

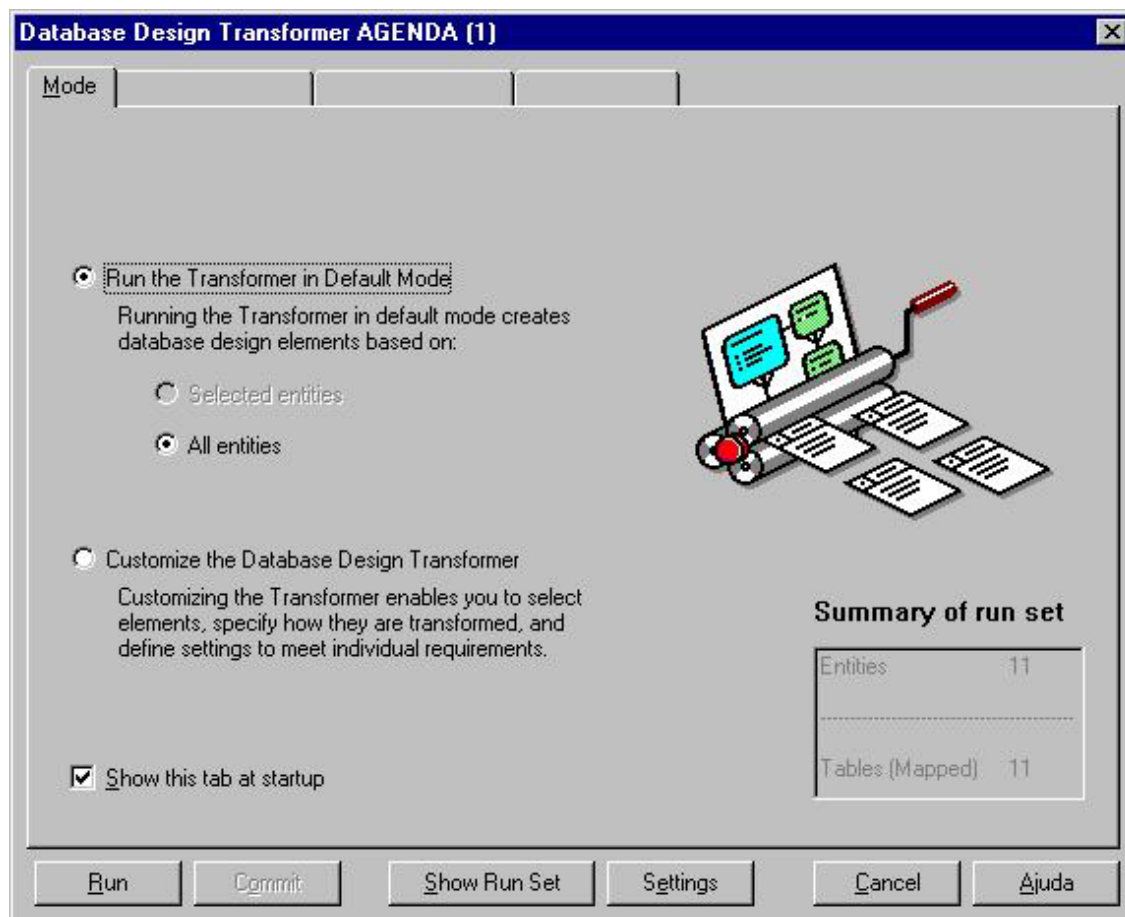
2 - Mapeamento do Modelo Conceitual para o Modelo Físico

O mapeamento do Modelo Conceitual para o Modelo Físico representa a fase de transição entre as fases de Planejamento para a fase de Projeto.

Para mapearmos o Modelo Conceitual de Dados utilizaremos a ferramenta Database Design Transformer e para mapearmos o Modelo Conceitual de Funções utilizaremos a ferramenta Application Design Transformer.

2.1 - Mapeamento do Modelo Conceitual de Dados

O mapeamento do Modelo Conceitual de Dados implica na tradução de todo o modelo E-R criado na modelagem conceitual para um projeto do banco de dados.

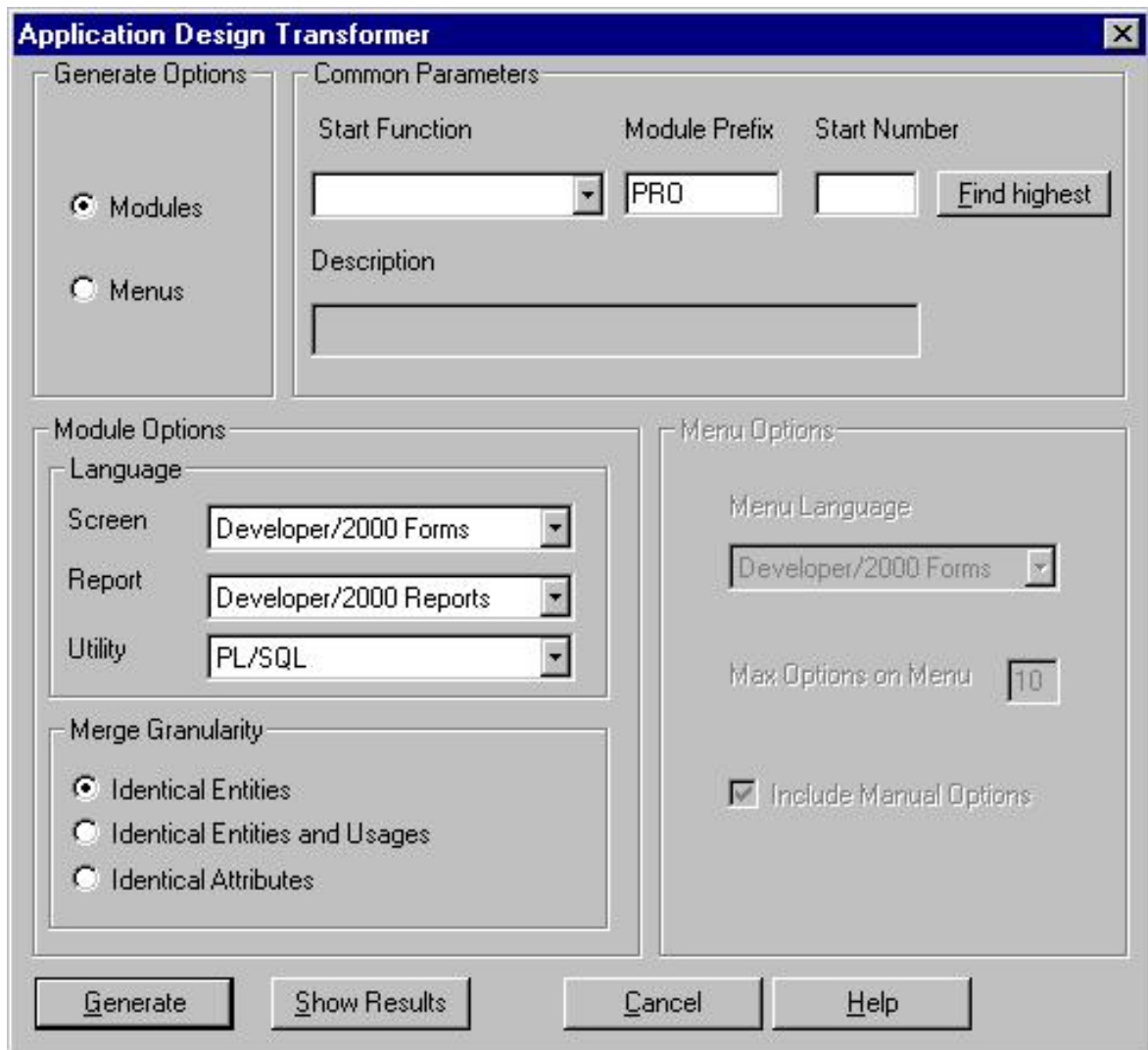


Ao acionarmos a execução do Database Design Transformer, teremos a tela acima.

Acionando a opção *Run*, se iniciará o processo de mapeamento do modelo E-R conceitual para o modelo E-R da fase de Projeto do Banco de Dados.

2.2 - Mapeamento do Modelo Conceitual das Funções

O mapeamento do Modelo Conceitual das Funções implica na tradução de todo o modelo hierárquico de funções criado na modelagem conceitual para um modelo hierárquico de módulos, representando os módulos do sistema.



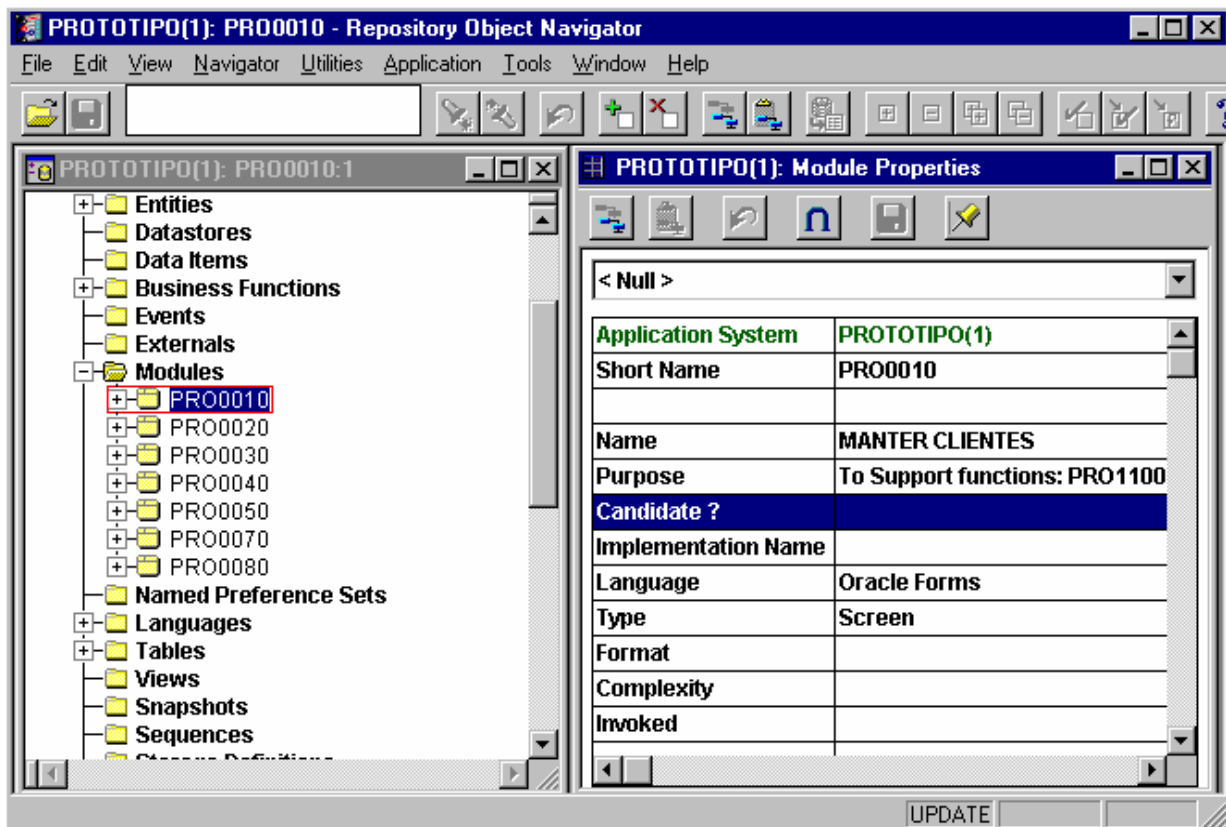
Ao executarmos o Application Design Transformer teremos a tela acima.

Devemos indicar a partir de qual função desejamos mapear, digitando o apelido (short name) da função. Para gerar todas as funções indique a função PRO10000. Na cláusula *Module Prefix* deve ser indicado um prefixo para o nome do módulo que será gerado. Será anexado a este prefixo um número, identificando o módulo no repositório.

Na cláusula *Generate Options* devemos indicar se vai ser gerado os módulos (tela, relatórios), ou os menus. A cláusula *Merge Granularity* indica que parâmetro utilizar para identificar as funções idênticas no negócio. Caso a ferramenta identificar duas funções idênticas, somente um módulo será gerado. Este parâmetro pode ser:

- Identical Entities - Quando duas funções tiverem entidades idênticas
- Identical Entities and Usages - Quando duas funções tiverem entidades e seu uso idênticos
- Identical Attributes - Quando duas funções tiverem atributos idênticos

O módulos gerados são módulos candidatos, para torná-los módulos efetivos do sistema, devemos acionar a ferramenta **Repository Object Navigator**, abrir a aplicação e, no ítem *Modules* selecionar cada módulo e modificar a propriedade *Candidate* para *No*.



Após gerados e aceito os módulos de dados, devemos gerar os módulos que representarão os itens de menu. Para tanto basta alterar a cláusula *Generate Options* para a opção *Menus* e os módulos de menu serão gerados. Após gerados estes deverão também ser aceitos pela ferramenta **Repository Object Navigator**.

Esta '*formalidade*' de aceitar os módulos ocorre porque num sistema com muitas funções, podem ocorrer casos em que o usuário não queira transformar aquela função em um módulo da aplicação, mas deseja documentá-la no repositório de dados.

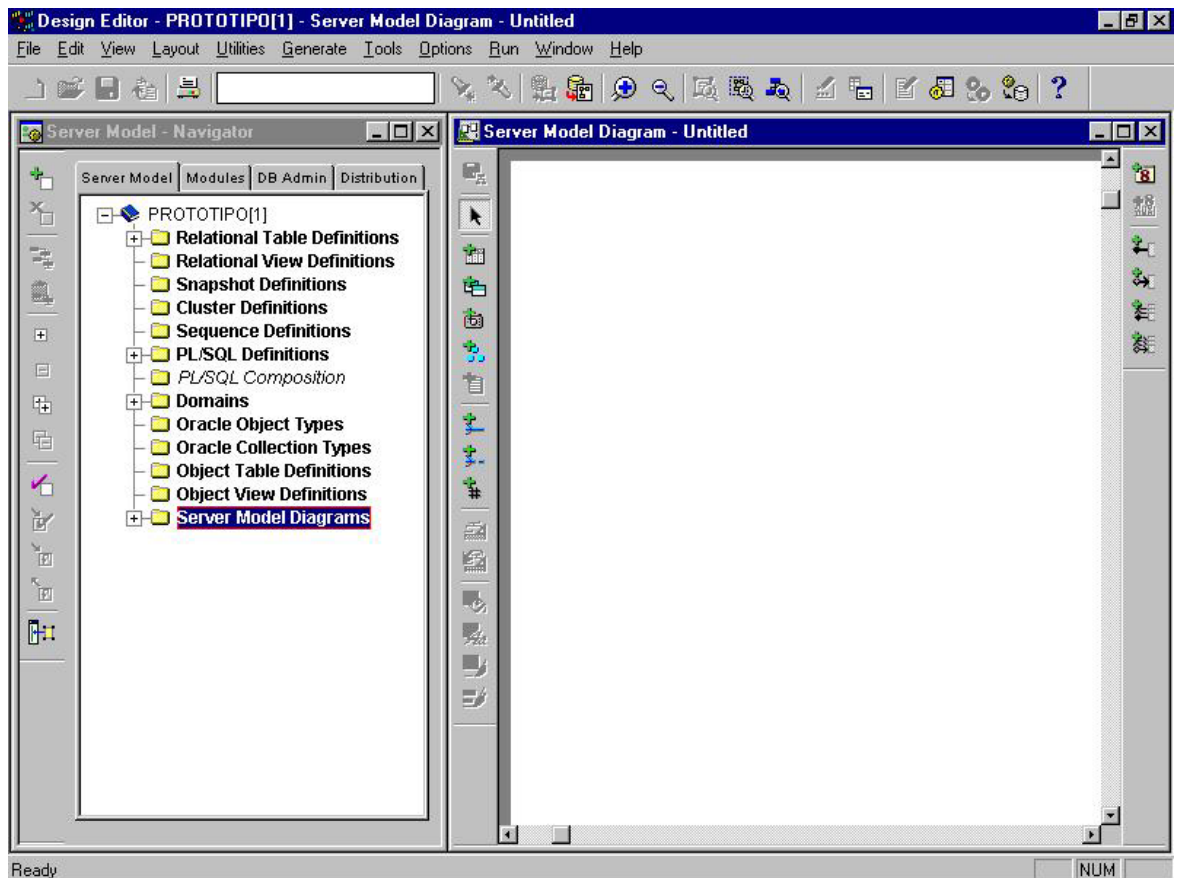
3 - Projeto do Banco de Dados e da Aplicação

Na Fase de Projeto do Banco de Dados e da Aplicação, iremos informar ao repositório de dados todas as informações complementares para possibilitar a geração da Base de Dados e dos módulos da aplicação, para isto utilizaremos algumas ferramentas que serão descritas a seguir.

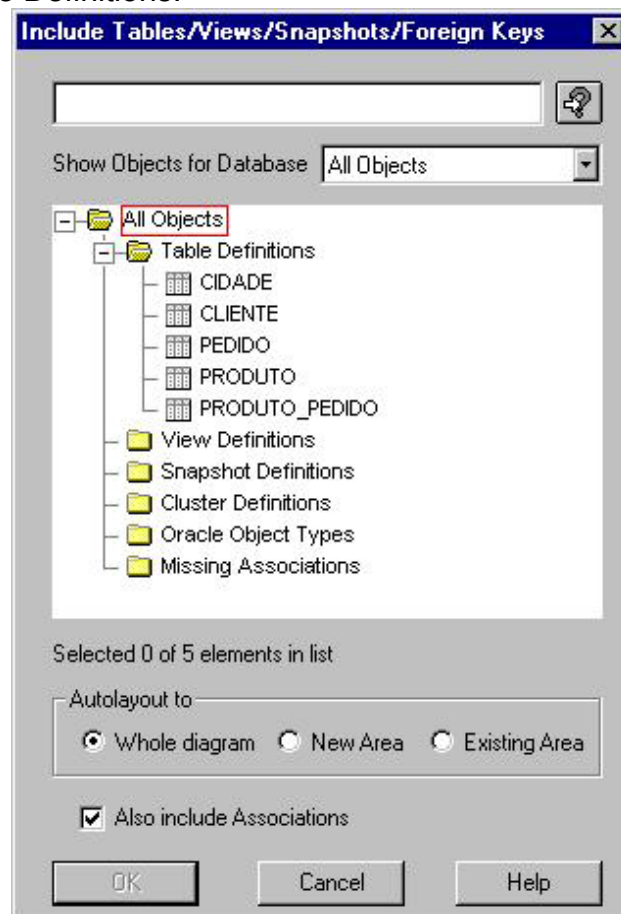
3.1 - Completando o Banco de Dados

Para consolidar o projeto do banco de dados, utilizaremos a ferramenta Design Editor.

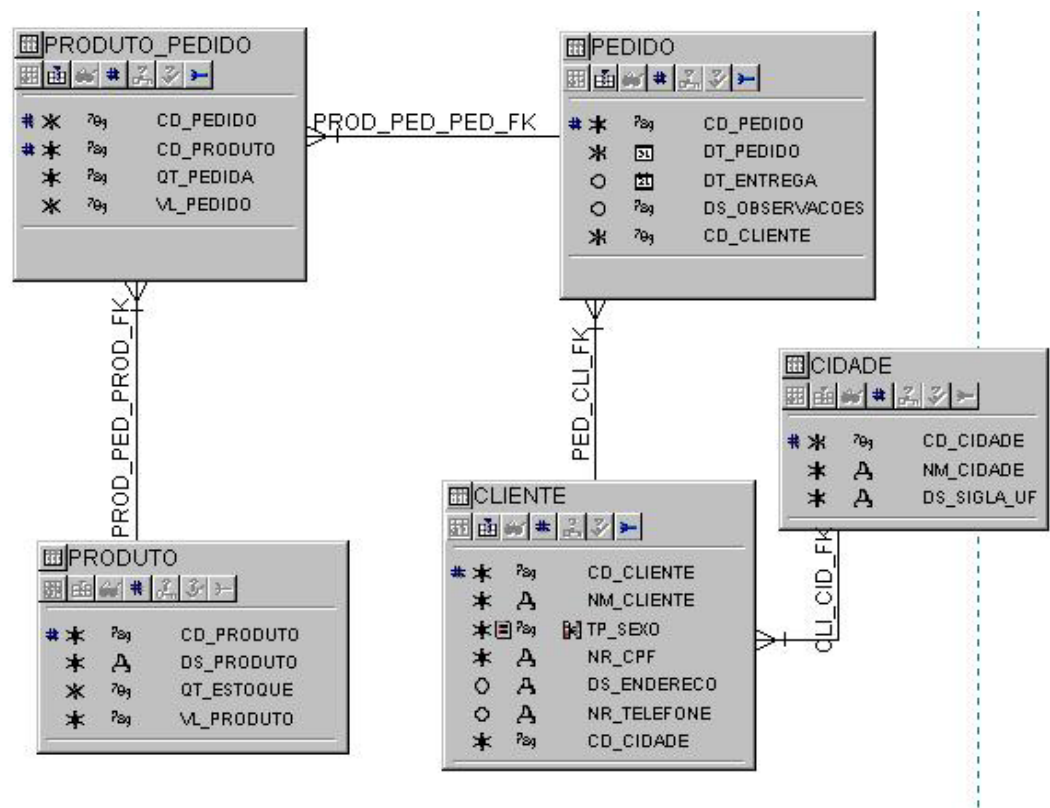
Executando-a devemos abrir a aba *Server Model* e selecionar a opção *Server Model Diagrams*.



Em seguida selecione o menu EDIT → Create. Um novo documento irá surgir e nele devemos incluir os objetos criados pelo mapeamento dos dados, realizando através do menu Edit → Include → Table Definitions.



Esta é a lista de objetos que devemos incluir no modelo. Basta selecionarmos todos os itens e acionar o botão OK.



Após incluídos os objetos, teremos o novo modelo E-R acima exposto. Aqui estão dispostos todos os objetos previstos na modelagem conceitual dos dados, tais como chave primária, chave estrangeira e índices.

Conforme a necessidade, podemos completar o modelo construindo mais índices, relacionamentos, tabelas,... enfim, modificando o modelo inicial afim de obtermos um modelo completo para o banco de dados.

A seguir teremos uma visão geral dos dados manipulados nesta ferramenta.

Tomamos como exemplo a tabela CLIENTE para completarmos o projeto de Banco de Dados, afim de explanarmos a potencialidade desta ferramenta.

Edit Table CLIENTE

Name Columns Display Controls UI

The minimum information required for a table is the name. The other naming conventions listed are used as defaults in the generated application.

Table name: CLIENTE

Table alias: CLI

Display title: Cliente

New column prefix:

Buttons: Cancel, Help, < Back, Next >, Apply, Finish

Editando a tabela CLIENTE teremos a tela acima, onde podem ser complementados os seguintes dados:

Guia Name: Este guia trata de dados referentes a definição da tabela, tais como: nome e apelido da tabela.

Edit Table CLIENTE

Name Columns Display Controls UI

Column properties entered here will be used as default values when the table is used by a generated application. You can create a valid list of values for a column, apply an existing domain to a column and create a new domain by entering a new name.

Column name: CD_CLIENTE

- CD_CLIENTE
- NM_CLIENTE
- TP_SEXD
- NR_CPF
- DS_ENDERECO
- NR_TELEFONE
- CD_CIDADE

Column Type: Scalar

Domain: <NONE>

Datatype: NUMBER

Length: 5, No. of Decimal places: 0

Default value:

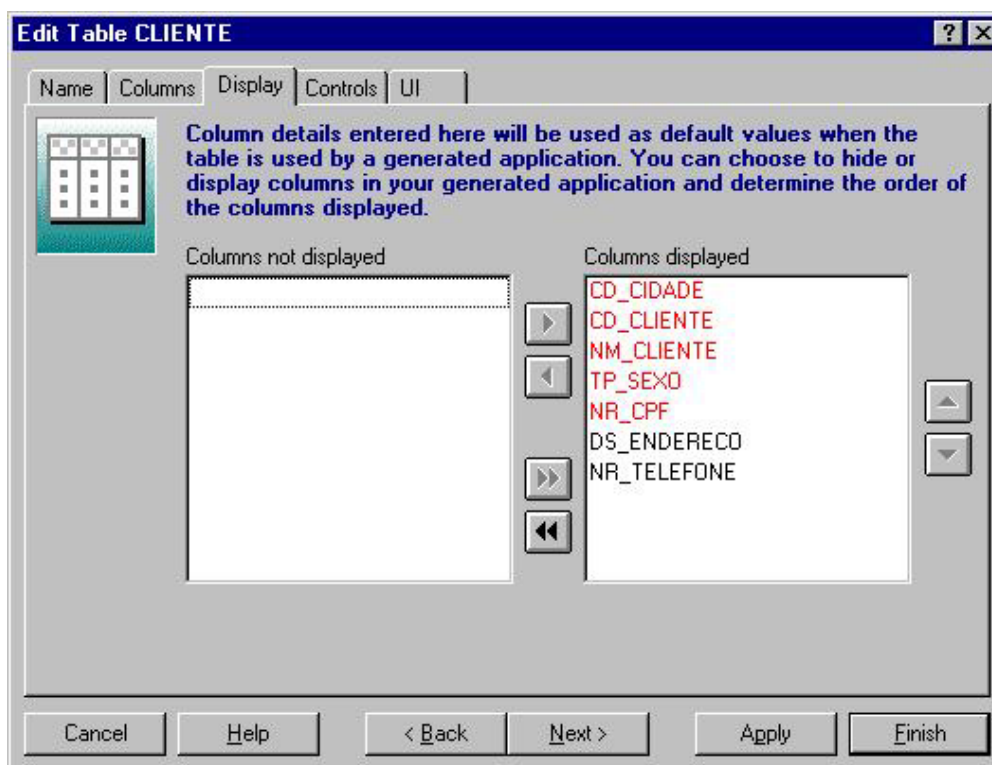
Mandatory

Buttons: Add, Delete, Allowable Values..., Cancel, Help, < Back, Next >, Apply, Finish

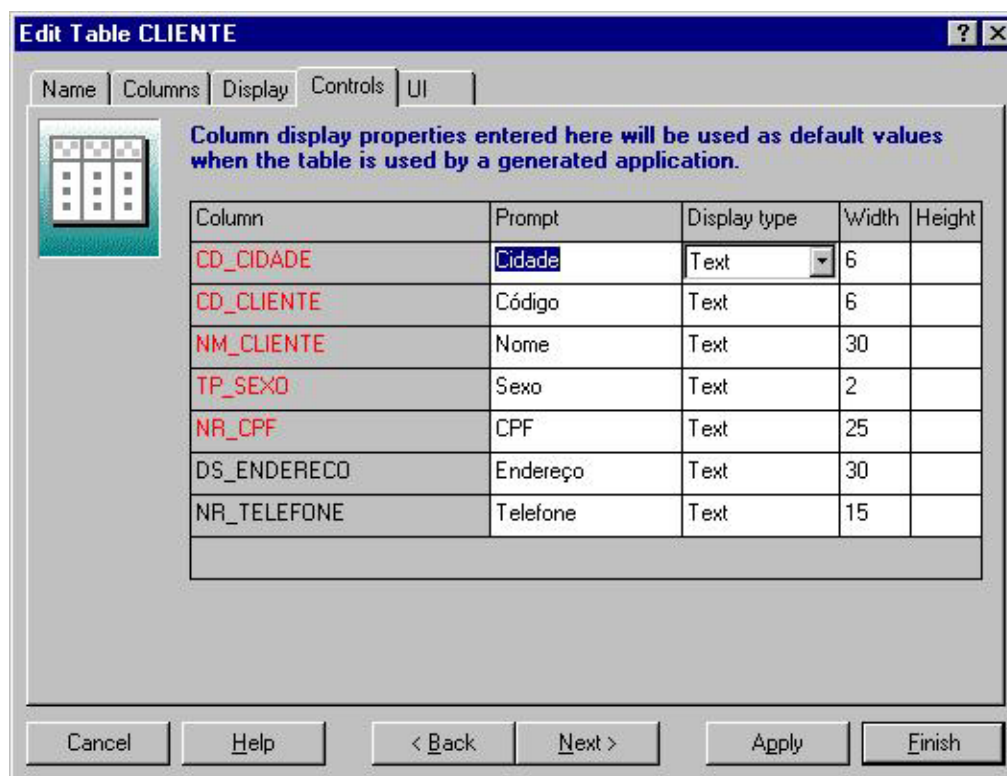
Guia Columns: Nesta guia trataremos da definição das colunas da tabela. As opções mais importantes são:

Domain: associar um domínio pré - definido para a coluna.

Default Value: associa um valor default para a coluna



Guia Display: Esta guia trata das colunas que irão aparecer aplicação.



Guia Controls: Esta guia trata de opções que influenciarão no display das colunas na aplicação.

Prompt: indica o nome da coluna na aplicação.

Display type: mostra o tipo de campo a ser utilizado na aplicação.

Width/Height: indica o tamanho do campo. Geralmente o *Height* deve ser setado para um (1).

Edit Table CLIENTE

Name Columns Display Controls UI

Column display properties entered here will be used as default values when the table is used by a generated application.

Column	Hint text	Uppercase?	Alignment	Display format
CD_CIDADE	digito da cidade	<input type="checkbox"/>	<input type="text" value=""/>	
CD_CLIENTE	Código do cliente	<input type="checkbox"/>		
NM_CLIENTE	Nome do cliente	<input type="checkbox"/>		
TP_SEXO	Sexo	<input type="checkbox"/>		
NR_CPF	CPF	<input type="checkbox"/>		
DS_ENDERECO	Endereço	<input type="checkbox"/>		
NR_TELEFONE	Telefone	<input type="checkbox"/>		

Cancel Help < Back Next > Apply Finish

Guia UI: Esta guia também trata de opções que influenciarão no display das colunas na aplicação.

Hint text: Indica a mensagem de rodapé, mostrada quando o cursor estiver sobre a coluna.

Uppercase: Caso assinalada esta opção para a coluna, está somente aceitará as letras maiúsculas.

Alignment: Nesta opção pode-se selecionar o tipo de alinhamento do campo (Center, Left, Right ou Null).

Display format: Esta opção pode ser preenchida para estabelecer um formato para os dados, por exemplo números 999,99.