

5 Conclusões

Neste trabalho é apresentada uma proposta de descrição arquitetural formal para os mecanismos de orquestração de recursos para provisão de QoS. A partir dessa descrição, feita através de estilos na ADL Wright (Allen, 1997), é possível utilizar ferramentas para inferir propriedades sobre os subsistemas descritos. Esse suporte formal possibilita a instanciação não-ambígua dos mecanismos de orquestração de recursos para suporte a aplicações multimídia.

Para simplificar o emprego da abordagem proposta, a DSL LindaQoS é introduzida. A linguagem é voltada especificamente para o domínio da provisão de QoS. Um compilador permite a tradução para uma descrição arquitetural equivalente em Wright, enquanto um outro compilador é deixado como trabalho futuro para permitir a tradução para o nível de implementação, gerando uma especialização dos frameworks genéricos para provisão de QoS (Gomes, 1999).

Enfim, a abordagem proposta possibilita que o usuário de QoS, através de uma única descrição concisa, obtenha uma base formal para a verificação de propriedades e para a implementação dos meta serviços de orquestração de recursos.

5.1. Contribuições esperadas

As principais contribuições do presente trabalho são:

- Levantamento de métodos para a descrição arquitetural voltados à provisão de QoS;
- Formalismo e refinamento dos Frameworks Genéricos para Provisão de QoS por meio de estilos arquiteturais na ADL Wright;
- A definição de uma linguagem de domínio específico para especificação dos meta serviços de orquestração de recursos;
- O compilador LindaQoS/Wright.

Levantamento de métodos para a descrição arquitetural voltados à provisão de QoS

O levantamento de ADLs para aplicação no contexto da provisão de QoS conduziu a um estudo profundo sobre diversas linguagens presentes na literatura. A ausência de suporte a composição de estilos em ADLs é uma deficiência significativa para expressar o aninhamento dos subsistemas de orquestração de recursos. É por esse motivo que a associação dos diversos subsistemas é feita através de um tipo de conector e não como uma composição de subsistemas. No mais, uma outra deficiência notada em ADLs é a impossibilidade de reuso de componentes em uma composição. Um mesmo componente interno não pode estar presente em dois componentes compostos distintos.

Formalismo e refinamento dos Frameworks Genéricos para Provisão de QoS por meio de estilos arquiteturais na ADL Wright

Este trabalho, baseado na terminologia apresentada em (Gomes, 1999), formaliza os frameworks para provisão de QoS. Apesar de serem descritos de forma precisa naquele trabalho, é apontada a necessidade de um maior formalismo para os padrões de estrutura e comportamento, especialmente dos mecanismos de orquestração de recursos. Por si só, esse formalismo permite um maior nível de refinamento da estrutura e semântica dos frameworks, já que é necessário expressar seu significado de forma não-ambígua.

A definição de uma linguagem de domínio específico para especificação dos meta serviços de orquestração de recursos

A DSL LindaQoS foi criada especificamente para o domínio da provisão de QoS. Através de notações e abstrações próprias do domínio da provisão de QoS, a linguagem permite a descrição concisa em alto nível dos mecanismos de orquestração de recursos.

O compilador LindaQoS/Wright

O compilador LindaQoS/Wright permite a utilização por parte dos projetistas de uma notação de alto nível, próxima do domínio do problema, como base de tradução automática para uma descrição arquitetural formal em Wright, usando os estilos desenvolvidos neste trabalho. Enfim, o código objeto resultante

pode ser usado em conjunto com as ferramentas de verificação formal de Wright para a checagem de propriedades.

5.2. Trabalhos futuros

Este trabalho oferece algumas possibilidades para trabalhos futuros:

- Implementação do compilador LindaQoS/JAVA;
- Extensão de ADLs para aumento de expressividade;
- Tradução de especificações em LindaQoS para estilos e não para configurações;
- Implementação de uma ferramenta gráfica para geração de código LindaQoS;
- Descrição dos mecanismos de parametrização de QoS;
- Descrição arquitetural formal do serviço principal (SOE – *Service Offering Environment*) de provisão de QoS;
- Inclusão de facilidades em LindaQoS para promover um maior reuso na descrição de subsistemas.

Implementação do compilador LindaQoS/JAVA

O desenvolvimento do compilador LindaQoS/JAVA possibilitará o emprego da mesma descrição alto nível em LindaQoS, já utilizada para a especificação em Wright, na geração de código objeto em JAVA (especializações dos frameworks genéricos para provisão de QoS em ambientes específicos). Os dois compiladores permitem o emprego da presente abordagem tanto no nível arquitetural quanto no de implementação. No mais, o conjunto das técnicas aqui utilizadas, que englobam os frameworks orientados a objetos, arquitetura de software, compiladores (geradores de código) e análise de domínio, servem para aumentar o reuso em todo o ciclo de desenvolvimento do sistema. Essa combinação de técnicas formam uma metodologia para engenharia de domínio que se chama DSSA (*Domain-Specific Software Architecture*) (Taylor et al., 1995).

Extensão de ADLs para aumento de expressividade

Neste trabalho, os subsistemas para orquestração de QoS são definidos individualmente com base em estilos arquiteturais e então ligados através de conectores especificados nos estilos *HierarchyNQoS* e *HierarchyTQoS*. Dessa forma, a composição aninhada de subsistemas não está sendo expressa em Wright, mas apenas a ligação entre eles, colocando todos os níveis de abstração em um mesmo plano (ex.: na Figura 4.1, o componente *Negotiator* do subsistema *QoSOS* deveria ser definido como um componente composto, contendo os subsistemas *CPU* e *Queue*, e não como estando ligado a eles). O aninhamento de subsistemas requer que um estilo *S* possa conter recursivamente componentes que seguem o mesmo estilo *S*, o que ainda não é suportado pelas ADLs atuais. Mesmo que isso fosse possível, como um estilo arquitetural não pode conter outro estilo, a descrição resultante impossibilitaria a descrição de arquiteturas, possibilitando apenas a descrição de configurações. Em outros termos, se as ADLs permitissem que um componente composto especificado em um estilo pudesse seguir esse mesmo estilo, ainda assim sua configuração interna precisaria ser definida. Uma sugestão de trabalho futuro, mais voltado para a área de arquitetura de sistemas, é a extensão de ADLs para oferecerem suporte a tais características, ou seja, permitir a composição de estilos e que tais composições pudessem conter recursivamente a si mesmas.

Adicionalmente, da mesma forma que certos conceitos de ADLs já foram levados para o contexto de autoria hipermídia (Muchaluat-Saade, 2003) com o intuito de aumentar o reuso e expressividade, o inverso poderia ser explorado. Uma proposta de trabalho futuro é o estudo de HALs (Hypermedia Authoring Languages) com a finalidade de oferecer algumas de suas funcionalidades no contexto da descrição arquitetural.

Tradução de especificações em LindaQoS para estilos e não para configurações

Conforme apresentado na Seção 4.4, especificações em LindaQoS são traduzidas como configurações e não como estilos arquiteturais em ADLs. Isso implica menos reuso e expressividade ao descrever famílias de sistemas. Uma proposta de trabalho futuro é a extensão de LindaQoS (ou apenas sua utilização como base) para a definição simplificada de estilos especificamente relacionados

aos mecanismos de orquestração de recursos. Um empecilho para essa tarefa, novamente, é a inexistência de ADLs que permitam composição de estilos. Com essa limitação, de nada adiantaria a definição dos estilos em LindaQoS, já que não seria possível traduzi-los para o nível arquitetural com a mesma expressividade.

Implementação de uma ferramenta gráfica para geração de código LindaQoS

Uma forma de facilitar ainda mais o uso de LindaQoS envolve a construção de uma ferramenta gráfica para geração de código. Nesse caso, os subsistemas de negociação e sintonização poderiam ser especificados graficamente (e mais rapidamente) pelo projetista, o que seria complementado por uma ferramenta para geração automática do código correspondente em LindaQoS.

Descrição dos mecanismos de parametrização de QoS

A utilização dos Frameworks de Parametrização de Serviço não é prática (consome tempo excessivo). É necessário haver uma forma diferente de representação das categorias de serviços, que seja mais declarativa e não-ambígua. Uma possível alternativa para essa questão seria a adoção de QML (Frolund & Koistinen, 1998) ou a extensão dessa linguagem para a definição precisa de categorias de serviços.

Descrição arquitetural formal do serviço principal (SOE – Service Offering Environment) de provisão de QoS

O presente trabalho se concentra na descrição arquitetural dos meta serviços de orquestração de recursos, que atuam sobre o serviço principal de provisão de QoS. Uma alternativa para trabalho futuro é prover também a descrição arquitetural do SOE, utilizando uma ADL com suporte a dinamismo, o que permitiria a criação e finalização de conectores associados a recursos virtuais que representam os *Media Pipes* (Colcher, 1999). Adicionalmente, se torna necessária a definição precisa de árvores de recursos virtuais (recursos virtuais e escalonadores) e da própria composição que gera as árvores de recursos virtuais não-primitivas.

Inclusão de facilidades em LindaQoS para promover um maior reuso de subsistemas

Especificações em LindaQoS ainda não promovem grande reuso. Certas facilidades, como suporte a subsistemas compostos e herança de subsistemas, são algumas opções que permitiriam um maior reuso. Além disso, a especificação dos componentes da linguagem como compostos aumentaria em muito a expressividade da linguagem. Mudanças sintáticas em LindaQoS devem ser guiadas (assim como em qualquer linguagem) pelo levantamento sistemático das necessidades de suas aplicações, o que requer sua utilização em vários contextos. É importante, também, ressaltar a necessidade de não se comprometer os requisitos estabelecidos para a linguagem, especialmente o de manter sua simplicidade.