

## 6 Conclusão

Este trabalho apresentou um modelo de escalonador colaborativo utilizando corrotinas, projetado dentro de uma estrutura flexível com o objetivo de ser facilmente adaptável a diferentes cenários de computação orientada a eventos. Foi feito um estudo dos modelos existentes para escalonamento onde evidenciou-se as vantagens e desvantagens do uso de processos, *threads* e corrotinas.

O escalonamento colaborativo utilizando corrotinas atende às necessidades iniciais do projeto, pois fornece um modelo simples, leve e portátil, e que apresenta um bom desempenho quando aplicado na prática. O modelo proposto foi aplicado em um ORB desenvolvido em Lua, com o objetivo de permitir o ganho de escala nos serviços oferecidos através do processamento concorrente das requisições recebidas. Também foi analisada sua aplicabilidade dentro do cenário da computação para dispositivos celulares através da integração com a tecnologia BREW, onde observa-se um diferente conjunto de restrições de processamento e necessidade de uso eficiente dos recursos computacionais disponíveis.

Foram realizados experimentos utilizando o *middleware OiL*, que mostraram que o modelo proposto apresenta um desempenho muito próximo ao observado em ORBs comerciais mantendo o uso de recursos computacionais dentro dos limites esperados. No *LuaBREW*, o exemplo de uso mostrou que o desenvolvimento de serviços pode ser simplificado pelo uso do processamento colaborativo por permitir um fluxo seqüencial de programação, mesmo quando inserido em um ambiente exclusivamente orientado a eventos e que segue um modelo assíncrono para sua interface de programação.

### 6.1 Trabalhos Futuros

Em trabalhos futuros, propõe-se a realização de testes com o *OiL* utilizando outras arquiteturas de serviços, visando avaliar de maneira mais

apurada a escalabilidade do modelo proposto em diferentes cenários de uso. Um cenário interessante para essa avaliação seria a implementação de um serviço de eventos distribuídos[5] procurando atingir limites de escalabilidade não explorados neste trabalho.

Além disso, existe a necessidade de integração de APIs síncronas como possíveis fontes de eventos para o *Escalonador*. Essa integração permitiria por exemplo a integração de um console de comandos para manipulação de servidores *OiL*. Uma outra frente de trabalho identificada é a utilização do *Escalonador* para integrar conexões provenientes de outros *middlewares* como por exemplo o *ALua*[35].

No campo dos dispositivos móveis celulares existe a motivação para uma integração de APIs mais amigáveis para o usuário, permitindo que o fluxo seqüencial de programação seja utilizado para criar ferramentas que simplifiquem tarefas de desenvolvimento mais complexas como por exemplo *frameworks* para o desenvolvimento de jogos.

Frente aos resultados de utilização de memória e processamento obtidos nos testes do *OiL*, pode-se investir também na adaptação desse *middleware* para o ambiente *LuaBREW* com o objetivo de permitir sua utilização para o desenvolvimento de sistemas distribuídos para dispositivos móveis.