



## GMMS – Ambiente Gráfico E Cooperativo Para Gerência De Modelos

**Rodrigo Smarzaro da Silva<sup>1</sup>**

UNILINHARES

29901-350 – Linhares – ES

{smarzaro@soce.br}

**José Luis Braga**

Departamento de Informática

Universidade Federal de Viçosa

36571-000 – Viçosa - MG

{zeluis@mail.ufv.br}

### RESUMO

Apesar da necessidade de gerenciamento de modelos nas organizações, um número reduzido de tomadores de decisões utiliza alguma ferramenta para gerenciamento de modelos. Algumas razões para este paradoxo são a complexidade inerente aos modelos e a falta de um ambiente amigável para os tomadores de decisões. Neste trabalho, é proposto um ambiente gráfico onde os tomadores de decisão possam formular, gerenciar e executar modelos de forma simples e objetiva. O ambiente possui suporte à SML (*Structured Modeling Language*), como linguagem de definição de modelos, e à SQL-M, uma extensão da SQL (*Structured Query Language*) para gerenciamento de modelos. As duas linguagens são utilizadas de forma visual. Os modelos são criados, visualmente, através de grafos de modelos e as instâncias de modelos são criadas através de um assistente que auxilia o tomador de decisão em todo o processo. A execução do modelo é feita por resolvedores externos ao ambiente, mas tudo de forma transparente para o tomador de decisão. Todas estas características formam um ambiente, onde o tomador de decisão pode gerenciar os modelos necessários à organização de forma simples, sem a necessidade de conhecer profundamente uma linguagem de definição ou de manipulação de modelos.

### PALAVRAS-CHAVE

Gerência de Modelos, Sistemas de Apoio à Decisão, Modelagem Estruturada.

### ABSTRACT

Despite the need for model use and management in organizations, a small number of decision makers use some tool for that purpose. Some reasons for this paradox are model complexity and the lack of a friendly environment for model construction, testing and management tailored for decision makers. In this work, we present GMMS (Graphical Model Management System), a graphical environment in which decision makers can formulate, manage and execute models in a simple and straight fashion. The environment provides support for SML (Structured Modeling Language), a model definition language, and for SQL-M, that is an SQL (Structured Query Language) extension for model management. Both languages are used through an easy to use graphical language. Models are created visually by dragging and dropping icons from a modeling palette, and model instances are created using a wizard that helps the decision maker along the process. Models are executed by using solvers external to the environment, in a way that is fully transparent to the decision maker. GMMS provides a friendly and easy to use environment where decision makers can play with models without having a deep technical knowledge on model management. This leaves spare time to get more concentrated on the problem and its domain.

### KEYWORDS

Model Management, Decision Support Systems, Structured Modeling

---

<sup>1</sup> Trabalho desenvolvido a partir de dissertação de Mestrado em Ciência da Computação através do convênio DCC-UFMG/DPI-UFV com financiamento parcial da FAPEMIG.



## INTRODUÇÃO

Depara-se hoje com um volume de informações de complexidade muito maior do que se verificava no passado. Esta complexidade deve-se a vários fatores como aumento do volume de informações disponível, cenários de disputa globais, aumento da capacidade computacional e disseminação de redes e internet, competição mais acirrada e mais complexa entre empresas, entre outros. Neste ambiente, onde uma decisão pode determinar o sucesso ou o fracasso de uma empresa, faz-se necessária a utilização de ferramentas de auxílio na tomada de decisão, que permitam aos tomadores de decisão gerenciar essa complexidade.

Apesar de as empresas estarem utilizando cada vez mais a tecnologia de informação e de comunicações para facilitar o acesso aos dados, os gerentes raramente conseguem aproveitar toda informação disponível e, geralmente, utilizam-na apenas para tomar decisões simples. Ferramentas e ambientes, tais como Sistemas de Apoio a Decisão (SAD), Sistemas de Apoio a Decisão em Grupo (SADG), Sistemas de Informação Executiva (SIE), Sistemas Especialistas (SE), e Redes Neurais Artificiais (RNA) [Turban98] estão sendo crescentemente utilizados para encurtar essa distância.

Dentre estas, talvez a que tenha uso mais difundido dentro das empresas são os Sistemas de Apoio a Decisão. Os SAD são sistemas interativos, flexíveis e adaptáveis, baseados em computador, especialmente desenvolvidos para solucionar algum problema gerencial, com o objetivo de melhorar a tomada de decisão. Devem utilizar dados, possuir uma interface amigável, e são compostos basicamente por três subsistemas: Gerenciamento de Dados, Gerenciamento de Modelos e Gerenciamento de Comunicação (Interface). Um quarto subsistema opcional que pode fazer parte de um SAD é o de Gerenciamento de Conhecimento, caso em que o SAD será chamado de SAD inteligente, ou SAD baseado em conhecimento [Turban98].

O Gerenciamento de Dados e a Comunicação são tecnologias que já estão bastante consolidadas [Blanni93]. Atualmente, os responsáveis pelo Gerenciamento de Dados são os Sistemas de Gerenciamento de Banco de Dados (SGBD). Os SGBD atuais já permitem a fácil manipulação dos dados até mesmo por usuários não especialistas.

O Gerenciamento de Modelos surgiu da constatação de que os modelos também são informação estratégica e, sendo assim, deveriam ser gerenciados e receber atenção semelhante à dispensada aos dados e informações. Mas, ao contrário dos dados que possuem os SGBD para seu gerenciamento, o gerenciamento de modelos ainda não possui uma solução similar, universal. O grande problema é a complexidade que o gerenciamento de modelos envolve, geralmente necessitando de funcionários altamente especializados para que seja feito com eficiência [Blanni93], o que ocasiona o afastamento do principal utilizador dos modelos, que são os gerentes responsáveis pelas tomadas de decisões.

Os Sistemas de Gerenciamento de Modelos (SGM), cujo principal objetivo é tornar a organização e a execução dos modelos fácil e transparente para os usuários, da mesma maneira que os Sistemas de Gerenciamento de Dados atuam em relação aos dados [Blanni93], são uma primeira resposta ao desafio da gerência de modelos.

Um grande problema encontrado no desenvolvimento dos SGM é o formato de representação dos modelos. O resultado mais promissor neste aspecto, até o momento, é a Modelagem Estruturada, introduzida por Arthur M. Geoffrion [Geoffr87], que fornece uma forte base teórica para a concepção, representação e manipulação de uma grande variedade de modelos. Um de seus principais objetivos é permitir que trabalhos baseados em modelos sejam feitos com ganho de produtividade e aceitação por usuários não especialistas. Por este motivo, a Modelagem Estruturada tem sido utilizada como base para a construção de protótipos de ferramentas para gerenciamento de modelos.

Outra dificuldade encontrada nos SGM é a obtenção dos dados para a instanciação dos modelos. Geralmente os ambientes dos SGM possuem uma linguagem própria para o gerenciamento de modelos, e devem efetuar chamadas externas a um SGBD para obtenção dos dados [Elmasri00]. Para solucionar este problema [Pereir97] desenvolveu uma extensão da linguagem SQL, para uniformizar o tratamento entre modelos e dados, chamado SQL-M.

A proposta deste trabalho é o desenvolvimento de um ambiente gráfico para gerência de modelos que auxilie o gerente a formular, desenvolver e utilizar seus próprios modelos, ou modelos



pré-existentes, utilizando a Modelagem Estruturada como forma de representação dos modelos via os grafos de modelos (*Model Graphs*) e a SML (*Structured Modeling Language*), e utilizando a SQL-M para a manipulação e instanciação dos modelos.

Nesse ambiente, o usuário terá a seu dispor uma interface gráfica amigável para formulação e instanciação de modelos. Para fornecer uma maior segurança aos usuários, uma série de verificações de consistência são efetuadas. Utilizando os grafos de modelos, o usuário poderá formular visualmente suas classes de modelos, que depois podem ser instanciadas facilmente pelo uso da SQL-M.

Para a utilização da SQL-M, também são utilizados recursos gráficos semelhantes aos oferecidos pelo QBE (*Query By Example*) [Elmasri00], usado para facilitar o uso de SQL nos SGBD. Nesta versão de QBE, a interface do modelo é reconhecida e o usuário poderá então fornecer o conjunto de dados que servirá de entrada para criar a instância do modelo.

Para demonstrar a validade do ambiente proposto, foi implementado o GMMS (*Graphical Model Management System*), que permite a criação e instanciação de modelos de forma gráfica, facilitando sua utilização por usuários não especialistas.

O restante deste artigo está organizado da seguinte maneira. No item dois será mostrado um levantamento dos pr.8( )]Telos.ce



permita, de certa forma, brincar com o problema, fazer simulações, alterar parâmetros e condições, traçando os possíveis cenários de decisão que terá que enfrentar. Desta constatação, surgiu a motivação para investir no desenvolvimento de uma ferramenta para gerência de modelos que preencha esta lacuna deixada pelos sistemas atuais.

Os requisitos relacionados a seguir foram estabelecidos a partir da observação dos sistemas existentes:

- **Abordagem gráfica** – é obrigatória uma abordagem gráfica para a formulação dos modelos e para o acesso aos dados. Através da utilização de uma interface gráfica, a capacidade de aprendizado dos usuários novatos aumenta consideravelmente, como já foi comprovado através de dados experimentais [Turban98]. Além da melhoria no aprendizado, os usuários novatos também se beneficiam em vários outros quesitos de performance como: maior rapidez e melhor qualidade no trabalho, alta produtividade e menor fadiga. Outro ponto importante é que, através da interface gráfica, a necessidade de conhecimento profundo da sintaxe de uma linguagem de definição de modelos, ou de uma linguagem de consulta a dados fica bastante reduzida, permitindo, deste modo, que os gerentes possam utilizá-la com segurança.
- **Poder de expressão** – Nem todas as necessidades de decisão de uma organização podem ser atribuídas a apenas uma determinada área de conhecimento. Portanto, um SGM deve ser baseado em uma linguagem que tenha grande capacidade de expressão, de modo que modelos das mais diferentes áreas de conhecimento possam ser representados.
- **Uniformidade de tratamento** – é desejável em um SGM que os dados e modelos sejam tratados de maneira uniforme. Portanto, a ferramenta deve fornecer mecanismos para que o gerente possa manipular dados e modelos de maneira sistemática e semelhante, do ponto de vista de manipulação e uso.
- **Suporte ao ciclo de vida de modelagem** – A ferramenta deve ser capaz de atuar nas diversas etapas no ciclo de vida dos modelos, ou seja, deve oferecer mecanismos para simplificar os processos de criação, consulta, execução e gerência dos modelos.
- **Independência entre modelos, dados e resolvedores** – os gerentes devem ser capazes de criar diferentes instâncias, a partir de uma classe de modelos e cada instância pode ser executada utilizando diferentes resolvedores. Deste modo, a ferramenta consegue a flexibilidade necessária para as necessidades dos gerentes dentro das organizações.
- **Auxílio ao usuário** – Como o processo de modelagem é complexo, a disponibilidade de alguma forma de auxílio ao usuário durante a utilização da ferramenta é bastante desejável e necessária. Tal auxílio pode ser feito através do monitoramento das ações do usuário, impedindo ou alertando sempre que alguma anormalidade for detectada.

Para solucionar a questão do poder de expressão da ferramenta, foi escolhida a SML [Geoffr87] (*Structured Modeling Language*) devido ao seu grande poder de expressão e algumas outras características. Além de ser uma linguagem formal, possui formas de representação gráficas próprias para os modelos, através dos grafos de *genus* (o *genus* é semelhante à noção de conjunto, ou classe), e de acordo com sua proposta, vem se tornando uma *lingua franca* para os SGM, uma vez que já existem diversos sistemas funcionando baseados em Modelagem Estruturada. A tendência é que os diversos sistemas possam interagir utilizando a SML como linguagem comum, no futuro.

Quanto à uniformidade de tratamento, a solução encontrada foi a utilização da linguagem SQL-M como linguagem de manipulação de modelos. Com a utilização da SML e da SQL-M, a ferramenta aqui apresentada suporta todo o ciclo de vida da modelagem, pois, através da SML, os modelos podem ser definidos, e, através da SQL-M, os modelos podem ser gerenciados, consultados e executados.

Tanto a SML, quanto a SQL-M permitem a independência entre modelos, dados e resolvedores. Na ferramenta, o modelo é criado utilizando SML, e as instâncias são criadas utilizando SQL-M. A mesma classe de modelo, criada em SML, pode possuir inúmeras instâncias com diferentes



dados através da SQL-M. Da mesma maneira, cada instância criada pode ser executada em mais de um resolvidor.

Com o intuito de facilitar a utilização do sistema, foi desenvolvida uma interface gráfica que permite a criação e manipulação de modelos, de forma que os usuários possam utilizar o sistema mesmo que não conheçam profundamente a sintaxe das linguagens. Desta maneira, um dos problemas identificados em sistemas de gerência de modelos pode ser suavizado, pois o modelo resultante não se tornará tão incompreensível e intimidador para o gerente, quanto o seria sem uma representação gráfica.

Além dos elementos gráficos envolvendo a ferramenta, o auxílio ao usuário é outro fator importante para o sucesso de uso pelos gerentes. Um dos tipos de auxílio identificado foi a checagem direta de sintaxe durante a formulação dos modelos. A checagem direta de sintaxe garante que os modelos formulados estejam sintaticamente corretos, eliminando assim alguns problemas que poderiam ocorrer durante a utilização de usuários menos experientes em modelagem.

Para viabilizar a proposta de tal ferramenta, foi desenvolvido um sistema chamado **GMMS** (*Graphical Model Management System*) que implementa as características citadas, de forma que o sistema resultante seja de fácil utilização por gerentes com um mínimo de treinamento.

### 3 – GMMS

O GMMS é um ambiente gráfico para gerência de modelos. Permite formulação, consulta, instanciação e composição de modelos utilizando uma interface gráfica baseada na Modelagem Estruturada, através da utilização de grafos de modelos, que são grafos direcionados acíclicos similares aos grafos de genus, como forma de representação gráfica para os modelos [Geoffr87]. Além de suporte à Modelagem Estruturada, o GMMS possui suporte para a SQL-M [Pereir97]. Este suporte se dá via entrada de comandos em SQL-M ou pelo uso de uma interface gráfica desenvolvida para a SQL-M, baseada no estilo QBE (*Query By Example*) [Elmasr00], onde o usuário poderá facilmente criar instâncias dos modelos existentes.

O GMMS se diferencia dos demais sistemas pela interface mais amigável e atual que auxilia o usuário em suas tarefas. Essa interface conta com a ajuda de um componente de conhecimento que fiscaliza as operações realizadas pelo usuário e o interrompe sempre que detectar alguma decisão errada ou alguma inconsistência na seqüência de uso do GMMS. Outro diferencial do GMMS é sua maior integração com a Modelagem Estruturada. A proposta do GMMS é a utilização da Modelagem Estruturada tanto para a representação gráfica dos modelos, utilizando os grafos de modelo, quanto para a representação interna, utilizando SML, de tal maneira que dado um modelo representado graficamente, pode-se obter seu equivalente em SML através de um mapeamento da linguagem gráfica para a linguagem textual.

Todas as informações sobre os modelos ficam armazenadas em um banco de dados relacional para permitir a fácil utilização dos modelos por sistemas externos como planilhas, e resolvidores. Os resolvidores poderão acessar as Tabelas de Detalhamento de Elementos (*Elemental Detail Tables*) das instâncias dos modelos, ou sua tradução no formato específico do resolvidor para conseguir os dados necessários para executar o modelo. Os dados gerados pelo resolvidor são retornados para o usuário do GMMS de forma simples e clara.

#### 3.1 - Arquitetura

A arquitetura proposta para o GMMS é mostrada na Figura 3.1. Podem ser vistos os principais subsistemas que compõem o GMMS, exibidos como elipses com fundo cinza, e suas interações.

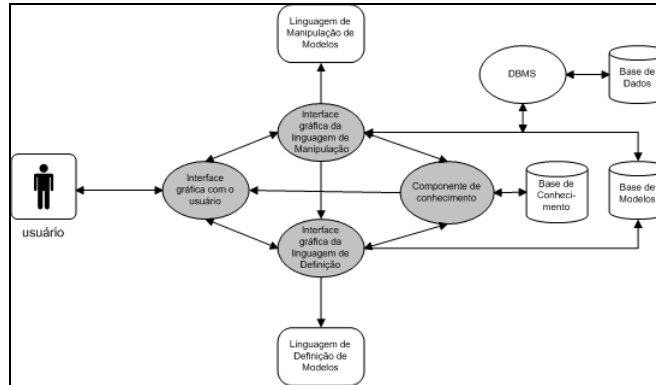


Figura 3.1 – Arquitetura do GMMS

### 3.1.1 – Interface Gráfica da Linguagem de Definição de Modelos

Este subsistema é o responsável pela ligação entre a Linguagem de Definição de Modelos, a SML, e o GMMS. Para que o gerente possa construir um modelo graficamente, foi necessário projetar uma representação gráfica para a SML, de forma que a linguagem gráfica resultante não perdesse seu poder de expressão. Como a modelagem estruturada já provê mecanismos de representação gráfica para os esquemas de modelos, foi feita uma adaptação para que o modelo gráfico, além de fornecer a informação sobre a estrutura do modelo, também forneça as informações mais detalhadas sobre cada elemento que compõe o modelo. Esta adaptação foi feita através de grafos de modelos, para representar a estrutura do modelo, onde cada elemento do modelo pode ter suas características fornecidas em formulários próprios para cada tipo de elemento.

Os elementos da SML foram representados como formas geométricas simples, mostradas na Figura 3.2, onde um quadrado representa uma entidade primitiva (pe), um retângulo representa uma entidade composta (ce), um círculo representa uma variável (a), um triângulo representa uma função (f), e um losango representa um teste (t).

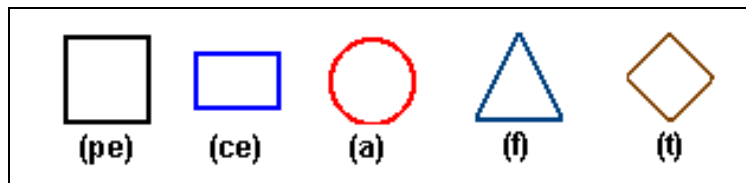


Figura 3.2 – Correspondência entre os símbolos e os tipos de elementos da Modelagem Estruturada

Um esquema de modelo em SML, como o mostrado na Figura 3.3, pode ser representado graficamente pelo diagrama mostrado na Figura 3.4, gerado usando o GMMS.

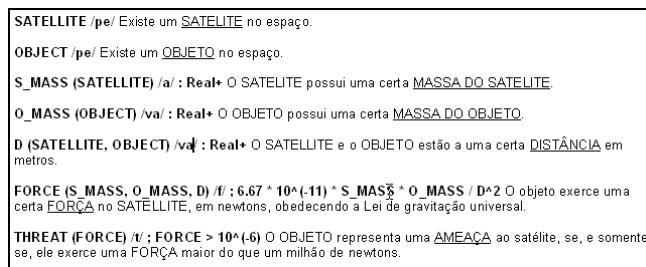


Figura 3.3 – Esquema de modelo SATELLITE [Geoffr92a]

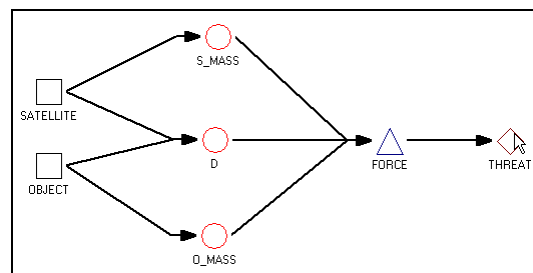


Figura 3.4 – Representação gráfica para o modelo SATELLITE



### 3.1.2 – Interface Gráfica da Linguagem de Manipulação de Modelos

A interface gráfica para a linguagem de manipulação de modelos é responsável pela ligação entre a linguagem SQL-M e o GMMS. Através dela, o usuário pode criar instâncias de modelos, simplesmente fornecendo dados para suas variáveis de entrada. Também é possível executar a instância de modelo criada, ou editar alguma instância criada anteriormente. Outra responsabilidade deste subsistema é a ligação entre o GMMS e os resolvedores. As instâncias criadas possuem um resolvidor padrão, e o resolvidor é capaz de reconhecer os dados necessários para a execução da instância do modelo. A saída gerada pelo resolvidor é reconhecida pelo GMMS, que as repassa ao usuário de maneira simples e clara.

Esse subsistema também possui uma ligação com o subsistema de linguagem de definição de modelos. Tal ligação deve-se ao fato de que a linguagem de manipulação deve reconhecer a interface do modelo, ou seja, as variáveis de entrada e saída bem como seus respectivos tipos de dados.

Deste modo, o trabalho necessário para o usuário criar uma instância completa do modelo é: fornecer o caminho, quais dados devem ser utilizados, especificar o resolvidor e indicar qual é a classe de modelo sendo instanciada. O fornecimento dos dados é realizado de maneira visual, utilizando ferramentas para a construção de consultas SQL, sem que seja necessário que o usuário conheça profundamente sua sintaxe. Ele deve simplesmente fornecer a tabela onde os dados se encontram, selecionar os campos necessários e fornecer algum critério de seleção caso seja necessário. Este processo deve ser repetido para cada variável de entrada que o modelo possuir. Assim, cada variável terá uma consulta SQL associada, indicando os dados necessários para a criação da instância do modelo.

Diferentemente do GBMS/SM, onde também é possível associar consultas SQL para a obtenção dos dados, o GMMS não associa as consultas SQL à classe de modelos criada, podendo existir várias instâncias do mesmo modelo utilizando consultas SQL diferentes. No GBMS/SM, seria necessário sempre modificar as consultas SQL para criar uma nova instância, sobrepondo-a à anterior que é perdida, ou então copiar o modelo inteiro e associar novas consultas SQL.

Com a consulta pronta, a instância pode ser enviada para solução em algum resolvidor. Mas, diferente do que acontece na implementação proposta por [Pereir97], onde o resultado da execução era retornado como uma tabela contendo uma única linha, e onde as colunas eram as variáveis selecionadas pela consulta, no GMMS os resultados do resolvidor são direcionados para as respectivas variáveis de saída. Deste modo, quando o usuário desejar verificar o resultado de alguma execução, basta selecionar a variável desejada que o seu valor atual será exibido. Assim, o resultado gerado pelo resolvidor pode ser consultado mais facilmente pelo usuário.

### 3.1.3 – Componente de Conhecimento

O componente de conhecimento centraliza todas as informações necessárias para o auxílio que o GMMS pode dispensar ao usuário. Possui como principais funções:

- Fornecer e verificar as regras para a checagem direta de sintaxe durante a construção do modelo, segundo a Modelagem Estruturada, que estabelece as ligações válidas entre os diferentes tipos de elementos que compõem um modelo. Deste modo, o GMMS assegura que os modelos criados graficamente sempre serão sintaticamente corretos. Estas regras podem ser vistas na Tabela 3.1.
- Atuar na checagem de compatibilidade entre os tipos e dimensões de dados que o usuário fornece para as variáveis de entrada do modelo. Caso alguma incoerência seja detectada, o usuário é avisado do erro, e o componente de conhecimento indica uma possível solução.
- Gerenciar o Assistente de criação de nova instância, que é uma seqüência de passos, guiada pelo GMMS e acompanhada pelo usuário, onde o produto final é uma instância de modelo completa. O Assistente entra em funcionamento quando o usuário desejar criar uma nova instância, a partir de alguma classe de modelo existente. Ao final de sua execução, o GMMS pode exibir a instância do modelo na tela de edição de



instâncias (Figura 3.7), permitindo ao usuário realizar modificações, executar o modelo, ou simplesmente salvar a instância criada para utilização futura. Segue três etapas distintas:

1. Definição das características da instância, tais como nome, descrição e o resolvidor que deverá ser utilizado para a execução.
2. Identificação da interface do modelo, selecionando as variáveis para entrada e saída.
3. Definição da fonte dos dados das variáveis de entrada.

Nodo Destino		Possíveis Nodos Origem
Entidade Primitiva	(pe)	Nenhum
Entidade Composta	(ce)	pe ou ce
Atributo	(a)	pe ou ce
Função	(f)	pe, ce, a, f ou t
Teste	(t)	pe, ce, a, f ou t

**Tabela 3.1 - Restrições de relacionamentos de nodos nos grafos de modelos**

### 3.2.4 – Interface gráfica com o usuário

O subsistema de interface gráfica com o usuário integra os demais subsistemas, tornando todos os processos e tarefas transparentes para o usuário. Todo o controle de ações disponíveis que o usuário pode realizar é centralizado na interface gráfica, que utiliza menus simples, menus *pull-down*, barras de ferramentas, ícones, caixas de listagem, mouse, e outros conhecidos recursos gráficos do ambiente Windows.

Na Figura 3.4 são mostradas, além da tela principal do GMMS (a), as telas responsáveis pela navegação da base de modelos (b), a tela de formulação gráfica dos modelos (c) e cada uma das telas utilizadas na inserção dos nodos no diagrama, como Entidade Primitiva (d), Entidade Composta (e), Atributo (f), Função (g) e Teste (h).

Na Figura 3.5 é mostrada a execução do Assistente de nova instância, que segue uma sequência de três etapas. Na primeira etapa deve ser fornecido ao Assistente informações gerais sobre a nova instância que está sendo criada, como seu nome, descrição e resolvidor padrão que será utilizado. Na segunda etapa é definida a interface da instância selecionando quais variáveis são de entrada e quais serão de saída. Na terceira etapa, deverão ser fornecidos dados para as variáveis de entrada do modelo. Para realizar esta tarefa, existem duas opções válidas. A primeira é fornecer os dados diretamente. A segunda é fornecer uma consulta SQL, indicando onde e quais os dados devem ser utilizados. Caso a segunda opção seja utilizada, o GMMS apresentará a tela de criação visual de consultas SQL (Figura 3.6), onde as consultas podem ser facilmente criadas, de maneira visual, semelhante às ferramentas QBE. Assim que uma consulta for criada, ela é repassada ao assistente que deverá verificar se ela é válida (os tipos de dados da consulta, e da variável devem ser compatíveis).



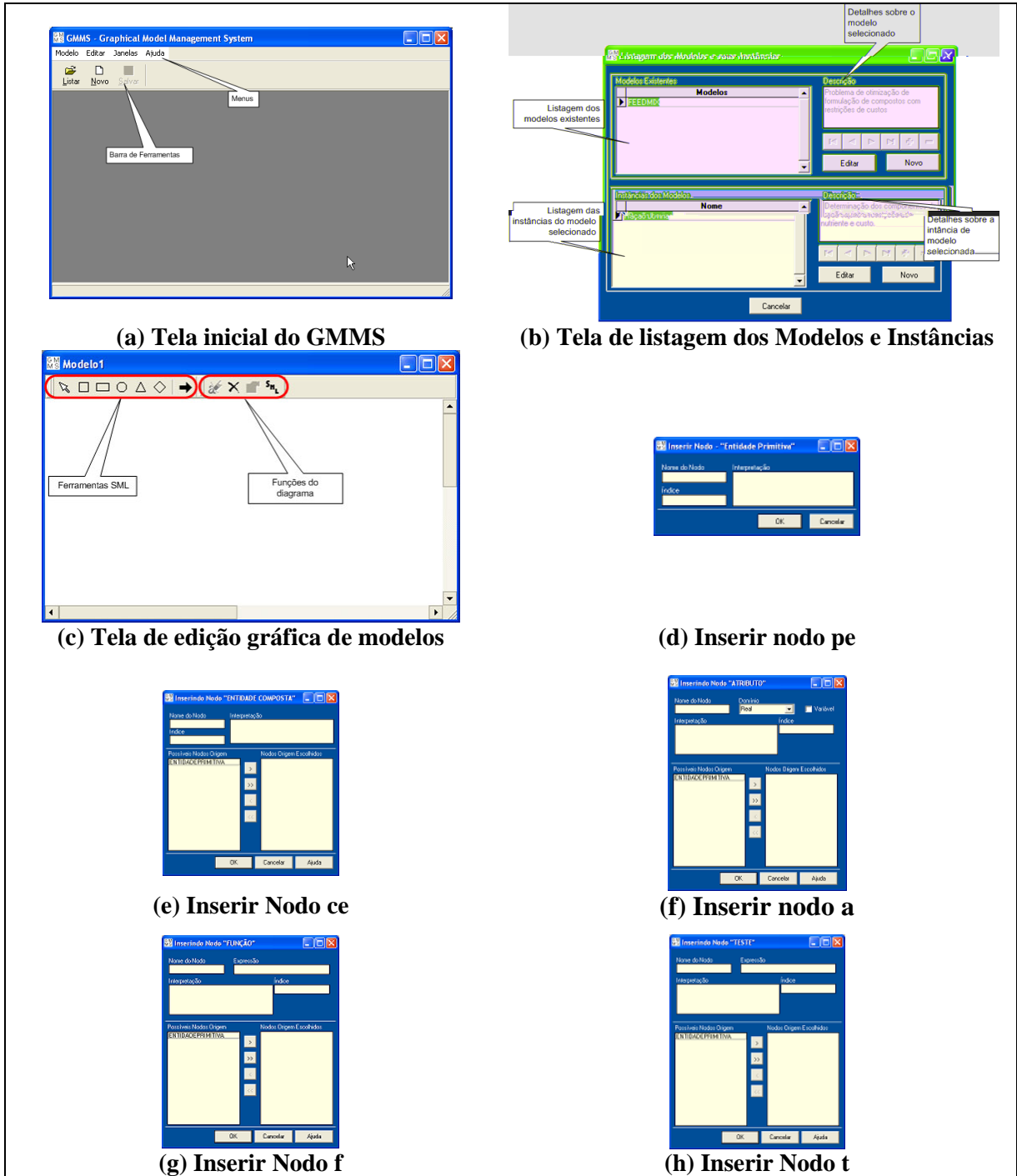


Figura 3.4 – Interface da modelagem gráfica de modelos

O assistente termina sua execução assim que as variáveis de entrada possuírem dados válidos. O GMMS exibe então a tela de edição de instância (Figura 3.7). Nesta tela, é possível visualizar os dados preenchidos durante a execução do assistente, modificar estes dados, ou então executar a instância utilizando o resolvedor que foi fornecido na primeira etapa do assistente.

Ao executar a instância, o GMMS se comunica com o resolvedor, passando o modelo e os dados instanciados, o resolvedor executa os cálculos necessários e devolve as soluções encontradas para o GMMS, que se encarrega de instanciar as variáveis de saída com os dados recebidos.

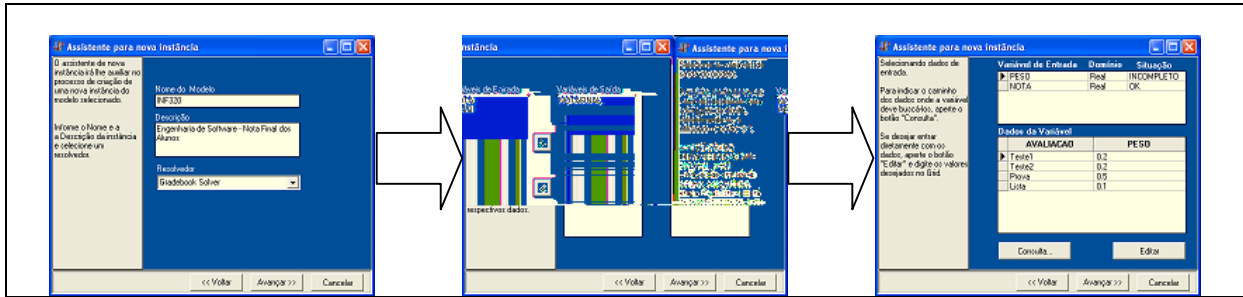


Figura 3.5 – Etapas do assistente para nova instância

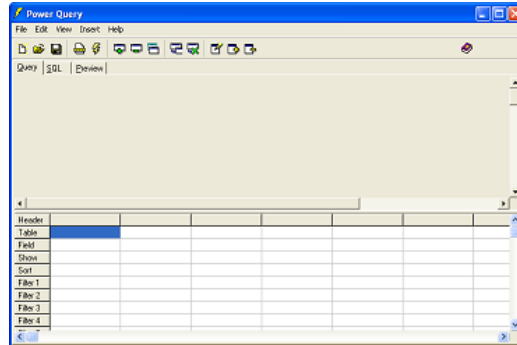


Figura 3.6 – Construção visual de consultas SQL

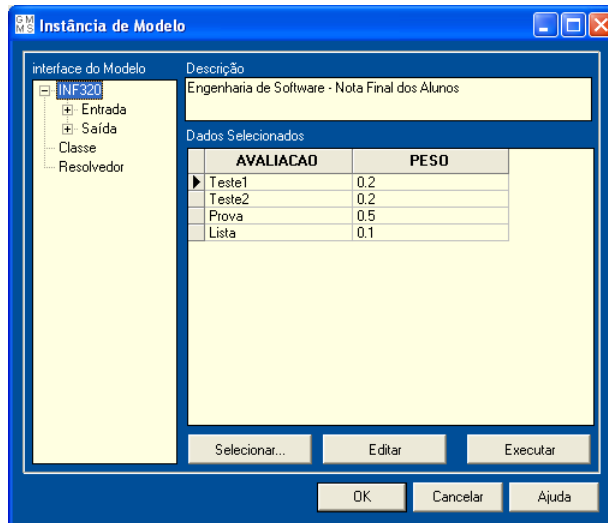


Figura 3.7 – Tela de edição das instâncias

#### 4. Conclusões

Fornecendo um único ambiente onde os modelos podem ser definidos, instanciados e executados, o ambiente gráfico proposto pode auxiliar os usuários de Sistemas de Apoio a Decisão nas tarefas de formulação e manipulação dos dados de maneira simples e visual. Apresenta-se como uma alternativa viável para grande parte dos objetivos desejados em um Sistema de Gerência de Modelos, fornecendo recursos para facilitar a manipulação de modelos e dados, integração com banco de dados, apoio nas tarefas essenciais como formulação e instanciação dos modelos, independência entre modelos, dados e resolvedores, e uma interface gráfica que permite sua utilização por usuários não especialistas.



Um sistema com as características do GMMS pode ser utilizado de diferentes maneiras no ambiente das organizações. A arquitetura aberta do GMMS permite que sejam elaboradas versões personalizadas da base de modelos e de resolvedores para organizações de diferentes escopos. Por exemplo, uma indústria alimentícia pode possuir uma versão do GMMS com modelos tais como análise de custos, depreciação de equipamentos, mix de produção e os resolvedores apropriados para cada modelo. Esta abordagem facilitaria ainda mais a utilização efetiva do sistema por usuários não especialistas, pois diminuiria, ou possivelmente elimina o processo de formulação dos modelos, que é a etapa mais especializada no ciclo de vida da modelagem.

Outra possível utilização do GMMS é a sua utilização como interface mais intuitiva e amigável para ambientes de modelagem algébricos, fornecendo uma visualização gráfica de modelos descritos nesta notação.

Algumas facilidades ainda podem ser implementadas no GMMS para facilitar ainda mais sua utilização, como uma biblioteca de resolvedores mais elaborada, permitindo consultas mais refinadas sobre os possíveis resolvedores que podem solucionar um determinado modelo. Uma possível abordagem para a construção destas consultas seria associá-las a uma ontologia sobre o domínio de conhecimento de modelos e resolvedores.

Seguindo o mesmo raciocínio, também seria interessante apoiar a consulta de modelos em ontologias para responder perguntas do tipo “Que modelo resolve o algum determinado problema?”.

Como visto, o GMMS possui diferentes possibilidades de utilização, e através de seu uso efetivo no ambiente de organizações, espera-se que ele se torne uma ferramenta de grande valor no apoio às decisões de gerentes.

## Bibliografia

- [Blanni93] Blanning, R.W., King, D.R. “Introduction: The Growth of Decision Support Technology”, *Current research in decision support technology*, IEEE Computer Society Press, p1-8, 1993.
- [Chari97a] Chari, K., Sen, T. K., “An Integrated Modeling System for Structured Modeling Using Model Graphs”, *INFORMS Journal on Computing*, Volume 9, No. 4, 1997.
- [Dolk88] Dolk, D. R., “Model management and Structured Modeling: the Role of an Information Resource Dictionary Systems”, *IEEE Computer* 17, 9 (September 1984), 89-97.
- [Elmasr00] Elmasri, R., and S. B. Navathe, “Fundamentals of Database Systems”, Benjamin/Cummings, Redwood, CA, 2000.
- [Geoffr87] Geoffrion, A.M., “An Introduction to Structured Modeling”, *Management Science*, Vol. 33, No. 5, Maio, p547-588, 1987.
- [Hamach93] Hamacher, S., Dejax, P., Lustosa, L., e Hamacher, P., “A Diagram Representation for conceptual Models of Operations Research Problems”, *Cahiers d’Etudes et de Recherche No. 93-04A*, Laboratoire Productique Logistique, Ecole Centrale Paris, France, 1993.
- [Jones92] Jones, C. V., “Attributed Graphs, Graph-Grammars, and Structured Modeling”, *Annals of Operations Research*, n.38, p281-324
- [Pereir97] Pereira, D. A. “Extensões à Linguagem SQL para Gerenciamento de Modelos”, *Dissertação de Mestrado*, UFMG, Belo Horizonte, Março, 1997.



- [Turban98] Turban, E. e Aronson, J. E. “Decision Support Systems and Intelligent Systems”.  
*Prentice Hall*, Englewood Cliffs, New Jersey, 1998.