

Gerenciamento de Serviços de Telecomunicações Utilizando o Ambiente CORBA

Rodrigo Paixão Franciscani
rpaixao@dcc.ufmg.br

Antônio A. Ferreira Loureiro
loureiro@dcc.ufmg.br

Departamento de Ciência da Computação
Universidade Federal de Minas Gerais
Caixa Postal 702
CEP 30123-970 Belo Horizonte, MG

Resumo

O grande aumento de complexidade das redes de telecomunicações e dos serviços ofertados pelas mesmas têm levado à uma necessidade de re-estruturação do modo como essas redes são gerenciadas. Isto ocorre principalmente nas camadas superiores, ou seja, gerenciamento de serviços e negócios. Os serviços de telecomunicações são cada vez mais distribuídos em ambientes heterogêneos, necessitando de integração com vários sistemas e com a própria gerência da rede, implementada, na maioria das vezes, através de OSI/TMN ou SNMP.

Este artigo mostra primeiramente a importância do padrão CORBA no sentido de facilitar o gerenciamento de serviços de telecomunicações nesse contexto, destacando alguns projetos que estão sendo implementados nessa área. É apresentada em seguida a implementação de um *gateway* CORBA a ser usado na área de administração de assinantes e que está sendo desenvolvido em conjunto com a Batik Equipamentos S.A, fabricante nacional de centrais telefônicas de comutação pública. Ao final é mostrado o desenvolvimento de uma aplicação de gerência que utiliza o *gateway* citado.

Abstract

The increasing complexity of telecommunications networks and services has led to the need of changing the way these networks are managed, mainly at higher TMN layers as service management layer (SML) and business management layer (BML). Telecommunications services are getting more and more distributed in heterogeneous environments and they need to be integrated with many systems, including network management systems normally based on OSI/TMN or SNMP.

This paper discusses the role of using CORBA as the integrating environment for telecommunications services management and presents the implementation of a CORBA gateway to be used in the customer administration area, developed jointly with Batik Equipamentos S. A., a public switch manufacturer. Finally it's described an example application that uses the gateway.

Palavras Chaves

Gerenciamento de Serviços de Telecomunicações, CORBA, TMN e Java.

1. Introdução

Uma tendência importante notada atualmente é que, com a maior presença de concorrência no mercado de telecomunicações e a grande similaridade dos equipamentos usados pelas operadoras, o diferencial competitivo procurado tem sido a oferta de serviços com mais qualidade, rapidez e flexibilidade pelo menor preço possível [1]. Com isso, é necessário que o ambiente de desenvolvimento/gerenciamento de serviços seja bastante dinâmico e eficiente. É preciso que se tenha uma grande integração com outros sistemas e com a gerência da rede de telecomunicações.

A gerência de redes é feita atualmente de forma centralizada a partir de Centros de Gerência de Rede. Neste contexto, no intuito de se conseguir uma maior padronização, são utilizados principalmente o modelo de gerenciamento OSI/TMN (Telecommunications Management Network) [2], que se baseia no protocolo CMIP e o gerenciamento Internet, que utiliza o protocolo SNMP [3].

No entanto, a utilização desses padrões não é adequada para o gerenciamento de serviços (SML – *Service Management Layer*, dentro do modelo TMN de cinco camadas), de acordo com os requisitos atuais. Isto ocorre principalmente devido ao modelo gerente-agente não ser adequado ao desenvolvimento e execução de aplicações distribuídas [4], por causa de sua natureza tipicamente estática e hierárquica. Além disso, é necessária a integração com outros sistemas que não seguem necessariamente os mesmos padrões, pois o ambiente de gerenciamento/desenvolvimento de serviços é bastante heterogêneo, tanto em se tratando de linguagens/ambientes de programação como de plataformas.

Dentro do contexto apresentado, uma proposta bastante interessante é a utilização de CORBA como ambiente para o desenvolvimento e execução de aplicações distribuídas na camada de gerenciamento de serviços. Este artigo baseia-se nesta abordagem.

Primeiramente é mostrada uma visão geral de CORBA e sua utilização no gerenciamento de serviços de telecomunicações, destacando pesquisas e implementações, algumas delas já comercialmente disponíveis, que vêm sendo feitas pelas empresas e institutos ligados à área de telecomunicações. Em seguida, é descrito o sistema que vem sendo desenvolvido seguindo essa abordagem, sendo dada especial atenção para o gateway CORBA. Ao final são mostradas as conclusões, destacando-se as vantagens do sistema proposto, mostrando as dificuldades encontradas e as implementações que podem ser feitas futuramente.

2. CORBA

O padrão CORBA (Common Object Request Broker Architecture) [5], [6] foi definido pelo OMG (*Object Management Group*) e seu objetivo principal é prover um ambiente para o desenvolvimento e execução de aplicações distribuídas de modo transparente. A tecnologia é totalmente baseada em objetos, que se comunicam através do ORB (*Object Request Broker*), que é tido como um “barramento de software” e encapsula as funções de localização dos objetos requisitados e transporte das informações.

Os objetos disponibilizam sua funcionalidade às aplicações clientes através de interfaces definidas em OMG IDL (*Interface Definition Language*). Esta linguagem possui uma sintaxe padronizada que permite definir os objetos, seus dados e métodos exportados e ainda exceções que podem ser geradas em caso de erro.

A linguagem IDL é completamente independente da linguagem de programação e do sistema operacional em que estão implementados os objetos. Uma aplicação, para ser disponibilizada em um ambiente CORBA, precisa inicialmente definir a sua interface IDL e realizar o mapeamento/compilação desta interface para a linguagem de programação que está sendo usada. O resultado desta compilação é a geração de dois módulos de software. O primeiro é o *skeleton*, que é utilizado pela implementação do objeto (servidor) para poder receber e responder às requisições vindas pelo ORB. O segundo módulo é o *stub*, utilizado pelo cliente para realizar a interface com o ORB para fazer as requisições ao servidor. Existe ainda a possibilidade de se implementar o cliente ou o servidor sem a geração prévia destes dois módulos de software. Isto é feito com a adição de um código um pouco mais complexo no servidor ou no cliente utilizando a interface de invocação dinâmica – DII (cliente) ou a interface de *skeleton* dinâmico – DSI (servidor). O módulo Adaptador de Objeto oferece diferentes funcionalidades usadas para integrar a implementação do objeto com o ORB (figura 1).

Atualmente já é possível implementar objetos para um ambiente CORBA em diversas linguagens, como C, C++, Java, Delphi, Ada, Smalltalk e COBOL. Como a linguagem IDL é padronizada e independente da linguagem de programação, é possível um cliente implementado em qualquer uma das linguagens suportadas utilizar serviços de um objeto implementado em outra (inclusive em sistemas operacionais diferentes).

Outro ponto importante que faz com que CORBA seja um ambiente distribuído bastante completo são os serviços disponíveis. Os chamados CORBAservices e CORBAfacilities são conjuntos de aplicações padronizadas que podem ser usadas para facilitar o desenvolvimento e execução de aplicações usuárias no ambiente CORBA. Alguns exemplos importantes são o serviço de nome, segurança, notificação/evento, transação e tempo.

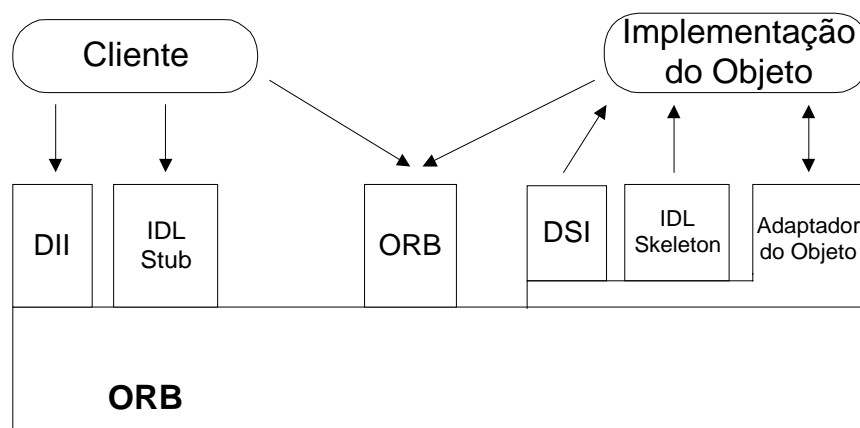


Figura 1 - Arquitetura CORBA

3. CORBA em Gerenciamento de Serviços de Telecomunicações

O padrão CORBA tem tido um destaque cada vez maior na área de telecomunicações. O seu poder como um ambiente integrador de sistemas distribuídos [7] vem fazendo com que várias iniciativas estejam sendo tomadas no sentido de se utilizar CORBA tanto no gerenciamento de redes como de serviços. Isto pode ser constatado através dos investimentos feitos principalmente pelos fabricantes de equipamentos e *softwares* para telecomunicações na criação de ambientes de testes e protótipos de aplicações. Na seção 3.2 são descritos alguns

desses projetos [4], [8], [9], [10], [11], [12], [13], [14], [15].

3.1 Características

No ambiente de uma operadora de telecomunicações, o gerenciamento de serviços tem um escopo maior do que o gerenciamento de redes [16]. Para se oferecer eficientemente os diversos serviços aos clientes (0800, 0900, bina, tele-conferência, chamada em espera, sigame, caixa postal e outros) é preciso que diversos sistemas estejam integrados. A disponibilização de um serviço a um cliente requer, além da própria configuração do(s) elemento(s) envolvido(s) (via interface com o gerenciamento de rede), a interação com outros sistemas. Um exemplo seria o sistema financeiro, que precisa ser acionado para que possa ser feita a cobrança do serviço.

Esta integração de sistemas heterogêneos em um ambiente distribuído é a principal motivação para o uso de CORBA. Com relação à interface com os sistemas de gerenciamento da rede, devido ao grande investimento já feito em sistemas proprietários ou padronizados (TMN ou SNMP), a integração é feita normalmente com a utilização de *gateways*. Estes *gateways* precisam fazer tanto a conversão de protocolos de comunicação como a conversão do modelo de informação utilizado. A parte relacionada à conversão de MIB's (*Management Information Base*) chamada de tradução de especificação já está padronizada pelo OpenGroup [17], [18]. No entanto, a tradução dinâmica de mensagens e protocolos, chamada de tradução de interação, ainda está em período de padronização pelo OMG [19].

A integração dos diversos sistemas de diversas áreas das operadoras de telecomunicações em um ambiente de gerenciamento de serviços utilizando CORBA pode ser visto da figura 2.

O sistema mostrado na seção 4 disponibiliza o gerenciamento de centrais telefônicas a partir de um ambiente CORBA utilizando um *gateway* para um protocolo proprietário.

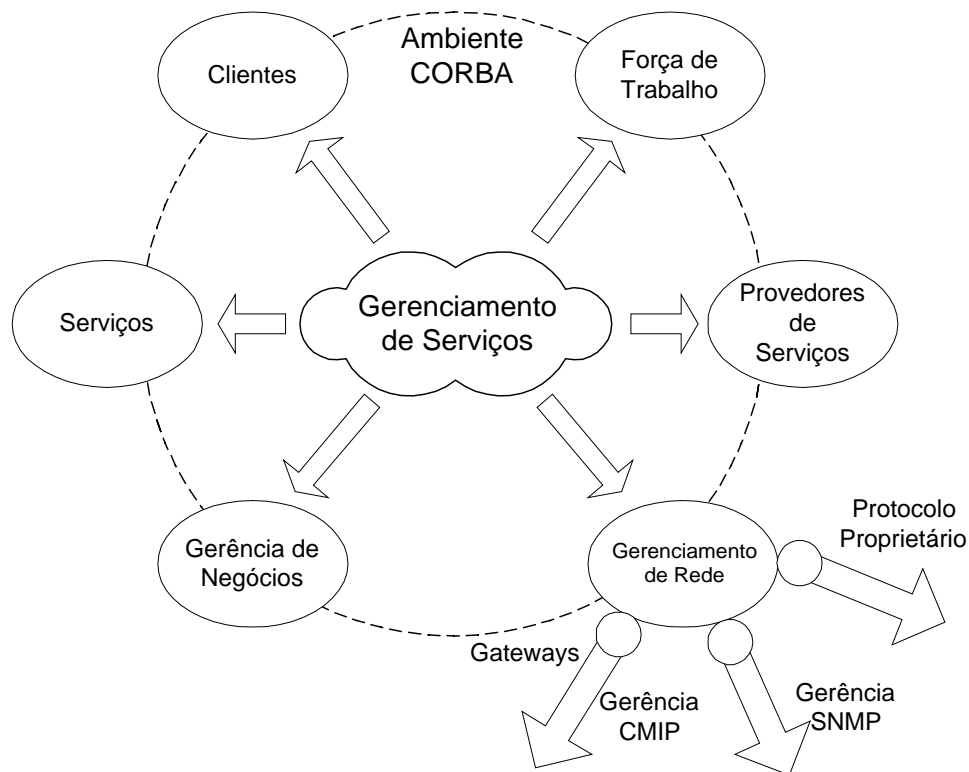


Figura 2 - Gerenciamento de Serviços e CORBA

3.2 Projetos e Produtos Relacionados

Como foi dito anteriormente, existem vários projetos que lidam com a utilização de CORBA na área de gerenciamento de redes e serviços. Também já estão disponíveis no mercado algumas plataformas de gerência que incorporaram o padrão CORBA, sendo que a maioria delas é baseada na disponibilização de um ambiente CORBA de gerência capaz de se integrar com agentes de supervisão SNMP e OSI/TMN.

Um exemplo deste tipo de plataforma comercial é o HPOpenView [12], que além de permitir esta integração, traz um ambiente com várias facilidades para a implementação de aplicações de gerenciamento distribuídas [13].

Outra iniciativa de um fabricante de equipamentos de telecomunicações consiste da implementação de um protótipo que também faz a integração com os protocolos CMIP e SNMP e ainda provê API's para o desenvolvimento de aplicações de gerenciamento baseadas na Web [8].

Um terceiro exemplo a ser citado é a implementação de um protótipo (por parte de um fabricante de *software* para telecomunicações) de um sistema que faz o gerenciamento de um serviço de vídeo sob demanda (VOD), fazendo o gerenciamento dos recursos da rede também através de um *gateway* TMN/CORBA [9], [10].

Outro exemplo é a implementação de protótipos e realização de pesquisas por parte de um grande fabricante de equipamentos de telecomunicações para testar e aprimorar a utilização de CORBA na implementação de serviços para redes “inteligentes” [11].

Finalmente pode-se destacar também a utilização de CORBA como o ambiente sobre o qual são executadas as aplicações de gerenciamento do padrão TINA-C [14], [15], tanto no que diz respeito à gerência de rede como de serviços.

4. Arquitetura Geral do Sistema

O gateway CORBA se situa dentro de uma nova arquitetura de gerenciamento que vem sendo desenvolvida para as centrais de comutação pública Batik. O projeto SGD (Sistema de Gerência Distribuído) tem como objetivo principal facilitar o acesso das aplicações de gerência às centrais via rede TCP/IP, encapsulando o protocolo proprietário.

O sistema é composto de um módulo de análise e execução de comandos (MAEC), que faz a comunicação com as centrais telefônicas executando o comando solicitado (seguindo a linguagem de comandos proprietária) e retornando a resposta, em formato texto plano. Este módulo foi desenvolvido totalmente em Java, utilizando fortemente o conceito de *threads* para que o tratamento das solicitações dos clientes pudesse ser feita de modo independente, tornando o sistema bastante robusto.

Sobre o MAEC são desenvolvidos outros módulos de software com o objetivo de prover vários tipos de acesso a diferentes sistemas de gerência. Um destes módulos provê o acesso ASCII via telnet. Outro módulo é exatamente o objetivo deste estudo, o gateway CORBA, conforme mostrado na figura 3.

Em um primeiro momento, os módulos MAEC+CORBA são responsáveis por grupos de centrais, que são acessadas via protocolo proprietário através de linhas privadas ou discadas. Assim, aplicações de gerência têm acesso às centrais através do ambiente CORBA de modo transparente sem se preocupar com o protocolo proprietário (figura 3).

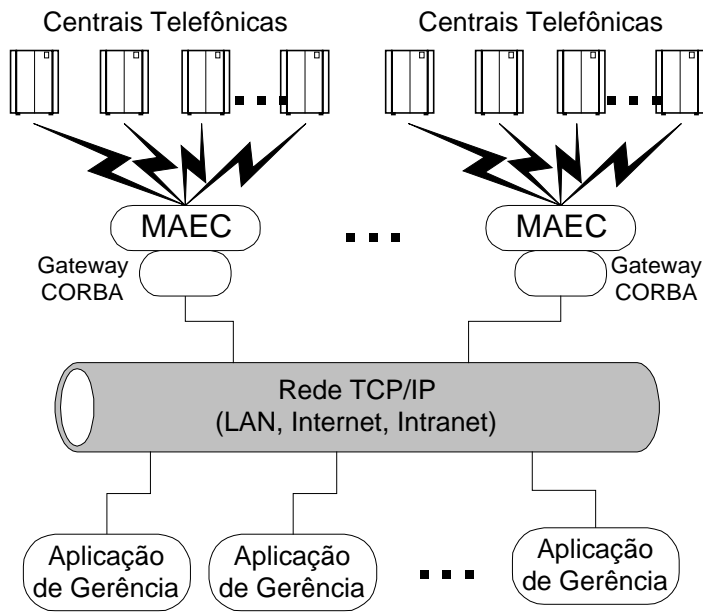


Figura 3 - Arquitetura do sistema (primeira versão)

No entanto, também já está disponível uma outra arquitetura mais apropriada ao desenvolvimento de aplicações de gerência distribuída, incorporando os módulos MAEC e CORBA na própria central telefônica. Assim, as operadoras de telecomunicações passam a ter acesso direto a cada central e aos serviços disponíveis no módulo CORBA via IIOP, como mostrado na figura 4.

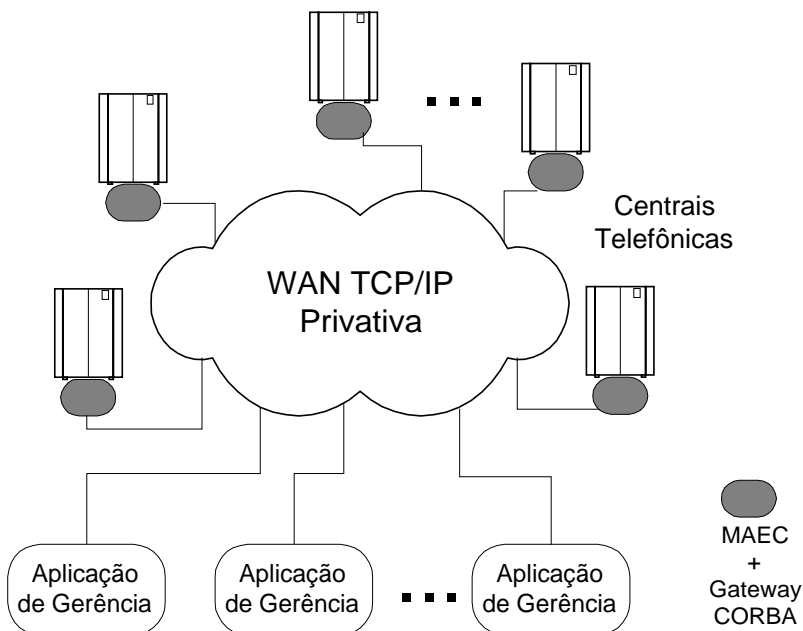


Figura 4 - Arquitetura do sistema (segunda versão)

5. Gateway CORBA

A primeira decisão tomada em relação ao desenvolvimento do gateway CORBA foi com relação à especificação do módulo IDL. Como o acesso às centrais Batik é feito de acordo com comandos que seguem uma linguagem proprietária, existiam várias alternativas em relação à especificação da IDL. A linguagem de comandos segue o seguinte padrão:

```
<VERBO COMANDO>-<OBJETO COMANDO> : <PARAMETRO>=<ARGUMENTO>  
                                     [ , <PARAMETRO>=<ARGUMENTO> ] ...  
                                     [ , <PARAMETRO>=<ARGUMENTO> ]
```

Sendo que as respostas seguem a convenção:

```
<PARAMETRO>=<ARGUMENTO> [ , <PARAMETRO>=<ARGUMENTO> ] ...  
                           [ , <PARAMETRO>=<ARGUMENTO> ]
```

Como exemplo, para se obter as informações relacionadas a um assinante, diga-se 499-5858, deve-se executar o seguinte comando:

```
> IT-ASS : ASS = 499-5858
```

Sendo que um exemplo de resposta seria resposta:

```
> ASS = 499-5858, TER = 1-1-12, EST = SRV, SNL = MF,  
    RTT = TOREST, RTO = IRREST,  
    SSU = BIN DDC
```

São usados mnemônicos na linguagem de comandos para simplificar a representação da informação. Na resposta acima, por exemplo, “EST = SRV” significa que o estado operacional do assinante é “em serviço”.

Assim, este é o tipo de interface que existe entre o gateway CORBA e o MAEC. O MAEC recebe comandos como mostrado acima e retorna a respectiva resposta.

A IDL poderia ser definida baseada nos comandos proprietários Batik. No entanto, este tipo de abordagem dificultaria a integração do sistema proposto com as aplicações de gerência, que precisam comunicar e atuar sobre centrais telefônicas (e outros equipamentos) dos mais diversos fabricantes.

Outra opção para a definição da interface IDL seria a utilização de um modelo que fosse padronizado e que pudesse ser facilmente integrado a aplicações de gerência genéricas. Neste sentido optou-se por definir o módulo IDL a partir de um modelo padrão, o modelo de informação (MIB) do TMN, baseado em especificações GDMO e ASN.1 [2].

É importante frisar que foi usado somente o modelo de informação TMN, não sendo implementada a pilha de protocolos Q₃ pois a comunicação com o gateway CORBA é feita via IIOP.

Como a MIB TMN das centrais Batik já havia sido especificada, era necessário fazer a tradução do modelo GDMO/ASN.1 para IDL. A primeira etapa do desenvolvimento consistiu da tradução da parte da MIB relacionada à Administração de Assinantes.

5.1 Tradução GDMO – IDL

A tradução foi feita utilizando um tradutor comercial [20] que é aderente aos padrões

XoJIDM [18], que especifica regras para o mapeamento GDMO-IDL. As regras que estão padronizadas e foram usadas no *gateway* se referem à tradução de especificação, ou seja, à tradução estática dos modelos de informação.

Com isso, algumas alterações tiveram que ser feitas no que diz respeito à dinâmica do processo de gerência. Em relação aos *name-bindings*, por exemplo, foi feito um processo de simplificação buscando a eficiência do sistema.

O modelo GDMO especifica um modo de se construir a árvore de instâncias de objetos que estão ligados ao objeto “Central Telefônica”, no qual devem ser instanciados todos os objetos - unidades, placas, circuitos, assinantes, juntores, etc.

A abordagem tomada na implementação do *gateway* é instanciar somente o objeto “Central Telefônica” e, a partir daí, os clientes que se conectarem a esta central no ambiente CORBA solicitam os objetos sobre os quais desejam atuar. Assim, depois de feita a tradução da MIB TMN para IDL, foram acrescentados os métodos para a obtenção dos objetos no módulo IDL referente à central. A sintaxe utilizada foi a seguinte (exemplo de obtenção da referência a um assinante):

```
Assinante obterAssinante(in string id);
```

Esta abordagem é bastante interessante em termos de desempenho pois evita a necessidade de se instanciar até dezenas de milhares de objetos, deixando o servidor mais leve.

5.2 Arquitetura

O gateway CORBA pode ser dividido em três módulos, o servidor CORBA, o *Parser* e o módulo que faz a interface com o MAEC conforme ilustrado na figura 5.

O servidor CORBA é responsável por instanciar os objetos “central” e por receber e responder as requisições vindas do ORB. Para responder a uma requisição ligada a um determinado objeto da central, como um assinante, por exemplo, o servidor verifica se já existe um objeto que representa aquele assinante; se existir, é passada a referência, senão é criado o objeto. A partir daí, todos os métodos que são solicitados sobre o objeto precisam ser transformados em comandos passados para a central.

Essa transformação é feita pelo módulo Interface MAEC, que passa o comando para o MAEC. O texto da resposta é passado para o módulo *Parser*, que retira a informação necessária que deve ser passada de volta para o servidor CORBA.

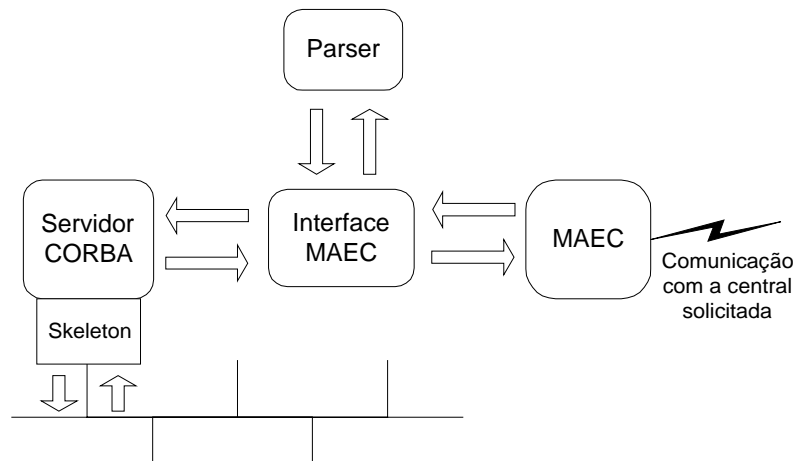


Figura 5 - Arquitetura do gateway CORBA

5.3 Aspectos da Implementação

A implementação do *gateway* foi feita em Java, utilizando-se a implementação de CORBA da Borland/Inprise, o Visibroker [21].

Inicialmente, foram detectados problemas no mapeamento do módulo IDL para as classes e tipos Java. O compilador utilizado não reconhecia tipos definidos através de mais de um nível de *typedef* nos módulos IDL. No entanto, com uma versão mais nova do compilador, este tipo de problema foi solucionado.

Em relação à implementação do próprio servidor CORBA, para que se pudesse tratar cada solicitação vinda do ORB de modo independente, foi necessário implementar um mecanismo que criava uma *thread* para cada cliente e monitorava a execução desta *thread* até o fim das solicitações daquele cliente.

Em se tratando da segurança do sistema, optou-se por utilizar a segurança do próprio MAEC, que possui um módulo de cadastro de usuários e comandos que podem ser executados. No entanto, este tipo de abordagem não é ideal, pois, para a identificação inicial do usuário cliente, são passados o seu nome e senha (via rede TCP/IP) em texto não criptografado. Para solucionar este problema está prevista para breve a utilização do modo SSL de segurança do Visibroker.

6. Exemplo de aplicação de gerência utilizando o sistema

Nesta seção é descrita uma aplicação de gerência implementada utilizando o sistema descrito anteriormente. Primeiro são apresentadas algumas características gerais e, em seguida, a especificação e implementação.

6.1 Características Gerais

Como foi mostrado anteriormente, o padrão CORBA traz muita flexibilidade em relação à utilização de diferentes linguagens e plataformas dentro do ambiente distribuído de modo transparente. Além disso, é de muita importância o suporte que é dado pela gama de serviços presentes.

Assim, disponibilizando-se o acesso às centrais telefônicas através de CORBA, fica facilitado o desenvolvimento de aplicações de gerenciamento de serviços distribuído no ambiente da operadora de telecomunicações. Um exemplo típico são as aplicações de tarifação, muitas vezes escritas em COBOL (que pode ser integrada ao ambiente CORBA) e que poderiam ter acesso direto a esses dados utilizando o sistema proposto anteriormente.

Com o intuito de mostrar a utilidade e flexibilidade do sistema proposto, está sendo desenvolvida uma aplicação de gerenciamento ligada à área de Administração de Assinantes e que utiliza o gateway CORBA. As operações escolhidas para constar da implementação são aquelas que estão ligadas diretamente à atribuição e manutenção de serviços dos assinantes.

Este tipo de solicitação por parte do cliente é feito normalmente através do contato com telefonistas da operadora, que fazem a operação solicitada e dão o retorno ao cliente imediatamente. Devido a essa característica, a aplicação está sendo desenvolvida em forma de *applet* Java, facilitando ao máximo o acesso aos gateways CORBA.

Neste ponto é interessante ressaltar as vantagens de utilização de CORBA mesmo quando não existe o par cliente-servidor escrito em linguagens diferentes, já que tanto a *applet* cliente como o gateway CORBA estão escritos em Java. Poderia-se pensar que seria mais

interessante fazer a implementação utilizando-se RMI (*Remote Method Invocation*) [22]. No entanto, a utilização de CORBA é altamente interessante também neste caso pela transparência de localização de objetos e pela possibilidade de utilização de outros serviços.

Com CORBA pode ser feita a comunicação entre a *applet* Java e qualquer outro objeto que esteja no ambiente através da utilização de um servidor *proxy* IIOP no servidor Web de onde a *applet* veio. Para o cliente esse servidor *proxy* é transparente, ou seja, é como se estivesse comunicando diretamente com o objeto.

6.2 Especificação e Implementação

Foram especificadas as seguintes operações relacionadas aos assinantes de uma operadora de telecomunicações:

1. Desliga/religa
Operações de bloqueio do terminal telefônico do assinante quando ocorre inadimplência e desbloqueio quando a situação é normalizada.
2. Serviços suplementares
Atribuir/retirar serviços suplementares aos assinantes, como chamada em espera, bina e tele-conferência.
3. Teste de linha de assinante
Operações para testar a qualidade da linha do assinante e descobrir a causa dos problemas reportados.
4. PABX virtual
Atribuição e manutenção das informações de linhas e assinantes ligados via PABX virtual.
5. Grupos de Busca
Atribuição e manutenção das configurações dos grupos de busca.

É interessante notar que a maioria das operações citadas anteriormente também precisam ter uma interface com outros sistemas da operadora de telecomunicações, como o sistema financeiro (no caso de atribuição de serviços suplementares ou bloqueio de assinante, por exemplo) ou com o sistema que gerencia a força de trabalho (no caso de teste de linha detectar algum defeito que necessite da visita de um técnico, por exemplo). Esta interface com outros sistemas é bastante facilitada com o uso do ambiente CORBA.

No desenvolvimento desta aplicação foi utilizado o serviço de nomes do CORBA (*Naming Service*) para facilitar a localização dos assinantes em suas respectivas centrais. Com a utilização do serviço de nomes, toda central, quando é instanciada, define que além do seu identificador normal (nome da central) ela pode ser encontrada no ambiente através dos prefixos ligados a ela. É como se a central tivesse um nome e vários apelidos.

Assim, a pessoa que está operando o sistema na *applet*, ao identificar o número do assinante que está solicitando o serviço, envia um pedido ao ORB para se ligar àquele assinante. Através do serviço de nomes, é localizada a central à qual o assinante está ligado e o módulo CORBA responsável pela central retorna a referência ao assinante desejado. Sobre essa referência, podem ser feitas as operações desejadas. Tudo isso é feito de modo transparente para quem utiliza o sistema.

Em relação à implementação da interface gráfica da aplicação, foi utilizada a biblioteca JFC/Swing da Sun [23]. Este sem dúvida foi um ponto negativo, pois pôde-se constatar que esta biblioteca ainda não está totalmente madura, sendo suportada somente pelas versões mais recentes dos principais *browsers* do mercado. Com isso, perde-se um pouco em termos de portabilidade da aplicação.

7. Conclusões e Trabalhos Futuros

A indústria de telecomunicações está investindo bastante em pesquisas envolvendo CORBA e à medida em que essa tecnologia vai amadurecendo espera-se que aumente seu uso comercial nas operadoras de telecomunicações.

Mais especificamente, pode-se constatar que CORBA terá um papel bastante importante na área de gerenciamento de redes, principalmente no escopo mais amplo do gerenciamento de serviços. É uma tecnologia com um grande poder de integração permitindo preservar o investimento feito nos sistemas legados e, ao mesmo tempo, prover um ambiente completo para o desenvolvimento e execução de aplicações distribuídas baseadas em objetos.

Assim, este trabalho procurou se adiantar a essa tendência mostrando a iniciativa de um fabricante de equipamentos de telecomunicações em disponibilizar da maneira mais aberta possível o acesso às funcionalidades de suas centrais telefônicas. A abordagem é diferente da maioria encontrada no mercado, pois partiu-se do protocolo proprietário de gerenciamento para se disponibilizar o acesso CORBA. A maioria das soluções disponíveis atualmente, como mostrado na seção 3, são focadas na utilização de *gateways* comerciais que permitem o acesso a partir de um ambiente CORBA a agentes de supervisão OSI/TMN ou SNMP que já estejam em operação.

O próximo passo na implementação deste projeto é implementar um mecanismo mais forte de segurança, fazendo com que as trocas de mensagens entre os clientes e o gateway CORBA seja feito utilizando criptografia.

8. Referências Bibliográficas

- [1] H. Vanderstraeten, C. Vermeulen, *Software Agents for Advanced Telecommunications Services*, Alcatel Telecommunications Review - 1st Quarter 1998
- [2] ITU-T. M3000 series TMN Recs.; M.3020: TMN Interface Specification Methodology; M.3100: Generic Network Information Model; M.3180: Catalogue of TMN Management Information; M.3200: TMN Management Services; M.3300: TMN Management Capabilities; M.3400: TMN Management Functions
- [3] W. Stallings, *SNMP, SNMP-v2 and CMIP: The Practical Guide to Network Management Standards*, Addison Wesley, 1993
- [4] J. P. Redlich, M. Suzuki and S. Weinstein, *Distributed Object Technology for Networking*, IEEE Communications Magazine, October 1998
- [5] Object Management Group OMG - *CORBA Especifications*:
<http://www.omg.org/library/downinst.html>
- [6] J. Siegel, *CORBA - Fundamentals and Programming*, John Wiley, 1996
- [7] S. Vinoski, *CORBA: Integrating Diverse Applications Within Distributed Heterogeneous Environments*, IEEE Communications Magazine, February 1997
- [8] S. Rahkila, S. Stenberg, *Experiences on Building a Distributed Computing Platform Prototype for Telecom Network and Service Management*, Nokia research Center

- [9] Q. Kong and G. Chen, *Integrating CORBA and TMN Environments*, CiTR Technical Journal - Volume 1, presented at 5th IEEE/IFIP Network Operations and Management Symposium, April 1996, Kyoto, Japan.
- [10] G. Chen, Q. Kong and Michael Neville, *Distributed Network Management Using CORBA/TMN*, presented at 7th IEEE/IFIP International Workshop on Distributed Systems Operation and Management, DSOM'96, L'Aquila, Italy
- [11] L. Gys and H. Zuidweg, *Intelligence in the Network*, Alcatel Telecommunications Review - 1st Quarter 1998
- [12] Hewlett-Packard, *HP Openview Telecom*, <http://www.hp.com/ovtelecom>
- [13] B. Neumair, *Distributed Applications Management based on ODP Viewpoint Concepts and CORBA*, Munich Network Management Team, MNM
- [14] A. Gys, H. Zuidweg, *TINA Service Architecture*, Alcatel Telecommunications Review - 1st Quarter 1998
- [15] J. Pavón, J. Tomas, *CORBA for Networks and Service Management in the TINA Framework*, IEEE Communications Magazine, March 1998
- [16] Q. Kong, G. Chen and G. Holliman, *Telecommunications Service Management*, CiTR Technical Journal - Volume 1, presented at OpenView Forum conference, June 1995, Seattle, USA
- [17] S. Mazumdar, *Inter-domain Management - CORBA, OSI, SNMP*, presented at the IM'97, San Diego, USA, May 12-17, 1997.
- [18] Open Group - Inter-domain Management: Specification Translation 3/97
<http://www.opengroup.org/pubs/catalog/p509.htm>
- [19] OMG - CORBA/TMN Interworking RFP:
http://www.omg.org/techprocess/meetings/schedule/CORBA_TMN_Interworking_RFP.html
- [20] UH Communications ApS, *Q3ADE*:
<http://login.dknet.dk/~uh/q3c2cgtw/q3c2cgtw.html>
- [21] Inprise Corporation, *Visibroker – CORBA Technology from Inprise*:
<http://www.inprise.com/visibroker>
- [22] Sun Microsystems - *RMI Remote Method Invocation*:
<http://java.sun.com/docs/books/tutorial/rmi/index.html>
- [23] Sun Microsystems - *Creating a GUI with JFC/Swing*:
<http://java.sun.com/docs/books/tutorial/uiswing/index.html>