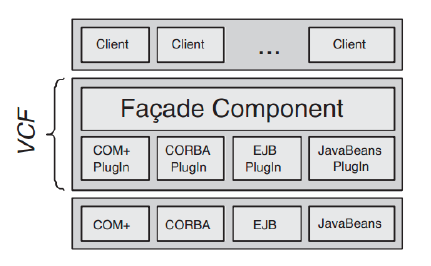


Figura 4.2 – Arquitetura do VFC {OBERLEITNER2003}.

Baseado no artigo de apresentação do VFC, nada em relação a comunicação entre componentes distribuídos foi apresentado. Ele apenas propôs uma solução que permite aos modelos de componentes diferentes se comunicarem de forma local. Diferentemente do InteropFrame que premite e realiza a comunicação desses componentes tanto de forma local como de forma distribuída. O VFC se assemelha muito ao InteropFrame em relação a sua arquitetura quando trata os modelos de componentes como plugins independentes. Uma vantagem grande pois permite ao software a extensibilidade.

4.3 Fraclet