

## 5 Conclusões

Nesta dissertação, foram estudadas algumas das principais características que dificultam a provisão de QoS em sistemas operacionais de propósito geral, de forma a relacioná-las com soluções e novas abordagens propostas por grupos de pesquisa da área. Essa análise ficou restrita aos subsistemas de rede e de escalonamento de processos pela importância que representam para as aplicações multimídia distribuídas, constatada pelo grande número de trabalhos relacionados. A partir da discussão sobre as propostas de provisão de QoS, foi observado que os mecanismos utilizados em cada uma delas recorrem a aspectos comuns de implementação, passíveis de serem modelados de forma genérica. Por isso, o presente trabalho propôs uma arquitetura adaptável para a provisão de QoS nos subsistemas de rede e de escalonamento de processos de sistemas operacionais, independente de implementação, por meio da utilização de frameworks.

O emprego de frameworks como ferramenta de modelagem facilitou a identificação dos pontos de flexibilização (*hot-spots*) que permitem a instanciação da arquitetura para a implementação de vários cenários reais. Como consequência da característica adaptável da arquitetura QoSOS, foram incluídos mecanismos de criação e modificação de serviços acionados durante a operação do sistema, funcionalidade desejável dada a evolução contínua das aplicações e de suas necessidades.

Os exemplos de aplicação da arquitetura foram apresentados considerando a implementação de um subsistema de escalonamento de processos hierarquizado, regido por um meta-algoritmo capaz de emular o comportamento de diversos algoritmos de escalonamento. Por fim, foi descrita a implementação de um cenário real de uso, no qual o sistema operacional Linux sofreu pequenas alterações para oferecer uma infra-estrutura adaptável de suporte ao modelo Intserv sobre os buffers de comunicação de estações finais e roteadores.

## 5.1 Contribuições da dissertação

As principais contribuições do presente trabalho são as seguintes:

- Definição de um modelo de funcionamento de sistemas operacionais, considerando os subsistemas de rede e de escalonamento de processos.
- Proposta preliminar de extensão do meta-algoritmo LDS para o escalonamento hierárquico de recursos;
- Definição de uma arquitetura genérica adaptável de provisão de QoS em sistemas operacionais;
- Definição de uma API genérica de solicitação de serviços com QoS a sistemas operacionais;
- Colaboração em projeto internacional de código aberto para a construção de uma API de configuração dos mecanismos de controle de tráfego do Linux.

### *Definição de um modelo de funcionamento de sistemas operacionais*

Concluindo a análise dos trabalhos relacionados, pôde-se construir um modelo simplificado do funcionamento de sistemas operacionais, considerando os subsistemas de rede e de escalonamento de processos. Tal modelagem foi capaz de ilustrar a dependência existente entre os subsistemas mencionados, e pôde servir como base para o modelo de orquestração de recursos da arquitetura definida.

### *Proposta preliminar de extensão do meta-algoritmo LDS*

O estudo do meta-algoritmo LDS foi motivado pela capacidade de adaptação que ele poderia oferecer se implementado para fins reais de escalonamento e não de análise, como definido originalmente. Baseado no conceito de escalonamento hierárquico, foi proposta uma extensão do LDS que, além de torná-lo apto a implementar diversas estratégias de escalonamento simultaneamente, permitiria a utilização de outros algoritmos quando da impossibilidade de emulação. Deve-se salientar que essa proposta foi feita de

forma preliminar e encontra-se pendente com relação a análises de desempenho e testes de verificação.

*Definição de uma arquitetura adaptável de provisão de QoS em sistemas operacionais*

A definição da arquitetura QoSOS é a principal contribuição do presente trabalho. Embora o cenário utilizado para exemplificar o uso da arquitetura tenha sido o sistema operacional Linux, ela é suficientemente genérica para ser aplicada a vários cenários de uso. Além disso, o conceito de *hot-spots* específicos de serviço torna possível a modelagem de sistemas operacionais altamente adaptáveis. O framework de adaptação de serviços, ausente no conjunto original de frameworks genéricos para provisão de QoS, utiliza-se de mecanismos já presentes em alguns sistemas operacionais para possibilitar a modificação de estruturas internas do sistema em tempo de execução. Ele permite, assim, que novos serviços sejam criados e que políticas de provisão de QoS sejam modificadas sem a necessidade de interrupção de serviços ou atualizações de hardware nas estações e roteadores de uma rede.

*Definição de API genérica de solicitação de serviços com QoS a sistemas operacionais*

Para a implementação do cenário de uso apresentado no Capítulo 4 foi desenvolvida uma API de solicitação de serviços independente de plataforma e das políticas de provisão de QoS empregadas pelo sistema operacional. Como ressaltado, procurou-se manter essa interface semelhante, o quanto possível, à API definida por (Mota, 2001), já que os dois cenários se complementam. O trabalho supracitado propôs a implementação de negociadores Intserv, sem, entretanto, concretizar os mecanismos de reserva de recursos e de controle de admissão sobre os buffers de comunicação, mecanismos esses definidos no presente trabalho.

*Colaboração em projeto internacional de código aberto*

A implementação do cenário de uso da arquitetura também levou à utilização de uma API de nível mais baixo para a configuração de controle de tráfego no sistema Linux. A TCAPI foi a única ferramenta encontrada com esse objetivo, criada por um projeto de código aberto internacional gerenciado pela

IBM. Essa interface precisou ser estendida quando se percebeu que não suportava algumas das funcionalidades necessárias ao cenário. As várias modificações feitas pelo autor do presente trabalho foram submetidas ao administrador do projeto e todas foram aprovadas e integradas ao software original<sup>22</sup>.

## 5.2 Trabalhos futuros

O presente trabalho abriu possibilidades de aprofundamento nos seguintes assuntos, passíveis de exploração em trabalhos futuros:

- Inclusão de outros recursos relevantes do sistema no modelo de orquestração;
- Análise e implementação da proposta de extensão do algoritmo LDS;
- Estudo de formas de verificação dos aspectos de segurança e da consistência nos mecanismos de adaptação;
- Exploração dos mecanismos de adaptação por outras funções de sistemas operacionais;
- Aperfeiçoamento do protótipo de implementação da arquitetura sobre o sistema operacional Linux;
- Estudo para portar o protótipo desenvolvido para utilização em set-top boxes de sistemas de TV interativa.

### *Inclusão de outros recursos relevantes do sistema no modelo de orquestração*

O estudo de outros subsistemas relevantes para a provisão de QoS em sistemas operacionais poderia trazer contribuições e refinamentos aos mecanismos definidos pela arquitetura. Se recursos como memória principal, sistema de paginação e acesso a disco forem incluídos no modelo de orquestração, o comportamento do sistema poderá ser predito com maior exatidão, oferecendo maior confiabilidade aos serviços instanciados a partir da arquitetura. Uma boa metodologia para fundamentar o modelo de orquestração seria descrever o

---

<sup>22</sup> Disponível em <http://www-124.ibm.com/developerworks/projects/tcapi/>.

relacionamento entre todos os recursos por meio de uma modelagem geral do funcionamento do sistema, como foi feito entre CPU e buffers de comunicação na Seção 2.3.

#### *Análise e implementação da proposta de extensão do algoritmo LDS*

A infra-estrutura de escalonamento de processos baseada no meta-algoritmo LDS hierárquico foi aqui abordada como uma proposta preliminar e utilizada para fins ilustrativos. Um trabalho que confirme a viabilidade de seu uso e promova sua implementação em um sistema operacional constituirá uma grande contribuição por disponibilizar uma porta de adaptação para estratégias de escalonamento de CPU, cujos componentes não necessitam ser codificados em linguagem de programação.

#### *Estudo de formas de verificação de segurança e consistência sobre a adaptação*

A aplicação de diretivas de segurança sobre certos mecanismos de provisão tem sido pouco explorada nos trabalhos sobre QoS. As solicitações de reservas de recursos devem ser disponibilizadas apenas a usuários autorizados e o gerenciamento de adaptabilidade do sistema deve ser cercado de cuidados. As formas de verificação dos aspectos de segurança levantados no Capítulo 3 devem ser estudadas de modo a solucionar questões do tipo: “Como verificar a confiabilidade de um componente?”; “Como assegurar a integridade do sistema frente à introdução de um componente?”; “Como verificar se um componente contém ações mal intencionadas?”; “Como monitorar as portas de adaptação para que não sejam usadas para outros fins, como invasão?”. Existe, ainda, a necessidade de verificação da consistência do sistema diante de algumas operações, como a remoção e a substituição de componentes. Uma análise detalhada sobre métodos capazes de manter uma base de informações sobre dependências entre componentes seria, portanto, de grande interesse.

#### *Exploração dos mecanismos de adaptação para outras funções do sistema operacional*

A aplicação do modelo de adaptabilidade em outras funções do sistema operacional é desejável, pois permitiria a configuração desses serviços por parte das aplicações ou de protocolos de reconfiguração em tempo de execução. Um

exemplo é a especificação de componentes que implementam tarefas específicas de uma pilha de protocolos adaptável, que é uma funcionalidade de baixo nível muito útil para ferramentas de programação de aplicações distribuídas, como a proposta por (Schmidt, 1994).

*Aperfeiçoamento do protótipo de implementação da arquitetura sobre o sistema operacional Linux*

A implementação de um protótipo completo e eficaz para a provisão de QoS em um GPOS específico vem a ser a validação prática de todas as estruturas apresentadas pela arquitetura proposta, principalmente o modelo de orquestração de recursos. O protótipo apresentado no Capítulo 4 pode ser estendido para que o código implementado no nível de usuário seja transportado para o *kernel* do sistema Linux. O próprio subsistema de escalonamento de pacotes (LinuxTC) pode ser aperfeiçoado para prover adaptabilidade na introdução de novas estratégias de escalonamento, utilizando também o algoritmo LDS estendido. Outro complemento do trabalho é a instanciação da arquitetura sobre o subsistema de escalonamento de processos, para o gerenciamento de QoS na CPU.

*Estudo para portar o protótipo desenvolvido para utilização em set-top boxes de sistemas de TV interativa*

Sistemas de TV interativa são aplicações multimídia distribuídas caracterizadas pelos requisitos já descritos no presente trabalho. Os terminais de recepção do sistema são chamados de set-top boxes (O'Driscoll, 2000), computadores de pequeno porte capazes de manipular os fluxos de áudio e vídeo para exibição e interação com o usuário. A escassez de recursos nesses terminais traz a necessidade de um estudo para a limitação das funcionalidades do sistema operacional com QoS àquelas estritamente necessárias. Além disso, os set-top boxes devem ser alvo de pesquisa para a identificação de outras necessidades de gerenciamento de QoS, por possuírem dispositivos de entrada e saída incomuns na maioria dos sistemas computacionais.