

Extensão da Plataforma de Gerência OSIMIS pela Implementação da Função de Medida de Contabilização

Luiz Fernando Kormann¹

Adriano Coser²

Elvis Melo Vieira³

Carlos Becker Westphal⁴

Universidade Federal de Santa Catarina - UFSC
Departamento de Informática - Núcleo de Processamento de Dados
Grupo de Redes e Gerência - Lab. de Redes e Gerência
UFSC - CTC - INE

Cx. Postal 476
CEP 88040-970

Florianópolis - S.C.
Fone:(048) 231-9739

Resumo

A heterogeneidade do ambiente de redes de computadores aliada a necessidade de soluções que integrem seus recursos e serviços, resultou no surgimento de padrões internacionais com o objetivo de possibilitar a interoperabilidade e a interconectividade entre sistemas de diferentes fornecedores. Dentro deste contexto, fazendo referência ao padrão ISO/OSI para gerência de redes, este artigo propõe a extensão da plataforma de gerência OSIMIS pela implementação da Função de Medida de Contabilização. A título de exemplo da função implementada, no final deste trabalho será apresentada uma aplicação de contabilização.

Abstract

Due to the computer network environment heterogeneity and the necessity of solutions that integrate its resources and services, international standards have been proposed making possible the interoperability and interconnectivity among systems from different suppliers. Concerning to the ISO/OSI network management standard, the aim of this paper is to propose an extension of the OSIMIS management platform by implementing the Accounting Meter Function. At the end of this work, an accounting application is been presented as a practical demonstration of the implemented function.

Palavras-chave: Redes de Computadores, Gerência de Redes, Função de Contabilização, Modelo OSI.

1. Introdução

A expansão dos recursos computacionais, aliada à variedade de fornecedores e produtos, tornaram o ambiente de redes de computadores muito heterogêneo e

1. Bacharel em Ciências da Computação pela UFSC. E.Mail: kormann@inf.ufsc.br

2. Bacharel em Ciências da Computação pela UFSC. E.Mail: coser@inf.ufsc.br.

3. Pós-graduando em Ciências da Computação na UFSC e gerente de suporte do Núcleo de Processamento de Dados. E.Mail: elvis@npd.ufsc.br.

4. Professor Titular do Departamento de Informática e do Curso de Pós-graduação em Ciências da Computação da UFSC. E.Mail: westphal@inf.ufsc.br.

complexo exigindo, para ser gerenciado, sistemas proprietários e incompatíveis entre si. A falta de interoperabilidade no gerenciamento dos recursos de uma rede reflete nos altos custos de manutenção, diminui a eficiência na monitoração dos equipamentos utilizados e impossibilita a implantação de medidas de segurança mais eficazes.

Desta forma, surgiram no mercado padrões com o objetivo de possibilitar o gerenciamento de sistemas heterogêneos. Dentre estes padrões destaca-se o modelo de gerência existente na arquitetura OSI (*Open Systems Interconnection*), proposto pela ISO (*International Organization for Standardization*).

Este modelo utiliza os conceitos de engenharia de software e orientação a objetos para prover a comunicação e estabelecer a representação das diversas entidades em um sistema de gerência. O modelo de gerência OSI, rico em funcionalidade, entra em contraste com o modelo *Internet* proposto pela IAB (*Internet Activities Board*). Este último embora muito empregado, apresenta várias restrições funcionais em virtude de sua simplicidade de implementação.

As restrições do modelo *Internet* aliadas à complexidade dos sistemas a serem gerenciados faz com que esforços sejam dispendidos com o objetivo de implementar e popularizar o modelo OSI. Dentro deste contexto, encontramos as ferramentas ISODE (*ISO Development Environment*) e OSIMIS (*OSI Management Information Service*). A primeira fornece uma plataforma para o desenvolvimento de aplicações OSI sobre a arquitetura TCP/IP (*Transport Control Protocol/Internet Protocol*). A segunda, construída sobre o ISODE, apresenta um conjunto de classes e funções que servem de interface para o desenvolvimento de aplicações de gerenciamento OSI.

O modelo de gerenciamento OSI, não está limitado ao tratamento de falhas, apresentando um modelo funcional que inclui também aspectos de desempenho, configuração, segurança e contabilização.

A gerência de contabilização possibilita a monitoração e tarifação da utilização de recursos e serviços na rede. Fornece também, junto com a gerência de desempenho, parâmetros que auxiliam na tomada de decisões a respeito da aplicação de novos recursos na rede ou da redistribuição daqueles já existentes.

Este artigo propõe a extensão da plataforma de gerência OSIMIS pela implementação da Função de Medida de Contabilização [ISO/IEC 10164-10] proposta no modelo funcional OSI. Esta função define objetos gerenciados genéricos, que servem de base para a construção de aplicações OSI envolvendo atividades de contabilização. A verificação do funcionamento dos componentes desenvolvidos foram realizados através de aplicações de gerência disponíveis na plataforma. Estas aplicações foram alteradas e configuradas para fazerem uso dos novos serviços de contabilização.

Afim de propor uma extensão a plataforma de gerenciamento OSIMIS, a próxima seção apresentará uma visão sobre o modelo de gerência OSI, seguido da seção 3 que descreve a plataforma de gerenciamento utilizada. As seções 4 e 5 apresentam, respectivamente, uma introdução e a implementação da função de

medida de contabilização. Neste cenário é exemplificada uma aplicação de contabilização na seção 6 e por fim, na última seção, será apresentada a conclusão do trabalho.

2. Modelo de Gerência OSI

O gerenciamento OSI está baseado no paradigma de orientação a objetos. Desta forma, os recursos gerenciados são representados pelos sistemas de gerenciamento por meio de entidades lógicas denominadas objetos gerenciados. As aplicações de gerenciamento são desenvolvidas através de processos distribuídos conhecidos como gerentes (sistema gerenciador) e agentes (sistema gerenciado).

Como pode ser observado na figura 1, os processos gerentes coordenam as atividades enviando solicitações aos processos agentes. Estes, por sua vez, executam as operações sobre os objetos gerenciados e enviam respostas às solicitações dos gerentes. Os agentes também emitem notificações aos gerentes relatando a ocorrência de alterações no estado dos objetos gerenciados. Estabelecida uma associação entre os processos de aplicação, o gerente é capaz de realizar operações sobre os atributos ou sobre o objeto gerenciado propriamente dito.

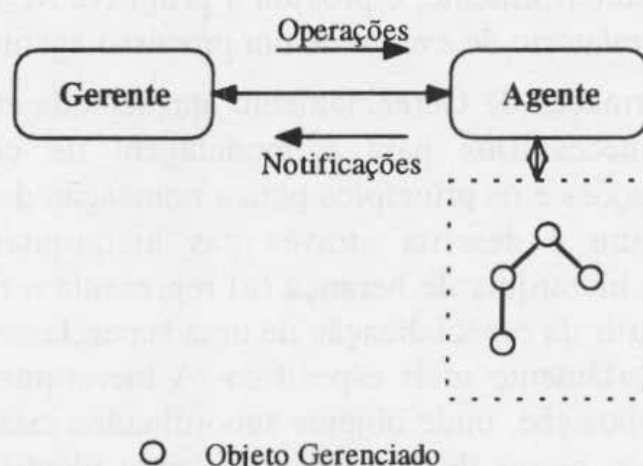


Figura 1: Relacionamento Gerente - Agente

O objeto gerenciado é uma representação lógica de um ou de vários recursos de comunicação ou processamento de dados. O conjunto destes objetos contendo informações de gerência constituem a Base de Informação de Gerenciamento (*MIB - Management Information Base*). Um objeto pode ser definido através de: seus atributos, que contêm informações relevantes na representação de um recurso; seu comportamento diante das operações que ele recebe; notificações indicando a ocorrência de algum evento; e ações que podem ser efetuadas sobre ele. A especificação deste objeto é realizada através *templates* em linguagem ASN.1 (*Abstract Syntax Notation One*) e que descrevem os Princípios para a Definição de

Objetos Gerenciados (*GDMO - Guidelines for the Definition of Managed Objects*[ISO/IEC 10165-4]).

Em um ambiente de gerenciamento OSI a comunicação entre os processos gerentes e agentes é efetuada através do protocolo CMIP (*Common Management Information Protocol*[ISO/IEC 9596]) que implementa as primitivas do serviço de informação de gerenciamento CMIS (*Common Management Information Service*).

Representadas pela tabela 1, as operações implementadas pelo protocolo CMIP podem ser orientadas a objetos, destinadas à realização de operações diretamente sobre as instâncias de objetos; ou orientadas a atributos, que atuam na alteração e consulta dos valores de atributos dos objetos gerenciados.

Operação	Primitiva	Finalidade
Orientada a Objeto	M_Create	Cria e inicializa uma instância de objeto gerenciado
	M_Delete	Destrói uma ou mais instâncias de objetos gerenciados
	M_Action	Executa uma operação sobre um objeto gerenciado
Orientada a Atributo	M_Get	Consulta valores de atributos de um objeto gerenciado
	M_Set	Altera valores de atributos de um objeto gerenciado

Tabela 1: Primitivas CMIP.

Além das descritas anteriormente, é provida a primitiva M_Event_Report para possibilitar o envio de um relatório de evento de um processo agente para um gerente.

A Estrutura de Informação de Gerenciamento apresentada em [ISO/IEC 10165-1] define os conceitos necessários para a modelagem de classes de objetos gerenciados, suas inter-relações e os princípios para a nomeação destes objetos e seus componentes. Esta estrutura é descrita através das hierarquias de herança, de nomeação e de registro. A hierarquia de herança (a) representa a modelagem através de classes de objetos. A partir da especialização de uma superclasse são obtidas novas subclasses com um comportamento mais específico. A hierarquia de nomeação (b) descreve a relação de composição, onde objetos subordinados estão contidos em um objeto superior e estabelece regras de *name binding* para identificar univocamente cada instância de objeto gerenciado. A hierarquia de registro (c) registra as definições das classes de objetos, atributos, ações, notificações e pacotes, com base nas regras estabelecidas pela notação ASN.1.

Em acréscimo ao modelo informacional, a arquitetura de gerência OSI define um modelo funcional onde estão abordados os aspectos de segurança, configuração, desempenho e contabilização. Cada área funcional define um conjunto de uma ou mais funções, cuja implementação provê os requisitos para o gerenciamento de sistemas.

3. Plataforma de Gerência OSIMIS/ISODE

Instalado sobre sub-redes OSI ou TCP/IP, o ISODE versão 8.0 [ISODE] apresenta uma biblioteca de funções que implementa os protocolos da camada de

transporte, sessão e apresentação, e os elementos de serviço da camada de aplicação ACSE (*Association Control Service Element*), ROSE (*Remote Operation Service Element*) e RTSE (*Reliable Transfer Service Element*). Dentre os serviços fornecidos por esta plataforma destacam-se o serviço de diretório X.500 e o de transferência de arquivos FTAM (*File Transfer Access and Management*). São encontradas também, ferramentas de auxílio ao desenvolvimento de aplicações como o compilador Pepsy que gera estruturas C a partir de especificações ASN.1.

Desenvolvido através da abordagem orientada a objetos, o OSIMIS versão 4.0 [OSIMIS] foi implementado em linguagens C/C++ e fornece um conjunto de APIs (*Application Program Interfaces*) para facilitar o desenvolvimento de aplicações de gerência. Estas APIs ocultam parte da complexidade do protocolo CMIP e possibilitam ao usuário do OSIMIS centralizar seus esforços muito mais na aplicação de gerência sendo desenvolvida, do que nos mecanismos de acesso ao protocolo de informação. A figura 2 representa a configuração de uma aplicação gerente - agente utilizando a plataforma OSIMIS/ISODE.

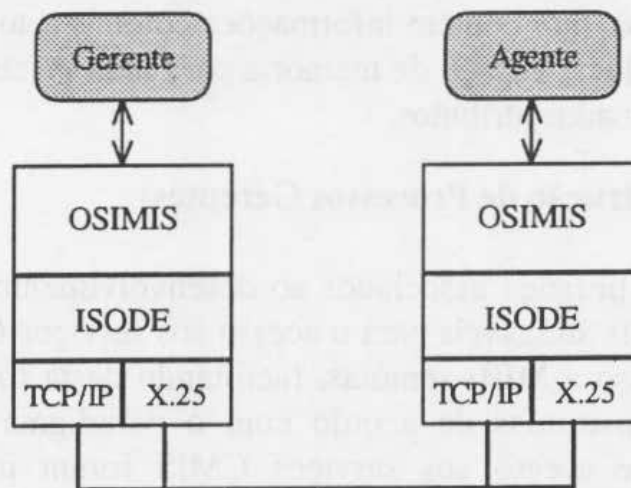


Figura 2: Plataforma de Gerência OSIMIS/ISODE

Estruturas comuns para a construção de processos gerentes e agentes implementam os aspectos para o suporte de aplicações assíncronas orientadas a eventos e a manipulação da sintaxe ASN.1 para a definição dos dados. O primeiro aspecto corresponde a uma estrutura orientada a objetos que permite a organização de aplicações em um regime totalmente orientado a eventos. Em conformidade com a abordagem assíncrona, cada evento interno ou externo é escalonado em uma fila. Este mecanismo prevê a integração com outros mecanismo de coordenação e é implementado através das classes C++ *Coordinator* e *Knowledge-Source*.

O segundo aspecto, suporte a sintaxe ASN.1, tem como objetivo preservar o programador de manipular com estruturas abstratas. Desta forma, os tipos de dados não são representados por estruturas ASN.1, mas por objetos em linguagem C++. Estes objetos contém métodos polimórficos permitindo que os dados sejam codificados e decodificados em estruturas C e ASN.1.

Em acréscimo a estas duas estruturas, suporte a aplicações assíncronas orientadas a evento e a manipulação da sintaxe ASN.1, a plataforma OSIMIS fornece interfaces distintas para a construção de processos agentes e gerentes.

3.1 Sistema Gerenciado Genérico

Apresenta uma interface para o desenvolvimento de aplicações de gerenciamento ocultando completamente detalhes de acessos de serviços CMIS relativos ao endereçamento de objetos, escopo sobre níveis de hierarquia, filtragem de atributos e verificação de erros.

Grande parte destas facilidades é implementada a partir das classes *MOClassInfo* e *MO*. A Classe *MO* é a superclasse de todas as classes de objetos gerenciados e possui métodos e atributos que automatizam o acesso às informações de gerenciamento através do protocolo CMIP. Informações relacionadas à posição dos objetos contidos na Árvore de Informação de Gerenciamento (MIT) também são fornecidas. A classe *MOClassInfo* contém informações comuns a todos os objetos de uma classe, que permitem alocar espaço de memória para uma instância e estabelecer valores *default* para determinados atributos.

3.2 Interfaces para a Construção de Processos Gerentes

Segundo [Pav93] os projetos associados ao desenvolvimento do OSIMIS têm atuado na construção de APIs amigáveis para o acesso aos serviços CMIS. Estas APIs fornecem um eficiente acesso a MIBs remotas, facilitando desta forma a construção de processos gerentes. Construídas de acordo com o paradigma de orientação a objetos, estas interfaces de acesso aos serviços CMIS foram implementadas no OSIMIS, através de duas abordagens distintas: *Remote MIB* e *Shadow MIB*.

A interface *Remote MIB* (RMIB) fornece uma abstração de MIBs remotas OSI empregando a noção de um objeto de associação. Este objeto é usado para encapsular a associação de gerenciamento com um agente remoto, ocultando a utilização dos parâmetros do protocolo CMIP e os requisitos de acesso à árvore de informação de gerenciamento (MIT) remota.

O conceito de *Shadow MIB* (SMIB) é representado pela abstração dos objetos em espaço de endereçamento local e possibilita que a utilização do protocolo de gerenciamento seja completamente ocultado. Parâmetros de acesso aos objetos como escopo e *Distinguished Names* podem ser substituídos por ponteiros.

Além das interfaces acima citadas, o OSIMIS provê várias outras facilidades para o desenvolvimento de aplicações de gerência, como por exemplo: compilador GDMO para a construção de classes de objetos gerenciados, mecanismos de transparência de localização por meio de um suporte ao serviço de diretório X.500, implementação do protocolo SNMP (*Simple Network Management Protocol*), *gateway* a nível da camada de aplicação para a conversão entre os protocolos CMIP e

SNMP, um conjunto de aplicações de gerenciamento genéricas, e agentes específicos para a camada de transporte OSI e para a versão OSI da MIB da camada TCP/IP.

4. Função de Medida de Contabilização

A Função de Medida de Contabilização segundo a definição em [ISO/IEC 10164-10] provê meios para medida e coleta de informações a respeito da utilização de recursos e serviços presentes na rede. E garante também que estes dados estejam disponíveis quando requisitados pelo sistema de gerenciamento, tanto durante o processo de coleta, quanto num momento posterior. A obtenção e representação das informações de contabilização devem ser padronizadas, de modo a garantir a interoperabilidade entre os serviços que seguem o modelo OSI.

Embora existam várias formas pelas quais podem ser realizadas a coleta dos dados, a função de contabilização deve ser genérica o suficiente para possibilitar que os dados coletados sejam tratados adequadamente, de acordo com as necessidades de cada aplicação. Em determinadas situações, os dados coletados podem ser utilizados para a aplicação de tarifas sobre os serviços; em outros casos, estes dados podem fornecer meios para o controle e a distribuição de quotas de consumo aos usuários.

A medida de contabilização é modelada através dos objetos gerenciados associados à contabilização da utilização de recursos. Estes objetos estão associados aos chamados "objetos contabilizados", que representam as demais características dos recursos monitorados.

Aspectos distintos devem ser considerados durante a implementação da medida de contabilização: o primeiro aspecto diz respeito ao controle do registro e emissão dos dados associados com a contabilização, sendo representado por objetos de controle de medida de contabilização (*accounting meter control objects*); o segundo, refere-se especificamente a coleta dos dados de contabilização, sendo representado pelos objetos de dados de medida de contabilização (*accounting meter data objects*). Por fim, mecanismos de armazenamento de informações são necessários para fornecer históricos da contabilização dos recursos. Estes mecanismos são implementados através de objetos de registro de contabilização (*accounting register objects*).

4.1 Objetos de Controle de Medida de Contabilização

Estes objetos habilitam o sistema de gerenciamento para coletar informações sobre a utilização de um recurso, selecionando quais os dados relevantes e sob quais circunstâncias esta coleta deve ser realizada.

O controle de medida especifica os eventos que resultam na atualização e notificação das informações sobre a utilização dos recursos. Tais eventos incluem o

escalonamento por períodos de tempo, ações de controle do próprio sistema de gerenciamento e estímulos provenientes da mudança de valores de atributos.

Os objetos de controle de medida fornecem uma visão genérica do gerenciamento, podendo ser especializados para a contabilização de recursos específicos. Outras classes de objetos gerenciados podem fazer uso dos pacotes definidos no controle de medida, para incorporarem funcionalidades inerentes à contabilização.

4.2 Objetos de Dados de Medida de Contabilização

Representa a utilização de um recurso por parte de um usuário, contendo informações que identificam, entre outras, o usuário do recurso, a unidade de medida utilizada e a quantidade consumida.

As informações sobre a utilização de um recurso podem ser obtidas tanto pelo uso da operação GET para obter valores de atributos de dados de medida, como pela inclusão de parâmetros de medida nas notificações emitidas pela gerência de contabilização.

Apenas as propriedades genéricas dos objetos de dados de medida estão definidas pelas normas OSI. Especializações destes objetos podem ser elaboradas para a contabilização de recursos específicos.

O relacionamento entre os Objetos de Controle de Medida, Objetos de Dados de Medida e Objetos Contabilizados pode ser observado na figura 3.

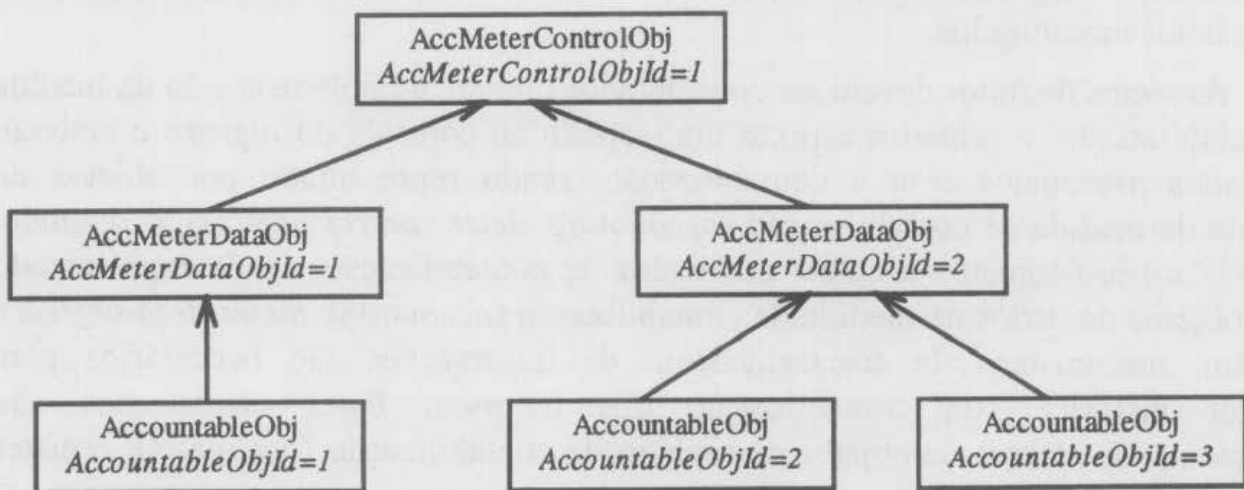


Figura 3: Relacionamento entre os Objetos Gerenciados.

Os objetos de controle de medida de contabilização (*AccMeterControlObj*) fazem referência aos objetos de dados de medida (*AccMeterDataObj*) que eles controlam. Cada Instância do objeto de controle pode controlar várias instâncias de objetos de dados. Cada objeto de dados de medida mantém uma referência à instância do objeto contabilizado sobre o qual são realizadas as coletas de informações.

Para que uma solicitação sobre um objeto contabilizado seja concedida, é necessário que exista pelo menos uma instância de objeto de dado de medida, responsável pela sua monitoração.

Um objeto de dados de medida só pode ser criado caso exista uma instância de objeto de controle de medida para controlá-lo. Em contrapartida, um objeto de controle de medida somente poderá ser destruído a partir do momento em que todos os objetos de dados sob o seu controle tenham sido previamente destruídos.

5. Implementação da Função de Medida de Contabilização

A função de medida de contabilização tem por objetivo permitir a coleta e o controle da coleta de dados sobre a utilização dos recursos e serviços disponíveis na rede. Objetiva também, criar registros sobre os dados coletados, permitindo seu posterior processamento. Os dados registrados podem ser processados, por exemplo, para a associação de tarifas sobre a utilização dos recursos.

Para a obtenção da funcionalidade de coleta, controle e registro de dados sobre a utilização de recursos, foram seguidos os requisitos especificados na normalização [ISO/IEC 10164-10], proposta no modelo funcional OSI e os aspectos sobre a implementação de atributos, ações e notificações, até a obtenção das novas classes de objetos gerenciados. Em acréscimo, foram desenvolvidos os processos para a criação de *name bindings*, que definem o relacionamento de nomeação entre as classes de objetos. Finalmente, questões referentes à integração da função implementada com a plataforma OSIMIS foram solucionadas, resultando em uma extensão à funcionalidade desta plataforma.

5.1 Atributos

Na plataforma OSIMIS, os atributos são representados através de classes. Cada objeto de uma classe de atributos é composto por uma estrutura de dados, que define sua sintaxe, e por um conjunto de métodos, que implementam seu comportamento.

Todas as classes de atributos implementadas são especializações da superclasse *Attr*. Esta classe provê, entre outros, os métodos genéricos para consulta e alteração do valor dos atributos, e métodos para a codificação (da estrutura em C para um elemento de apresentação em ASN.1) e decodificação (de um elemento em ASN.1 para uma estrutura em C).

Embora a plataforma OSIMIS forneça implementações das classes de atributos mais comumente utilizadas, os objetos definidos na função de medida de contabilização são compostos por uma série de atributos inexistentes na plataforma. Foi evidenciada a necessidade da construção de novas classes de atributos, como pode ser observado na tabela 2.

Cada linha desta tabela identifica a sintaxe do atributo especificada na norma de contabilização, a sua respectiva estrutura de dados em linguagem C, a classe C++ que a implementa e a classificação do atributo em multi-valorado (conjunto) ou não.

Sintaxe do atributo	Estrutura em C	Classe em C++	Set valued
AccountableObject-ReferenceList	type_ACCOUNT_AccountableObjectReferenceList *	AccountableObjectReferenceList	√
AccountingId	type_ACCOUNT_AccountingId *	AccountingId	√
ActionArgument	type_ACCOUNT_ActionArgument *	ActionArgument	
ActionResponse	type_ACCOUNT_ActionResponse *	ActionResponse	
DataErrors	type_ACCOUNT_DataErrors *	DataErrors	
MeterInfo	type_ACCOUNT_MeterInfo *	MeterInfo	√
ProviderId	type_ACCOUNT_ProviderId *	ProviderId	
RecordingTriggers	type_ACCOUNT_RecordingTriggers *	RecordingTriggers	√
ReportingTriggers	type_ACCOUNT_ReportingTriggers *	ReportingTriggers	√
ResponderId	type_ACCOUNT_ResponderId *	ResponderId	√
UnitsOfUsage	type_ACCOUNT_UnitsOfUsage *	UnitsOfUsage	√

Tabela 2: Tabela de Classes de Atributos Implementadas.

A implementação de uma classe de atributo acarreta na geração da estrutura de dados e funções de codificação e decodificação utilizando o compilador *Pepsy*. Após o processo de compilação, dois arquivos em linguagem C são construídos. Um arquivo de definição (*header*) e um arquivo de implementação. O primeiro define as estruturas de dados para os valores associados ao atributo e declara as funções de codificação e decodificação específicas para cada atributo. O segundo arquivo, contém a implementação das funções *build* (gera um elemento de apresentação com o valor do atributo em sintaxe ASN.1), *parse* (retorna uma estrutura em C a partir de um elemento de apresentação em ASN.1) e *print* (imprime os componentes de um elemento de apresentação em uma cadeia de caracteres).

Os processos de gerência necessitam ter acesso a funções para manipulação dos valores dos atributos que compõem os objetos gerenciados. Para cada atributo implementado, as seguintes funções devem ser providas:

- *encode*: invoca a função *build*, gerada pelo compilador *Pepsy*;
- *decode*: invoca a função *parse*, gerada pelo compilador *Pepsy*;
- *print*: retorna o valor do atributo na forma de uma cadeia de caracteres;
- *parse*: monta o valor do atributo a partir de uma cadeia de caracteres;
- *compare*: compara dois valores de atributos.

Os passos apresentados até o momento visam a implementação dos aspectos relacionados exclusivamente à sintaxe de um atributo. Aspectos relacionados à semântica do atributo, tais como seu comportamento, são implementados em uma classe. Cada classe será efetivamente utilizada para a composição de classes de objetos gerenciados e para a construção de outras classes de atributos.

As classes de atributos devem apresentar métodos para a codificação e decodificação do valor associado ao atributo, permitindo a manipulação transparente da sintaxe ASN.1; e métodos para o atendimento das primitivas orientadas a atributos do serviço CMIS, tais como *GET* e *SET*.

Para implementar uma nova classe de atributos, deve-se herdar, direta ou indiretamente, a classe *attr*. Os métodos genéricos devem ser especializados para implementar as características particulares de cada classe.

A implementação de novos tipos de atributos é concluída com a construção de bibliotecas de atributos em linguagem C++.

O diagrama da figura 4 apresenta os principais componentes para a construção de classes de atributos.

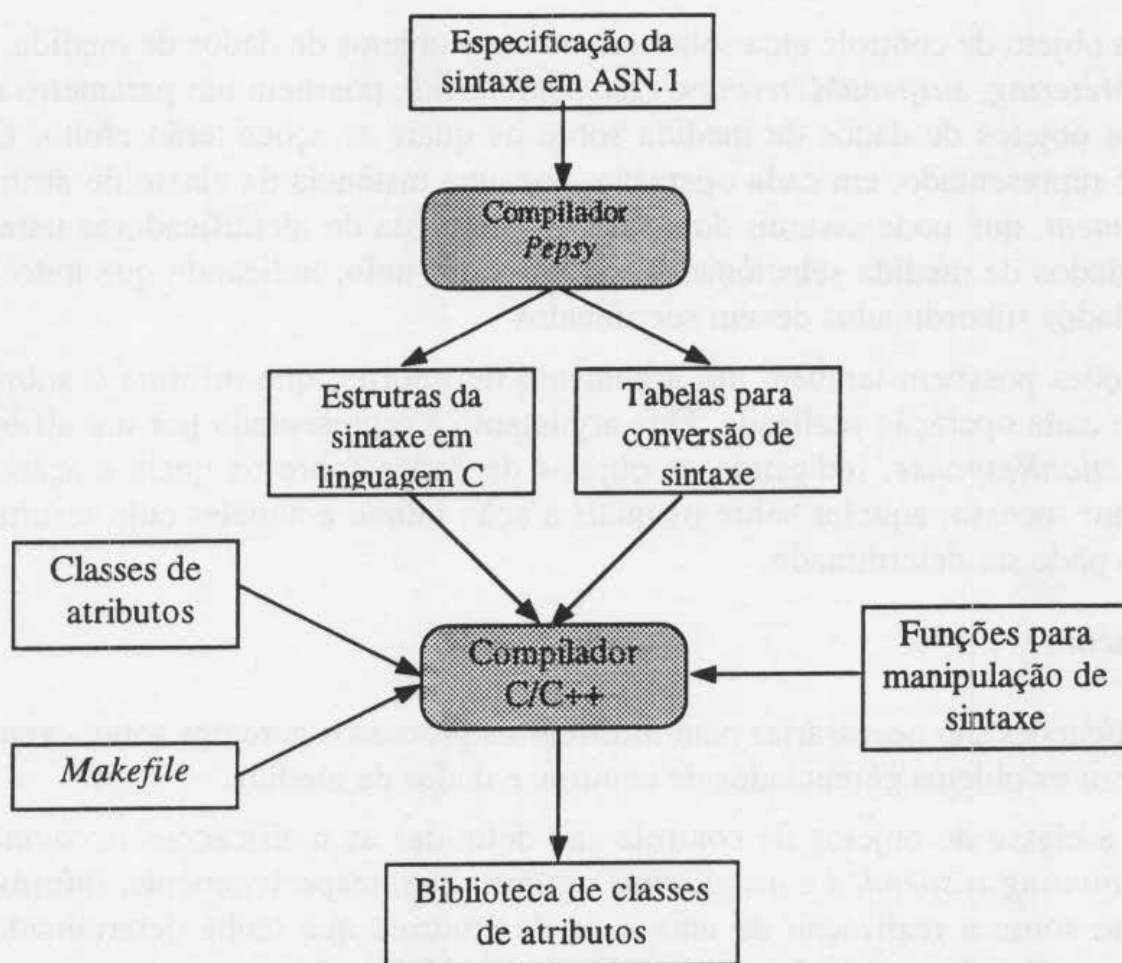


Figura 4. Diagrama da construção da biblioteca de classes de atributos.

Os *templates* em linguagem ASN.1 das diversas sintaxes implementadas, são enviados ao compilador *Pepsy*, que gera dois arquivos contendo funções e tabelas para codificação e decodificação C/ASN.1. Estas tabelas e funções são utilizadas na implementação dos métodos para manipulação das sintaxes. Para cada sintaxe de tipo atributo, é implementada uma classe. Todos os fontes gerados, inclusive aqueles produzidos pelo *Pepsy*, são compilados pelo compilador C/C++, utilizando as diretivas contidas no arquivo *Makefile*. O processo de compilação resulta na construção da biblioteca de classes de tipos de atributos.

5.2 Ações

De acordo com a função de medida de contabilização, somente para os objetos de controle de medida de contabilização estão associadas ações. Estas ações provêm meios para que um agente possa controlar, em resposta às solicitações de um gerente, a coleta de dados sobre a utilização dos recursos, através de operações sobre os objetos de controle. Tais ações possibilitam o comando do início (*start metering*), a suspensão (*suspend metering*) e a retomada das medidas de contabilização (*resume metering*).

Cada objeto de controle atua sobre um ou mais objetos de dados de medida. As ações *startMetering*, *suspendMetering* e *resumeMetering*, possuem um parâmetro que identifica os objetos de dados de medida sobre os quais as ações terão efeito. Este parâmetro é representado, em cada operação, por uma instância da classe de atributo *ActionArgument*, que pode assumir dois valores: uma lista de identificadores para os objetos de dados de medida selecionados; ou um valor nulo, indicando que todos os objetos de dados subordinados devem ser afetados.

As ações possuem também um argumento de retorno, que informa o sobre o resultado de cada operação realizada. Este argumento é representado por um atributo da classe *ActionResponse*, indicando os objetos de dados sobre os quais a ação foi realizada com sucesso, aqueles sobre os quais a ação falhou e aqueles cujo resultado da ação não pôde ser determinado.

5.3 Notificações

Notificações são necessárias para informar os processos gerentes sobre eventos ocorridos com os objetos gerenciados de controle e dados de medida.

Para a classe de objetos de controle são definidas as notificações *accounting started*, *accounting suspended* e *accounting resumed* que, respectivamente, informam a um gerente sobre a realização de uma ação de controle que tenha determinado o início, a suspensão ou a retomada das medidas de contabilização.

Estas notificações possuem como argumento e estrutura de dados *control info* contendo informações sobre o resultado da ação que acarretou a emissão da notificação e dados opcionais sobre o objeto de controle que está emitindo a notificação referentes a unidade de medida empregada e o conjunto de objetos de dados controlados.

No caso da classe de objetos de dados de medida, foi definida uma notificação, que é emitida a um gerente quando se deseja informar sobre a coleta de novos dados de utilização de um recurso por um usuário ou assinante. O argumento associado a esta notificação é representado pela estrutura denominada *accounting data info*, que contém uma série de informações sobre o objeto de dados que identificam o recurso contabilizado, a unidade de medida utilizada, a quantidade despendida e o tipo de tarifa associada.

5.4 Objetos Gerenciados

Para a implementação dos objetos gerenciados foi utilizado o compilador GDMO [OSIMIS GDMO], fornecido pela plataforma OSIMIS. Este compilador gera a estrutura básica de uma classe de objetos gerenciados a partir de um *template* escrito em GDMO. Parte das funções para manipulação dos componentes de um objeto como, atributos, notificações, e ações são geradas pelo compilador. Fica a cargo do projetista de aplicações de gerência, desenvolver a estrutura de dados destes componentes de acordo a respectiva especificação em GDMO e implementar o seu comportamento.

Uma das vantagens da utilização do compilador GDMO, está relacionada ao código C++ gerado, que mantém conformidade com o restante do código implementado na plataforma de gerenciamento. Este fator permite maior facilidade na integração com as interfaces já existentes no OSIMIS.

No caso da função de contabilização, as classes *accountingMeterControl*, *accountingMeterData* e *accountingRecord* foram parcialmente implementadas pelo compilador GDMO. Além do código gerado pelo compilador, métodos para a manipulação de ações e notificações foram implementadas manualmente, para atender aos requisitos particulares de cada classe. No caso da classe *accountingMeterControl*, por exemplo, o método *action*, definido na super classe *MO* existente no OSIMIS, foi especializado para implementar o comportamento das ações inerentes à classe.

Em acréscimo, foram implementadas funções para a geração de identificadores (RDN - *Relative Distinguish Name*) para os objetos instanciados. Tais identificadores são necessários para viabilizarem o funcionamento do esquema de *name binding*. Este esquema é utilizado na construção da hierarquia de nomeação ou de *containment*, que permite identificar, de forma unívoca, cada instância de objeto gerenciado no sistema.

A construção de classes de objetos gerenciados utilizando o compilador GDMO requer, além da especificação em GDMO, a elaboração das tabelas:

- *mo.dat*: tabela dos objetos gerenciados referenciados na especificação. Cada entrada da tabela contém o nome de uma classe de objetos presente na especificação, o nome da classe C++ que a implementa e o nome do arquivo *header* onde a classe está definida;
- *attr.dat*: todos os atributos utilizados na especificação são identificados nesta tabela. Cada entrada contém o nome de um atributo, a classe C++ que o implementa e o nome do arquivo *header* onde a classe está definida;
- *syntax.dat*: identifica as sintaxes empregadas na especificação. Cada entrada contém o nome de uma sintaxe, a classe C++ que a implementa e o nome do arquivo *header* onde a classe está definida.

Os arquivos com as tabelas e as especificações dos objetos gerenciados são enviados ao compilador GDMO, que gera a definição em linguagem C++ da estrutura básica das classes de objetos gerenciados, os arquivos com a implementação das classes especificadas, uma tabela contendo os identificadores de objetos (OID -

Object Identifier) para as estruturas geradas e por fim, um arquivo *Makefile* com as diretivas de compilação para a construção de uma biblioteca com as classes definidas.

O diagrama da figura 5 representa os elementos envolvidos na construção da biblioteca com as classes de objetos gerenciados.

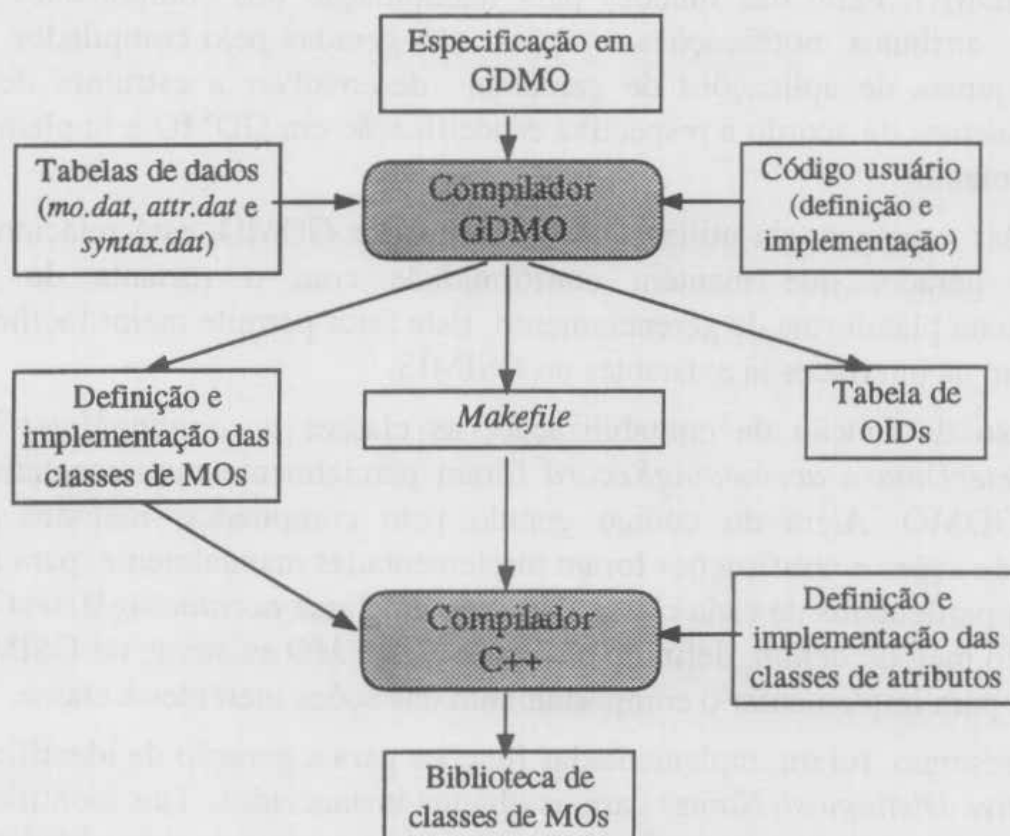


Figura 5: Diagrama de Implementação de classes para a plataforma OSIMIS.

Obtida a implementação dos requisitos básicos para os objetos gerenciados que compõem a função de contabilização, o passo seguinte foi a integração destes com a plataforma de gerenciamento.

O OSIMIS fornece um processo agente, denominado *System Management Agent* (SMA), que implementa parte das funções de gerenciamento de sistemas definidas no modelo funcional da arquitetura OSI, como por exemplo, funções para a emissão de relatórios de eventos, registro de *logs* e objetos métricos. Parte do trabalho realizado, consistiu na alteração do agente SMA, de modo que este passasse a suportar a função de medida de contabilização. Três tabelas existentes no agente foram alteradas para comportarem os componentes da nova função:

- **tabela de objetos gerenciados:** nesta tabela são listadas as classes de objetos que serão suportadas após a inicialização da MIB;
- **tabela de notificações:** esta tabela identifica as notificações que serão suportadas, realizando um mapeamento destas para as respectivas classes de objetos gerenciados;

- **tabela de sintaxes:** identifica todas as sintaxes (para ações, notificações e atributos) que serão suportadas pelo agente após sua inicialização. Cada entrada desta tabela contém o nome de uma rotina, responsável pela inicialização das sintaxes de uma função de gerenciamento.

Em acréscimo, a arquitetura de gerenciamento OSI utiliza-se do conceito de serviço de diretório X.500 para identificar, de forma unívoca, todas as entidades de um sistema. O OSIMIS implementa mecanismos de suporte a este serviço. No entanto, caso estes mecanismos não estejam instalados, pode ser utilizado um meio alternativo, que faz uso de tabelas para mapear os diversos identificadores de objetos. Desta forma, identificadores relacionados a atributos, ações e notificações foram adicionados em uma tabela de configuração do OSIMIS denominada *oidtable.at*. Finalmente, identificadores relacionados a objetos gerenciados e *name bindings* foram incluídos em uma segunda tabela, *oidtable.gen*.

6. Cenário de aplicação da Função de Medida de Contabilização

A medida de contabilização de forma análoga às demais funções de gerenciamento não deve acarretar degradações na performance da rede. Sua execução pode ser efetuada em resposta a uma ação periódica de manutenção. Neste caso, a função de medida de contabilização provê meios de configuração que delimitam sua atuação nos horários de pouco tráfego na rede. Definido o ambiente a ser contabilizado, ações são emitidas para iniciar a coleta de dados, uma vez coletadas estas informações o ambiente anterior ao início da contabilização é restabelecido.

Embora atue sobre recursos compartilhados, os atributos que constituem os objetos gerenciados monitorados pela função não representam informações que possam ser configuradas pelo usuário. Como os serviços providos pela função representam intrinsecamente a coleta de informações, apenas as operações orientadas a atributos referentes a consulta (GET) são utilizadas. Desta forma, mecanismos adicionais de controle de concorrência não são requisitados.

A arquitetura OSI define gerenciamento de sistemas como o gerenciamento do um ambiente OSI ou dos componentes que fornecem seus serviços. Não estão incluídos neste contexto a plataforma de *hardware* e o sistema operacional de suporte aos serviços OSI. Para estes recursos o gerenciamento OSI pode ser também aplicado através de objetos gerenciados. A criação, manipulação e remoção destes objetos configuram o gerenciamento de sistemas.

A título de verificação da funcionalidade da função implementada foi proposta a contabilização da utilização do recurso de CPU entre estações de trabalho distribuídas em rede. Por ser este um recurso constantemente utilizado, este contém parâmetros comumente empregado em aplicações de contabilização.

Como visto anteriormente, a extensão de contabilização desenvolvida para a plataforma OSIMIS fornece toda a atividade de controle e coleta de informações sobre a utilização de recursos. Um vez configurado o cenário de contabilização fica a

cargo do usuário da ferramenta representar por meio de objetos os recursos a serem gerenciados.

O sistema operacional UNIX fornece um sistema de contabilização denominado *sa* que sumariza informações sobre recursos referentes a processamento, memória e dispositivos de I/O. No caso da utilização de CPU, são obtidas informações que discriminam para cada usuário o valor em segundos da quantidade de CPU dispendida, o número de comandos executados e a quantidade de operações de I/O inferidas. A reinicialização destes valores pode ser diária ou mensal.

No desenvolvimento da MIB que contém informações sobre o recurso de CPU a ser contabilizado foram especificados objetos distintos que representam as diferentes características sobre o recurso e o meio de coleta de informações. Específicos para a integração com o sistema operacional os novos objetos incluídos *saObj* e *tariffObj* não são padronizados pela arquitetura OSI.

O esquema apresentado na figura 6, identifica o relacionamento entre as instancias de objetos envolvidos na contabilização.

O objeto gerenciado *saObj* atua como interface direta ao sistema UNIX com o objetivo de abstrair as informações referentes a utilização de CPU. Enquanto que o objeto *tariffObj* identifica o tipo de tarifa a ser empregada. Único para cada recurso da ser contabilizado, este último fornece parâmetros para que taxas distintas sejam empregadas de acordo com a variação da utilização do recurso.

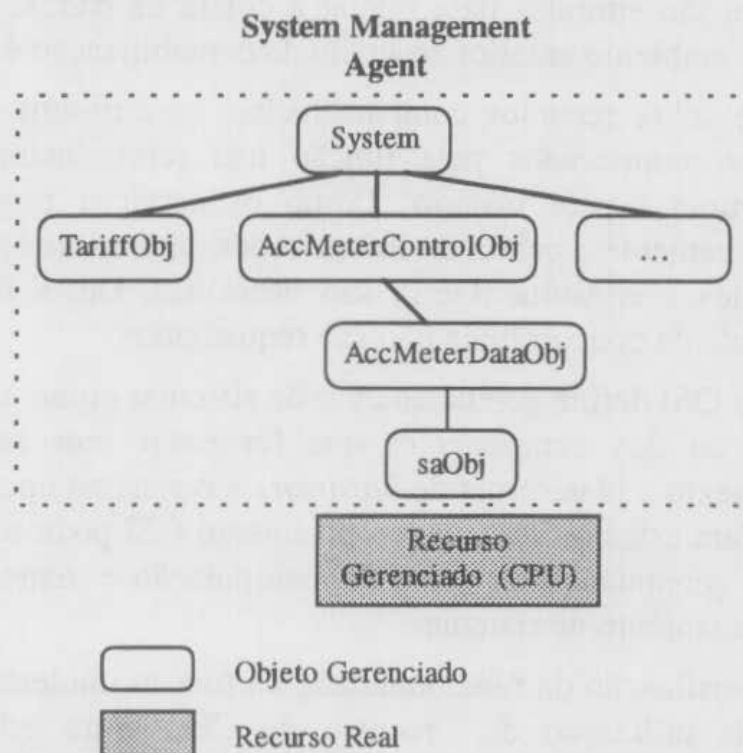


Figura 6: Contabilização da utilização de CPU.

Através da primitiva *Create* fornecida pelo protocolo CMIP, é possível a criação dos objetos de Dados de Medida (*AccMeterDataObject*) para cada usuário

conectado a um terminal. Estes Objetos fazem referência aos *saObj* com o objetivo de obterem informações atualizadas sobre quais os usuários que estão utilizando o recurso, quantidade despendida e a tarifa empregada.

Os objetos de Controle de Medida (*AccMeterControlObject*) coordenam a contabilização através da emissão de ações que determinam o início, retomada e suspensão da contabilização para todas as instâncias de Dados de Medida que tiverem sido identificadas no corpo da ação.

Uma vez realizada a coleta e processada a tarifação notificações podem ser enviadas ao gerente ou registro de *Log* podem ser armazenados. Ambos procedimentos são fornecidos pelo OSIMIS através das classes *EventForwardDiscriminator* e *EventLog*.

7. Conclusão

A proposta inicial do trabalho estabelecia como objetivo central a construção de aplicações para a contabilização de recursos e serviços específicos na rede UFSC. Porém, no início de seu desenvolvimento, foi observado que a plataforma não fornecia nenhum recurso dedicado à gerência de contabilização. Diante deste fato, os objetivos foram alterados, no intuito de enriquecer a plataforma OSIMIS com uma estrutura para a construção de aplicações de contabilização. Como trata-se de uma plataforma totalmente em conformidade com o modelo de gerenciamento OSI, a opção pela implementação da Função de Medida de Contabilização, proposta neste modelo representou a melhor abordagem.

Viabilizada a implementação de uma estrutura de contabilização segundo os padrões definidos pela ISO, pôde-se fazer uso de todos os conceitos e facilidades previstas em sua Estrutura de Informação de Gerenciamento e nos serviços oferecidos pelo CMIS.

Facilmente estendida, como foi observado durante a construção de contabilização, a inclusão de um novo recurso a ser contabilizado é realizado através de objetos gerenciados que o represente logicamente enviando informações à função de contabilização. Desta forma, por meio da função de contabilização implementada, pode-se incrementalmente, criar uma Base de Informação de Gerenciamento que represente as características específicas de cada recurso.

A plataforma OSIMIS/ISODE, em sua proposta de possibilitar a desenvolvimento de aplicações de gerência de acordo com os requisitos propostos na arquitetura OSI, apresenta interfaces que facilitam consideravelmente a construção de novos processos. No entanto, durante o desenvolvimento deste trabalho diversas dificuldades foram encontradas. A falta de documentação a respeito da plataforma de gerenciamento utilizada, fez com que muito tempo fosse despendido no entendimento dos processos para o desenvolvimento de novas aplicações sobre a mesma.

Agradecimentos

Os autores gostariam de agradecer os membros do Departamento de Ciências da Computação, Department of Computer Science, University College of London, Sr. Kevin McCarthy e Sr. George Pavlou cujo apoio e valiosos comentários foram muito importantes para o desenvolvimento deste trabalho.

Referências

- [ISO/IEC 9596] **Information Technology - Open System Interconnection - Common Management Information Protocol**, jul. 1991.
- [ISO/IEC 10164-10] **Information Technology - Open Systems Interconnection Systems Management - Part 10: Accounting Meter Function**, mar. 1993.
- [ISO/IEC 10165-1] **Information Technology - Open Systems Interconnection - Structure of Management Information - Part 1: Management Information Mode**, mar. 1991.
- [ISO/IEC 10165-4] **Information Technology - Open Systems Interconnection - Structure of Management Information - Part 4: Guidelines for Definition of Managed Objects**, mar. 1991.
- [ISODE] Rose, M. T. , **The ISO Development Environment User's Manual**, jul. 1991.
- [OSIMIS] G. Pavlou, S. N. Bhatti, G. Knight, **The OSI Management Information Service User's Manual**, Version 1.0, fev. 1993.
- [OSIMIS GDMO] **OSIMIS GDMO Compiler User Manual**, University College London, ago. 1993.
- [Pav 93] G. Pavlou, G. Knight, S. N. Bhatti, at all, **The OSIMIS Platform: Making OSI Management Simple**, 1994.