3
Árvores de Recursos Virtuais (VRT): Um meta modelo para MDRM

A especificação de um meta modelo para MDRM constitui, sem dúvida, ponto-chave da abordagem, uma vez que muitos de seus aspectos de projeto devem ser nesse momento considerados. Especialmente, o meta modelo deve descrever de que forma os recursos serão representados para oferecer a uniformização de gerenciamento necessária e como abstrações de alto nível podem ser definidas para atender aos diferentes atores e infra-estruturas de QoS.

O meta modelo de MDRM é denominado "Árvores de Recursos Virtuais" (VRT) (Moreno, 2006) e é fruto de uma reengenharia da proposta de Gomes (2001), no âmbito de seus frameworks de alocação e compartilhamento de recursos. Originalmente, o poder de recorrência que garantia aos demais frameworks a capacidade de representar os mecanismos de QoS em diversos níveis de abstração não se refletia em um subsistema de gerenciamento de recursos fim-a-fim. As árvores de recursos virtuais se limitavam a modelar o gerenciamento de recursos reais apenas, e a tratá-los individualmente.

Neste capítulo, o meta modelo VRT é apresentado verbalmente, auxiliado por representações gráficas estilizadas. A notação pictórica utilizada visa facilitar o entendimento dos conceitos e, caso fosse um pouco enriquecida, poderia se estabelecer como linguagem de domínio específico gráfica para MDRM. No entanto, o presente trabalho se limitou a especificar VRT formalmente apenas por meio da descrição da DSL Pan, realizada no próximo capítulo.

3.1.
O conceito básico de Recurso Virtual

O conceito básico de *recurso virtual* é definido inicialmente como a representação abstrata da capacidade total de um *recurso real*. O significado exato de "capacidade" varia de acordo com o recurso. Por exemplo, para o recurso "CPU", trata-se da capacidade de processamento, que pode ser expressa em

PUC-Rio - Certificação Digital Nº 0321123/CA

termos de MIPS, MFLOPS, SPECInts, etc. Para o recurso "canal de comunicação", trata-se da capacidade de encaminhamento de dados, expressa em termos da vazão de dados, em bits por segundo.

Como a denominação indica, um recurso virtual possui as mesmas funcionalidades do recurso real abstraído. Sua capacidade deve, da mesma forma, ser compartilhada entre os fluxos[2] que o utilizam. Esse compartilhamento tem sua lógica definida por meio de uma *estratégia de escalonamento*. A estratégia de escalonamento determina quando cada fluxo terá um bloco de suas unidades de informação manipulados pelo recurso virtual, o que se traduz no tipo de garantia de QoS que os fluxos podem obter no uso do recurso virtual, ou ainda, no tipo de serviço oferecido. Por exemplo, para uma CPU virtual, uma estratégia de escalonamento que descreve o algoritmo *Rate Monotonic* (Liu, 1973) determina garantias quanto à periodicidade de escalonamento de processos que poderiam ser oferecidas; e possibilitaria um tipo de serviço apropriado a aplicações de tempo real severo. Por outro lado, há estratégias de escalonamento que não provêem quaisquer garantias, configurando um serviço de melhor esforço, que, no caso de uma CPU virtual, pode ser exemplificado pelo uso do algoritmo Round Robin.

Vários tipos de estratégias de escalonamento dependem de um certo controle que selecione quais fluxos podem ser atendidos, de forma que as garantias de QoS esperadas possam realmente ser oferecidas. Um fluxo somente pode ser aceito para manipulação por um recurso virtual se suas características de desempenho puderem ser garantidas pela estratégia de escalonamento do recurso virtual, sem prejuízo às garantias já concedidas aos fluxos aceitos. A lógica de verificação dessa viabilidade de aceitação é descrita em separado da estratégia de escalonamento, por meio de uma *estratégia de admissão*. Normalmente, uma estratégia de admissão recolhe as informações sobre alocações já efetivadas para inferir qual a capacidade do recurso ainda está disponível. Com essa informação, os parâmetros que caracterizam as necessidades de desempenho do novo fluxo podem ser avaliados, determinando se ainda há possibilidade do novo atendimento com as garantias esperadas.

[2] O conceito de fluxo neste contexto abrange qualquer seqüência de unidades de informação a ser manipulada por um recurso, como os fluxos de pacotes manipulados por um canal de comunicação ou os fluxos de instruções manipulados por processadores.

PUC-Rio - Certificação Digital Nº 0321123/CA

A estratégia de admissão é descrita em separado da estratégia de escalonamento para que uma maior variedade de serviços possa ser oferecida. Por exemplo, para um canal virtual de comunicação, pode-se conjugar um escalonamento de melhor esforço a um controle de admissão baseado nas taxas médias dos fluxos admitidos, oferecendo, assim, um serviço de carga controlada (Braden, 1994). Em breve, ficará claro que essa separação de interesses vem também a beneficiar a modularidade de especificação de um recurso virtual, permitir o reuso e adaptação de estratégias de gerenciamento e formar os subdomínios de MDRM.

Um recurso virtual possui associado, ainda, um conjunto de *regras de controle de acesso* que restringem as ações de uso e de gerenciamento a certas condições, formuladas em torno dos usuários/serviços solicitantes, características dos fluxos, horários de solicitação, entre outros. Por exemplo, uma regra de controle de acesso pode definir que "apenas fluxos originados da rede X podem ser admitidos", ou que "um usuário A somente pode realizar tarefas de gerenciamento entre 08 e 18h". Assim, toda operação sobre um recurso virtual deve ser submetida a um controle de acesso *a priori*, para verificação de permissões. Mais uma vez, deve haver uma separação de interesses, para o estabelecimento das regras de controle de acesso, que poderiam formar mais um subdomínio de MDRM. Juntas, estratégias de escalonamento, estratégias de admissão e regras de controle de acesso formam o conjunto de estratégias de gerenciamento sobre um recurso, conforme mencionado anteriormente.

A Figura 5 ilustra as associações entre recurso virtual e suas estratégias de gerenciamento. Em resumo, um recurso virtual é a abstração de um recurso real qualquer, criada com o intuito de tornar uniformes os conceitos preliminares relacionados ao gerenciamento de recursos com suporte a QoS.

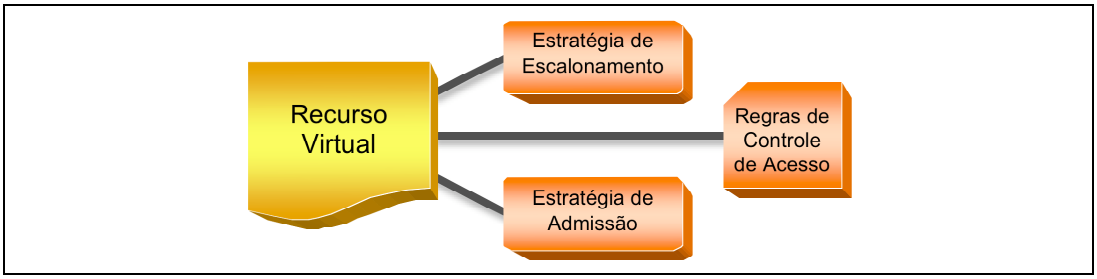

Figura 5 – Recurso Virtual e suas estratégias de gerenciamento

PUC-Rio - Certificação Digital Nº 0321123/CA

Deve-se atentar para o fato de que o tempo não é a única dimensão de escalonamento de recursos computacionais. Por exemplo, o recurso "memória" deve ser escalonado no espaço, para determinar onde unidades de informação serão armazenadas, e sua capacidade representa o espaço de armazenamento disponível, expresso em bits. Há, ainda, recursos que são compartilhados tanto no tempo quanto no espaço. É o caso de dispositivos de memória secundária, que possuem uma fila de requisições de acesso a ser escalonada no tempo e uma área de armazenamento a ser compartilhada no espaço.

Outro detalhe digno de nota é o gerenciamento de um agregado de recursos, como CPUs em multiprocessamento simétrico (SMP). Trata-se da reunião de recursos de mesma natureza e finalidade para oferecer uma soma de capacidades, possível devido ao uso em paralelo dos recursos participantes. Acompanhando a forma de gerenciamento comumente utilizada para esse tipo de recurso, a abstração de um agregado de recursos pode ser feita por meio de um único recurso virtual. Cabe à estratégia de escalonamento considerar a existência de múltiplos recursos reais para possibilitar que a capacidade do recurso virtual seja vista como a soma dos recursos agregados.

PUC-Rio - Certificação Digital Nº 0321123/CA

3.2.
O conceito básico de Árvores de Recursos Virtuais

O conceito de recurso virtual pode agora ser estendido para representar não a totalidade de um recurso real, mas uma porção deste. Assim, o conceito básico descrito na seção anterior passa a ser um caso especial de recurso virtual, no qual a porção que ele representa é a capacidade total de um recurso real. A intenção por trás dessa extensão é representar, também como recursos virtuais, as alocações concedidas como garantias aos fluxos admitidos. Pode-se afirmar, assim, que o fluxo admitido tem à sua disposição um recurso virtual que a ele dedica toda sua capacidade. Ou seja, cada reserva de recursos reais estabelecida com sucesso gera uma abstração de recurso virtual.

Um recurso virtual pode representar, também, uma parcela de um outro recurso virtual já abstraído a partir do recurso real, formando uma hierarquia de recursos virtuais. Essa hierarquia não denota apenas a concessão de alocações de um recurso real em cascata, mas também a delegação do gerenciamento de cada

parcela alocada, uma vez que cada recurso virtual pode possuir suas próprias estratégias de gerenciamento.

A hierarquia de recursos virtuais é denominada *árvore de recursos virtuais* (VRT – *Virtual Resource Tree*), conceito que dá nome ao meta modelo. Reiterando, uma VRT denota a divisão hierárquica da capacidade de um recurso ou de um conjunto de recursos gerenciados em grupo. A cada divisão, o controle do compartilhamento da capacidade alocada é delegada a estratégias de gerenciamento associadas aos recursos virtuais filhos.

O *recurso virtual raiz* de uma VRT representa a capacidade total do recurso real ou de um grupo de recursos reais. Da raiz da árvore até seus nós folhas, a capacidade do recurso é dividida hierarquicamente e finalmente alocada para fluxos. A definição do que é o fluxo de interesse determina onde a divisão da árvore termina. No compartilhamento de uma CPU, por exemplo, pode-se considerar como fluxo um conjunto de processos, assim um recurso virtual folha compartilharia sua capacidade entre esses processos. Pode-se também considerar como fluxo a seqüência de instruções de máquina definida por um único processo, assim um recurso virtual folha estaria dedicado a esse processo.

Um *recurso virtual folha* representa uma garantia concedida pelo seu recurso virtual pai, sem que outras garantias sejam dali em diante concedidas. Assim, recursos virtuais folhas não possuem uma estratégia de admissão associada, uma vez que filhos não serão criados. Além disso, a estratégia de escalonamento normalmente associada a recursos virtuais folhas é simples o bastante para tratar as unidades de informação de um fluxo sem qualquer distinção entre elas. Exemplos de estratégias de escalonamento adequadas para recursos virtuais folhas são as estratégias FIFO para pacotes de rede e Round Robin para processos.

Um *recurso virtual intermediário* representa uma garantia concedida pelo recurso virtual pai, sendo que a partir dessa garantia, outras podem ser concedidas a recursos virtuais filhos. Com essa definição, pode-se afirmar que o conceito de árvore de recursos virtuais preenche a necessidade de oferecimento de diferentes estratégias de gerenciamento sobre um mesmo recurso. Cada um dos recursos virtuais intermediários, com suas próprias estratégias de gerenciamento, define um tipo diferente de serviços oferecidos.

PUC-Rio - Certificação Digital Nº 0321123/CA

A solicitação de uma garantia é especificada por meio de parâmetros que variam de acordo com o tipo de serviço, uma vez que compreendem os parâmetros de QoS e de caracterização do tráfego necessários para a avaliação de viabilidade por parte de uma estratégia de admissão e para a configuração da alocação junto à estratégia de escalonamento. A descrição de um tipo de serviço e de seu conjunto de parâmetros recebem o nome de categoria de serviço. Assim, cada recurso virtual não-folha de uma VRT disponibiliza aos recursos virtuais filhos uma categoria de serviço própria.

A Figura 6 ilustra uma árvore de recursos virtuais, destacando os elementos associados a recursos virtuais folhas e recursos virtuais não-folhas, ou seja, raiz e intermediários.

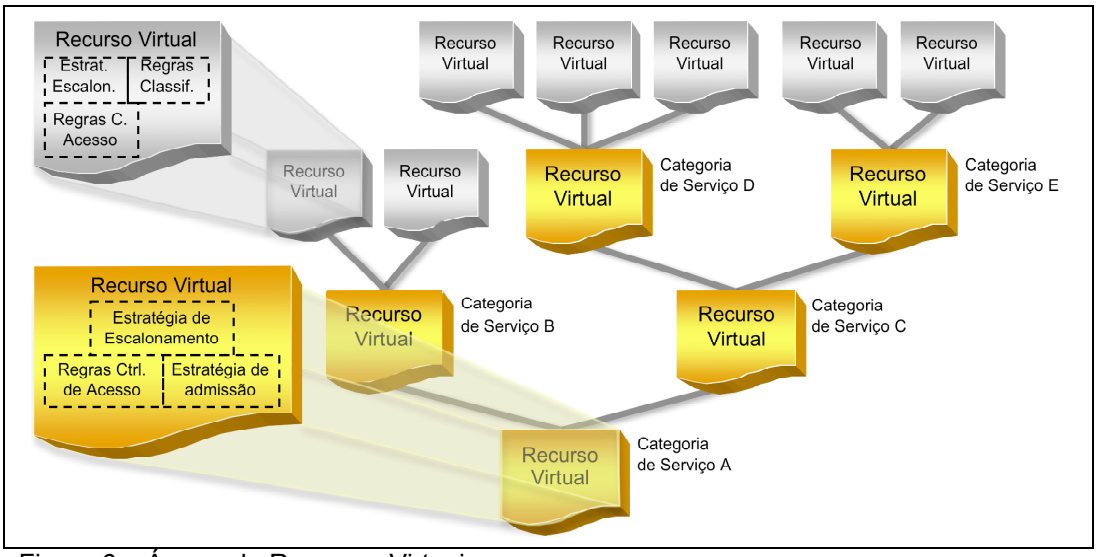

Figura 6 – Árvore de Recursos Virtuais

Quando unidades de informação de um fluxo chegam à raiz de uma VRT, devem ser analisadas para a identificação do fluxo a que pertencem, e, assim, de qual recurso virtual folha deve tratá-las. Isso é tarefa para o *classificador associado à VRT*, que se baseia em regras de classificação configuradas nos recursos virtuais folhas. Cada regra especifica um padrão de casamento a ser testado contra o conteúdo das unidades de informação, no intuito de prover características para a identificação do fluxo. O recurso virtual folha que tiver uma regra casada com sucesso é o destino das unidades de informação testadas.

Nota-se que essa descrição sobre classificação se encaixa muito bem a recursos de natureza assíncrona, em que o momento ou a ordem da chegada de unidades de informação não podem ser previstos. Em outros tipos de recursos, de

PUC-Rio - Certificação Digital Nº 0321123/CA

natureza síncrona, a classificação de fluxos é um pouco diferente. Por exemplo, em uma VRT que gerencia uma CPU, a classificação é feita uma única vez, quando da criação do processo ou de sua associação ao recurso virtual folha.

Uma vez classificadas e enfileiradas nos recursos virtuais folhas, as unidades de informação dos fluxos aguardam que o escalonamento hierárquico atribua a elas o direito de manipulação pelo recurso real. O escalonamento hierárquico se inicia quando o *escalonador associado à VRT* consulta a estratégia de escalonamento da raiz, para obter o recurso virtual filho cuja reserva deve ser atendida naquele momento. A estratégia de escalonamento desse recurso virtual é, então, a próxima a ser consultada, e assim sucessivamente até um recurso virtual folha, que realmente escalonará as unidades de informação do fluxo.

Toda tarefa de gerenciamento que modifique a estrutura de uma árvore deve ser condicionada à liberação por parte de um *controlador de acesso associado à VRT*. O controlador de acesso consulta as regras relativas à tarefa solicitada que encontram-se configuradas no recurso virtual alvo da modificação. Um exemplo específico de tarefa de gerenciamento é a criação de um recurso virtual, como resultado de uma admissão. A criação de um recurso virtual, além de ter de ser liberada pelo controlador de acesso, deve ser avaliada quanto à sua viabilidade por um *controlador de admissão associado à VRT*. O controlador de admissão consulta a estratégia de admissão associada ao recurso virtual candidato a ser pai. Se a resposta for positiva, o novo recurso virtual pode ser criado.

Concluindo, o conceito de árvore de recursos virtuais descrito nesta seção tem como características principais a uniformização do gerenciamento de recursos reais com QoS, a modularidade de especificação dos aspectos relacionados ao gerenciamento e o suporte à especificação de diferentes tipos de serviços sobre um mesmo recurso. Além disso, o conceito abre caminho para implementações de suporte em plataforma que permitam adaptações de serviços, a partir da modificação em tempo de operação da estrutura de uma VRT instanciada e/ou de suas estratégias de gerenciamento. Nas próximas seções, o conceito de VRT será refinado sucessivamente para atender a diversos outros requisitos.

PUC-Rio - Certificação Digital Nº 0321123/CA

3.3.
Árvore de Recursos Virtuais Primitiva

O refinamento do conceito de árvores de recursos virtuais se inicia pela definição de que toda VRT associada a um recurso real denomina-se *árvore de recursos virtuais primitiva*. Ou seja, todas as árvores definidas sob os princípios descritos na seção anterior são VRTs primitivas. É importante relembrar que o recurso real gerenciado por uma VRT pode ser singular, como um canal de comunicação, ou agregado, como multiprocessadores.

O recurso virtual raiz de uma VRT primitiva pode, alternativamente, estar associado a um outro recurso virtual, intermediário em uma outra VRT primitiva, que pode-se afirmar ter sido "separado" da VRT original para maior facilidade ou transparência no gerenciamento. Por exemplo, uma parcela de CPU pode se tornar a raiz de uma VRT dedicada a um usuário específico do sistema operacional. A mesma abstração pode também ser interessante a ferramentas de virtualização (Chisnall, 2007), que tornariam transparentes aos sistemas operacionais convidados (*guest*) toda a alocação de reservas da CPU feita pelo sistema operacional hospedeiro. Conceitualmente, a nova VRT que tem como raiz um recurso virtual de uma outra VRT primitiva nada mais é que uma sub-árvore dessa VRT de origem.

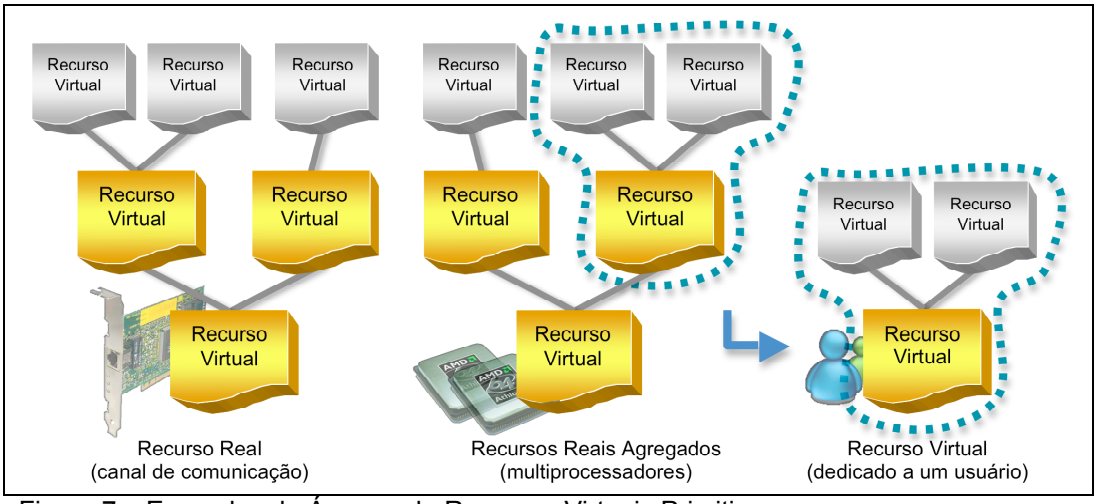

Figura 7 – Exemplos de Árvores de Recursos Virtuais Primitivas

Portanto, são três os casos em que uma VRT é dita ser primitiva, quando sua raiz está associada: a um único recurso real; a um agregado de recursos reais; ou a um recurso virtual intermediário de outra VRT primitiva. Em suma, elas são

PUC-Rio - Certificação Digital Nº 0321123/CA

denominadas primitivas por terem suas estratégias de gerenciamento atuando diretamente sobre recursos reais.

3.4. Floresta de Recursos Virtuais

Um sistema operacional, por definição, é um tipo de plataforma que poderia suportar a instanciação de VRTs primitivas para ter seus recursos reais disponibilizados para modelagem. Para cada recurso real de interesse, o sistema operacional ofereceria a possibilidade de inicialização de uma VRT, a ser provida por um ator autorizado a fazê-lo. O conjunto de VRTs primitivas controladas por um sistema operacional representa um domínio de administração em comum, atribuído ao administrador da estação, bem como um nível de abstração em comum, o dos recursos reais.

O conceito de *floresta de recursos virtuais* (VRF - *Virtual Resource Forest*) denota exatamente um conjunto de árvores de recursos virtuais que podem ser reunidas por terem alguma característica em comum, como localização, autoridade administrativa, nível de abstração etc. Não há limitações para a definição de quais VRTs podem ou não ser incluídas em uma VRF, mas o conceito é formulado para trazer facilidades para a organização de VRTs, que estão instanciadas por todo um sistema distribuído. Por isso, quanto maior a coerência na criação de florestas, mais simplificadas podem ser as atividades de manutenção dos modelos posteriormente.

O conceito de floresta é hierárquico, ou seja, florestas podem conter VRTs e outras florestas. Este é o caso de um aglomerado de alto desempenho (*cluster*), cuja floresta é formada por diversas florestas de sistemas operacionais, uma para cada nó. Por ser um conceito hierárquico e de livre modelagem, é permitido que mais de uma floresta contenha uma mesma VRT ao mesmo tempo: pelo caminho da hierarquia de florestas, obviamente, uma VRT integrante de uma certa floresta *F1* é também integrante de uma floresta *F2* que contenha *F1*. Mas há outros casos, como uma VRT associada a um canal de comunicação de um roteador que participa, ao mesmo tempo, de uma floresta IntServ (Braden, 1994) e de uma floresta DiffServ (Blake, 1998).

PUC-Rio - Certificação Digital Nº 0321123/CA

Quando um serviço é solicitado a uma floresta, componentes específicos da infra-estrutura de QoS verificam as permissões e a viabilidade de sua admissão, levando em conta a utilização atual dos recursos representados pela floresta e a demanda da solicitação. Esses componentes também são chamados, respectivamente, controlador de acesso e controlador de admissão, porém atuam em um nível de abstração mais alto. Eles também se baseiam em regras de controle de acesso e em uma estratégia de admissão, agora associados à floresta.

Se autorizado pelo controlador de acesso, o controlador de admissão da floresta inicia seu trabalho chamando um de seus componentes de negociação filhos. O negociador chamado inicia um processo denominado orquestração de recursos, identificando as VRTs e florestas que compõem o subconjunto de recursos envolvidos na provisão do serviço. A orquestração continua associando parcelas de responsabilidade de provisão de QoS para cada VRT ou floresta escolhida e ativando componentes de mapeamento que traduzem a categoria de serviço solicitada (e seus parâmetros de QoS) para categorias diretamente relacionadas a cada VRT ou floresta.

Após o mapeamento, os controladores de acesso e de admissão associados a cada VRT e floresta escolhidas são chamados. No caso de uma floresta, seu controlador de admissão realiza novamente os passos descritos acima, recorrentemente, até que controladores de admissão de VRTs sejam atingidos. Se todos os controladores de admissão de uma floresta retornarem respostas afirmativas, uma resposta afirmativa é passada para o controlador de admissão que solicitou os serviços da floresta, e por isso dito de nível mais alto. Esse procedimento se repete até que o controlador de admissão que iniciou todo o processo seja atingido. Em seguida, todas as reservas de recursos aprovadas durante a fase de controle de admissão são confirmadas, ou seja, os recursos virtuais folhas são finalmente criados em todas as VRTs primitivas envolvidas. Por outro lado, basta uma única resposta negativa para que o processo seja abortado e o controlador de admissão da floresta negue o pedido.

3.5.Árvore de Recursos Virtuais Composta

Conforme discutido anteriormente, há a necessidade de se disponibilizar abstrações no sistema de gerenciamento de recursos que permitam o acesso a toda

PUC-Rio - Certificação Digital Nº 0321123/CA

uma composição de recursos fim-a-fim. Além disso, as mesmas abstrações podem servir para tornar transparente aos atores de interesse toda a complexidade de gerenciamento de recursos reais distribuídos.

Para se adequar a esse cenário, o meta modelo VRT estende novamente o conceito de recurso virtual, dessa vez para simbolizar uma composição formada por outros recursos virtuais. Um *recurso virtual composto*, como é denominado, é a representação em um certo nível de abstração do conjunto de recursos virtuais envolvidos em uma orquestração de recursos. A VRT que possui como raiz um recurso virtual composto é denominada *árvore de recursos virtuais composta*. VRTs compostas possibilitam, assim, que todo um caminho de recursos seja representado em algum nível de abstração, de forma que sua gerência possa ser vista como se atuasse sobre um único recurso.

Na orquestração de recursos descrita na seção anterior, bastaria que uma raiz de VRT composta fosse criada para cada floresta que participou do processo, representando a composição dos recursos que foram dali alocados. A raiz da VRT composta criada pelo controlador de admissão de mais alto nível pode dividir sua capacidade criando recursos virtuais filhos, dos quais cada um, por sua vez, pode dividir sua própria capacidade, e assim por diante, como em uma VRT primitiva. Porém, todas as outras raízes de VRTs compostas criadas nesse processo de orquestração não podem ser compartilhadas, já que são recursos virtuais completamente alocados a outras VRTs compostas. Conseqüentemente, VRTs compostas intermediárias sempre possuirão apenas um nó, raiz.

Da mesma forma, recursos virtuais de VRTs primitivas participantes de uma composição devem obrigatoriamente ser folhas, pois estão dedicados ao recurso virtual composto que integram.

Outro fato digno de nota está na lógica de compartilhamento de VRTs compostas, que, na maioria das vezes, não precisará ser descrita por uma estratégia de escalonamento. Isso porque em níveis de abstração mais altos, normalmente não existe a noção de escalonamento de um recurso, abstrato, mas apenas os controles de acesso e de admissão.

A Figura 8 ilustra um exemplo simplificado de orquestração de recursos, em um sistema operacional. A VRT composta (①) representa uma orquestração entre duas VRTs primitivas associadas a um processador (②) e a um canal de comunicação (③). Cada recurso é gerenciado por suas respectivas VRTs

PUC-Rio - Certificação Digital Nº 0321123/CA

primitivas, as quais compõem a floresta do sistema operacional. As linhas tracejadas representam a orquestração que originou a raiz da VRT composta.

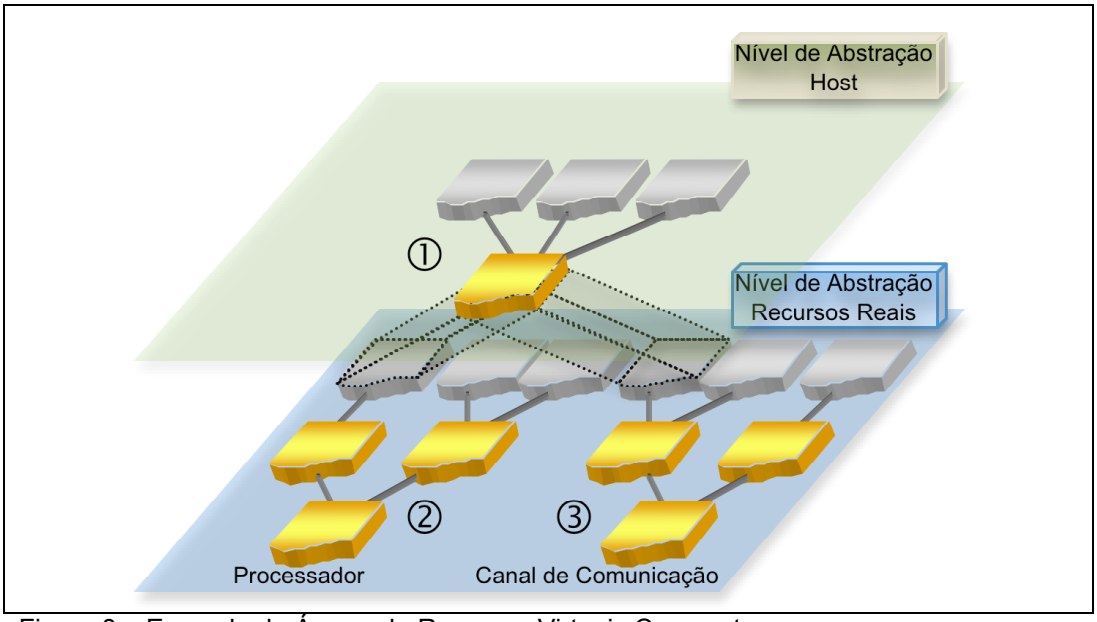

Figura 8 – Exemplo de Árvore de Recursos Virtuais Composta

Outro exemplo, apresentado pela Figura 9, ilustra como o processo de orquestração realizado por uma infra-estrutura de QoS pode coordenar suas ações junto ao subsistema de gerenciamento de recursos fim-a-fim, baseado no meta modelo VRT. Um usuário de um sistema de vídeo sob demanda solicita à aplicação suas necessidades de QoS em termos da qualidade desejada da mídia, como na expressão "Vídeo em alta definição (HDTV)" (passo ① da Figura 9)[3]. O módulo da aplicação que trata essa requisição é o controlador de admissão de mais alto nível, uma vez que ele iniciará todo o processo recorrente de negociação. O pedido é repassado a um outro módulo da aplicação, o negociador de QoS, que deve descobrir quais florestas e árvores participarão do serviço e dividir a responsabilidade de QoS entre elas. No caso, o negociador da aplicação está preparado para dividir a solicitação entre três florestas: a do sistema operacional cliente, a do sistema operacional servidor e a da rede fim-a-fim. O negociador aciona previamente mecanismos de mapeamento que traduzem a requisição do usuário para parâmetros adequados a cada floresta (②). Aí sim, os resultados do mapeamento são repassados aos respectivos controladores de admissão (③). O

[3] Nesta descrição não são citadas as intervenções de controladores de acesso, por simplificação. Mas cada chamada a um controlador de admissão deve ser antes avaliada por um controlador de acesso.

PUC-Rio - Certificação Digital Nº 0321123/CA

processo continua de forma análoga dentro de cada floresta. No caso das florestas dos sistemas operacionais, a requisição é repassada ao controlador de admissão do sistema operacional, que aciona seu negociador para distribuir a responsabilidade de QoS entre os recursos que participarão do serviço. O negociador de QoS do sistema operacional mapeia os parâmetros recebidos para parâmetros adequados aos recursos escolhidos (④), e então os repassa aos controladores de admissão de cada VRT primitiva (⑤). Os mesmos procedimentos são realizados na floresta da rede fim-a-fim, porém mais passos de recorrência serão necessários, uma vez que haverá a orquestração dos nós que participarão, e, ainda, poderão estar envolvidos diversos provedores de serviços, diferentes tipos de rede, etc (⑥). A negociação na rede termina quando as VRTs primitivas associadas aos canais de comunicação e mantidas pelos sistemas operacionais dos roteadores participantes são atingidas (⑦). Para todas as VRTs primitivas, as respostas de cada controlador de admissão são repassadas no caminho de retorno das sucessivas chamadas. Quando o controlador de admissão da aplicação recebe uma resposta positiva de todo o processo de negociação, ele solicita, pelo mesmo caminho, uma confirmação que, finalmente, dispara a criação dos recursos virtuais folhas nas VRTs primitivas (⑧) e dos recursos virtuais raízes das novas VRTs compostas, criadas nos níveis de abstração de cada floresta envolvida (⑨).

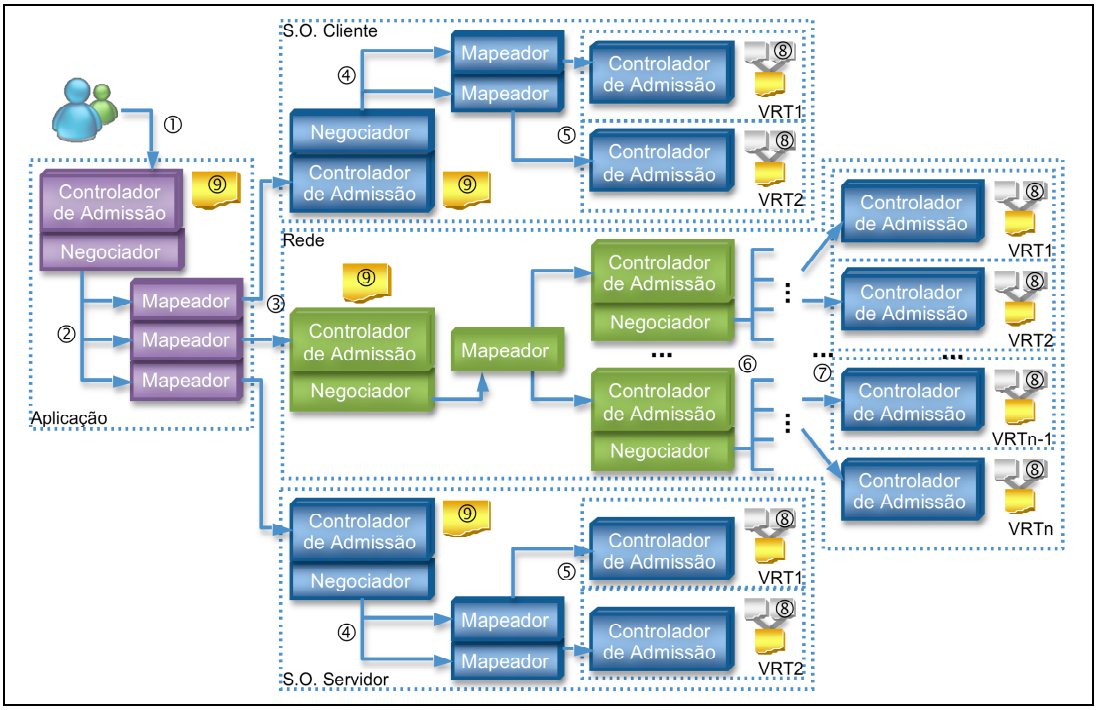

Figura 9 – Orquestração coordenada ao meta modelo VRT

PUC-Rio - Certificação Digital Nº 0321123/CA

Com o conceito de árvore de recursos virtuais composta, o meta modelo VRT tem sua estrutura básica completa. A Figura 10 apresenta um diagrama de classes UML para resumir os relacionamentos presentes no meta modelo VRT, conforme discutido ao longo deste capítulo. Ele deve ser usado como orientação à implementação de suporte ao meta modelo VRT nas plataformas de interesse. Observa-se que tal notação é inadequada a boa parte dos atores envolvidos em MDRM.

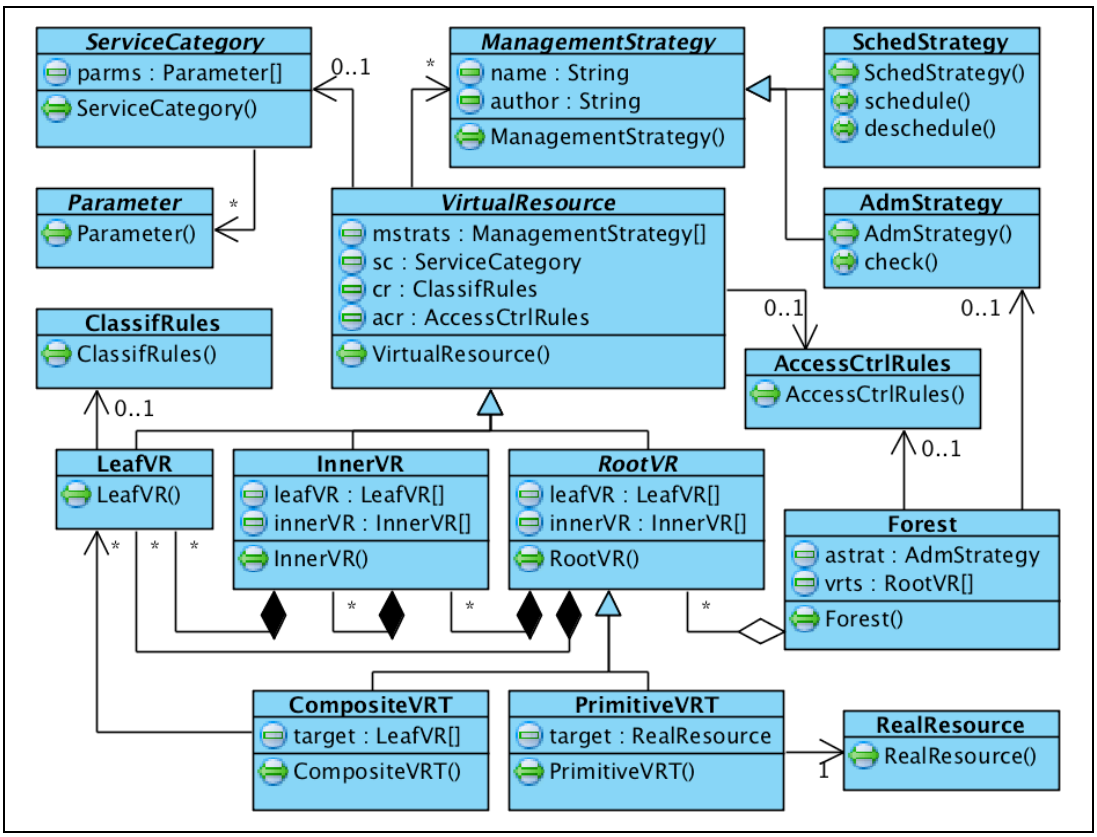

Figura 10 – Diagrama de classes UML para o meta modelo VRT

3.6. Operações de manutenção de modelos VRT

O objetivo de MDRM é oferecer um método de modelagem exclusivamente voltado para o gerenciamento de recursos fim-a-fim que possa ser dinâmica e colaborativamente modificado tanto em tempo de projeto quanto em tempo de operação. Para isso, o meta modelo VRT especifica não somente a estruturação de abstrações para a criação de modelos de gerenciamento de recursos, mas também um conjunto de operações que permite a manutenção desses modelos depois de implantados. Em outras palavras, modelos especificam as árvores e florestas a

PUC-Rio - Certificação Digital Nº 0321123/CA

serem instanciadas inicialmente nas plataformas, enquanto operações de manutenção permitem a modificação dos modelos em tempo de operação.

O meta modelo VRT prevê o uso de sete operações básicas para o gerenciamento de árvores de recursos virtuais e florestas, são elas: *init, traverse, split, merge, tune, release* e *adapt*. Antes de sua efetivação, toda operação deve ser submetida à liberação por parte de um controlador de acesso associado à árvore ou floresta envolvida. A descrição de cada operação encontra-se a seguir.

Init

A operação *init* permite a criação de uma nova árvore de recursos virtuais. A operação recebe como parâmetros de entrada a estrutura inicial da nova árvore a partir de seu nó raiz, e qual deve ser a floresta que a abrigará. O retorno da operação expressa o sucesso ou falha de sua efetivação.

Para a criação de uma VRT primitiva, deve ser informado o recurso real (ou recurso virtual de outra VRT primitiva) a ser gerenciado. Para a criação de uma VRT composta, deve ser informada uma lista de recursos virtuais folhas que participam da composição.

A mesma operação *init* permite também a criação de um nova floresta. A operação recebe como parâmetros de entrada a estrutura inicial da floresta, possivelmente com novas árvores especificadas e qual deve ser a floresta que a abrigará.

Traverse

A operação *traverse* permite o caminhamento pela estrutura de uma floresta ou árvore de recursos virtuais, para fins de captura do estado atual do modelo, rastreamento de reservas, ou qualquer outro tipo de recuperação de informações do subsistema de gerenciamento de recursos. A operação recebe como parâmetros de entrada a floresta ou recurso virtual a ser o ponto de partida e a direção do caminhamento (*child*, *parent*, *down*, *up*). O retorno da operação é o recurso virtual ou floresta na direção de caminhamento solicitada a partir do nó informado.

O caminhamento em uma floresta permite atingir suas VRTs ou suas subflorestas (*child*). O caminhamento em uma VRT, por sua vez, permite a visitação de cada recurso virtual presente na hierarquia, de um pai para um filho (*child*), ou vice-versa (*parent*). Também é possível o caminhamento de uma VRT

PUC-Rio - Certificação Digital Nº 0321123/CA

composta para os recursos virtuais de outras VRTs que compõem sua raiz (*down*) e vice-versa (*up*).

Split

A operação *split* permite realizar uma divisão em um recurso virtual, ou seja, criar um recurso virtual filho. Os parâmetros de entrada são: a especificação da parcela alocada a partir do recurso virtual pai, qual o tipo de recurso virtual a ser criado (intermediário ou folha) e qual é o recurso virtual pai. O retorno da operação expressa o sucesso ou falha de sua efetivação.

A operação *split* somente é efetivada se a parcela solicitada for viável, segundo a avaliação do controlador de admissão associado à árvore. Obrigatoriamente, o recurso virtual pai da divisão deve ser não-folha.

Merge

A operação *merge* permite a criação de uma raiz de VRT composta, a partir da composição de recursos virtuais folhas já existentes. A operação recebe como parâmetros de entrada a lista de recursos virtuais folhas que participarão da composição e a floresta que conterá a nova VRT composta. O retorno da operação expressa o sucesso ou falha de sua efetivação.

A operação *merge* tem o mesmo efeito de uma operação *init* para VRTs compostas. Obrigatoriamente, os recursos virtuais participantes da composição devem ser folhas.

Tune

A operação *tune* permite a sintonização dos parâmetros de QoS que definem as garantias concedidas a um certo recurso virtual. A operação recebe como parâmetros de entrada a nova especificação de parcela a ser alocada do recurso virtual pai e qual é esse recurso virtual pai. O retorno da operação expressa o sucesso ou falha de sua efetivação.

A operação *tune* somente é efetivada se a nova parcela solicitada for viável segundo a avaliação do controlador de admissão associado à árvore. Se malsucedida, a operação não incorre em perda das garantias anteriormente concedidas ao recurso virtual em questão.

Não é possível realizar *tune* de um recurso virtual raiz, uma vez que ele sempre representa a capacidade total do recurso real ou virtual que gerencia.

PUC-Rio - Certificação Digital Nº 0321123/CA

Release

A operação *release* permite a liberação da parcela alocada a um recurso virtual, ou seja, sua destruição. Se o recurso virtual solicitado para remoção possuir recursos virtuais filhos, a operação falha, a não ser se informada a opção de remoção recursiva. A operação recebe como parâmetros de entrada o recurso virtual a ser removido e a opção de remoção recursiva de recursos virtuais filhos. O retorno da operação expressa o sucesso ou falha de sua efetivação.

A mesma operação *release* permite a destruição de uma floresta, com opção de remoção de subflorestas e VRTs que a compõem.

A remoção de uma raiz de VRT composta não incorre na remoção dos recursos virtuais que a compõem.

Adapt

A operação *adapt* permite a modificação das estratégias de gerenciamento já associadas a um recurso virtual ou floresta. A operação recebe como parâmetros de entrada o conjunto de novas estratégias de gerenciamento e o recurso virtual ou floresta que deve ter as estratégias substituídas. O retorno da operação expressa o sucesso ou falha de sua efetivação.

As estratégias de gerenciamento possíveis para cada caso de recurso virtual são:

- Estratégia de admissão e regras de controle de acesso, para as florestas;
- Estratégia de admissão, estratégia de escalonamento e regras de controle de acesso, para os recursos virtuais intermediários; e
- Estratégia de escalonamento e regras de controle de acesso, para os recursos virtuais folhas.

PUC-Rio - Certificação Digital Nº 0321123/CA