

## Trabalhando com múltiplos schemas no PostgreSQL

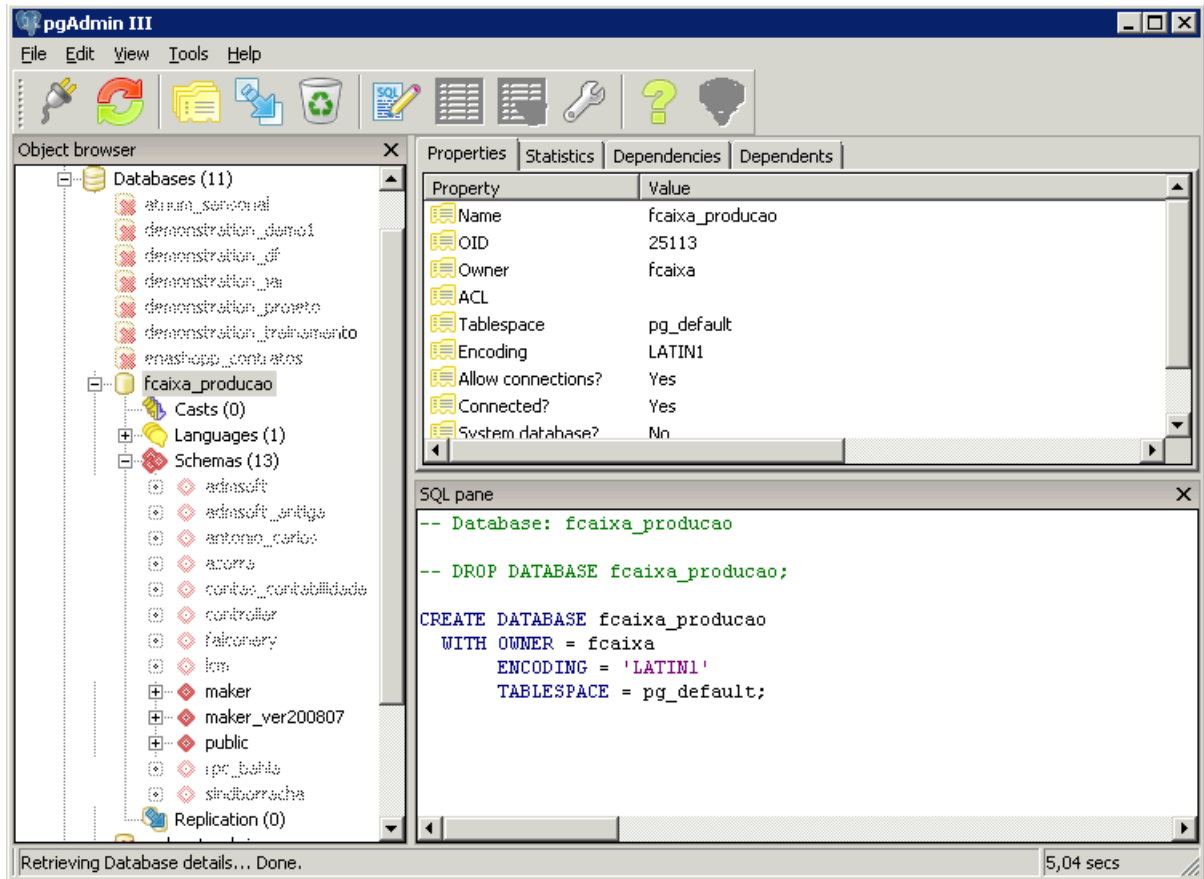
Muitos SGDB's implementam o suporte a schemas, este recurso é muito útil para o agrupamento de tabelas de forma concisa. Há muitas aplicações para isto mais focando em Maker podemos destacar duas boas razões para tal:

1. Versionamento do sistema – Usando schemas podemos ter diversos troncos de desenvolvimento, testar versões betas da aplicação, e separar as implementações maiores para que não interfiram nas versões estáveis.
2. Múltiplos clientes em uma única base – Neste caso teremos diversos “bancos de dados” dos clientes compartilhando uma mesma versão da aplicação, ou seja, somente 1 base para dar manutenção/atualização e diferentes dados de clientes, pois há somente um código fonte (tabelas FR's).

A técnica que costumo empregar é o uso de múltiplos schemas para separar os dados de cada cliente (ou unidade de negócio), separado dos dados do Maker (Tabelas FR\_\*). Para tal eu faço o seguinte, crio três schemas:

1. **public** – Este schema já existe por padrão, ele irá conter as tabelas FR que contém os fontes da aplicação, é bom mantermos este nome porque evitar inicializar o banco acidentalmente ao tentar criar um novo projeto no Maker;
2. **sistema** – Conterá as tabelas e dados de exemplo do sistema sendo desenvolvido, elas serão usadas para testes durante o desenvolvimento;
3. **modelo** – Conterá as tabelas do sistema somente com os dados necessários para uma nova implantação do sistema, ou seja as novas “contas” serão criadas duplicando esse schema;

Após criar um novo projeto no Maker todas as tabelas estão no schema public, assim para separar as tabelas do projeto das tabelas FR basta criar um novo schema chamado **sistema** e executar de criação de suas tabelas neste esquema (experimente o comando `set search_path=sistema`). É necessário mover algumas tabelas FR para este schema (sistema) pois elas fazem parte do seu sistema e devem ser movidas do schema public, pois guardam dados referentes à implantação e não ao projeto. (fontes).



O próximo passo é a configuração do WFRE para que o Maker passe a enxergar dois schemas, para isso pressione e mantenha pressionada a tecla SHIFT esquerda ao abrir o WFRE do projeto, isso fará com que a tela de configuração do WFRE seja exibida antes da conexão com o banco de dados, nesta tela edite a propriedade **RoleName** definindo-a para o seguinte valor: **sisistema,public**. Pressione o botão OK e o Maker deve conectar-se normalmente na base de dados, já “enxergando” os dois schemas.

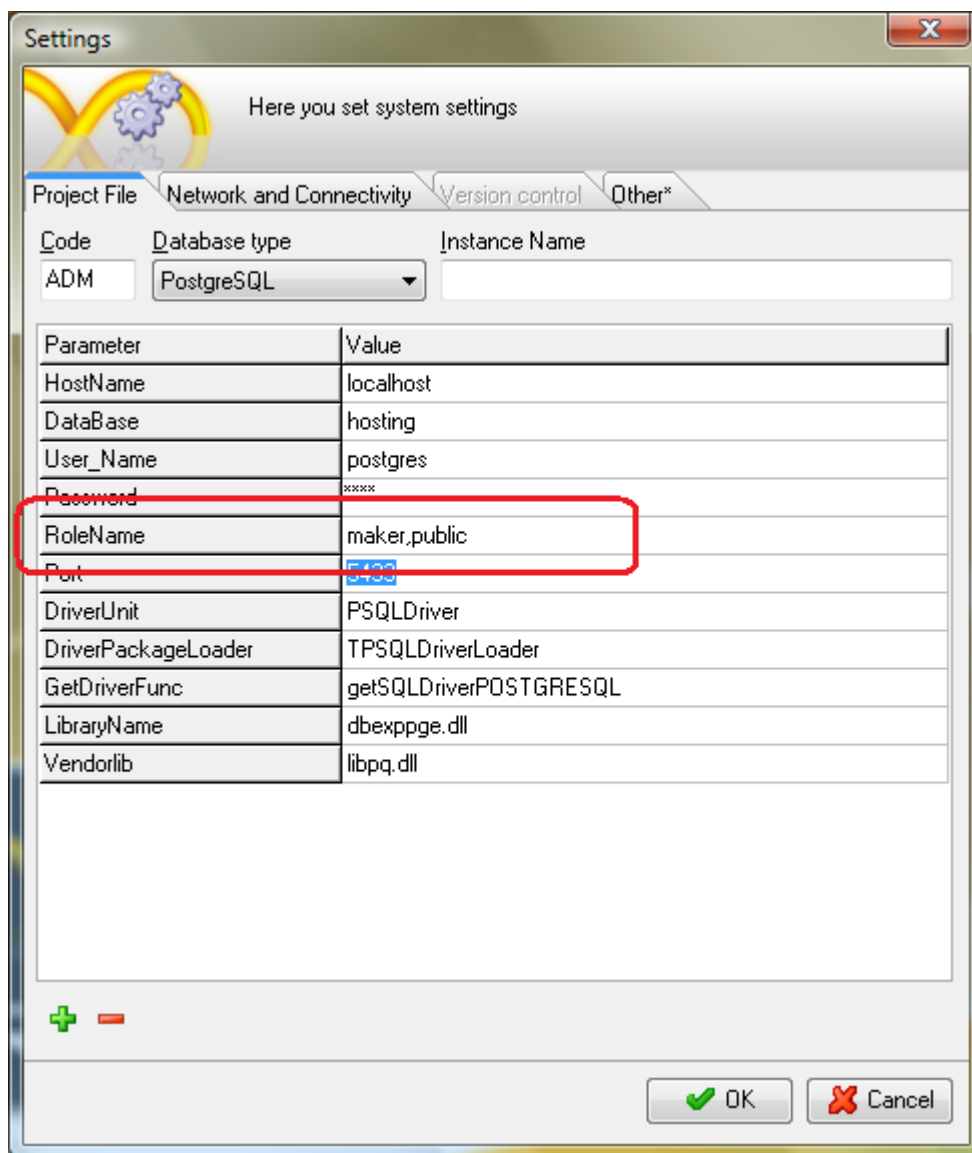


Figura 1 Configurando o WFRE para múltiplos schemas

Neste ponto você criou o WFRE de desenvolvimento, agora crie um novo schema no postgres com o nome **modelo**, e crie uma “base vazia” somente com as tabelas do seu sistema e os dados iniciais, mantenha sempre este esquema atualizado, ele será usado em cada implantação nova.

Para criar cada nova base, implantação, ou unidade de negócio faça um backup somente do schema modelo, depois mude o nome do schema modelo para o nome da nova base a ser implantada, por exemplo: cliente1, depois restaure o backup do schema modelo para que ele seja recriado e continue servindo de modelo para as próximas implantações. Depois de criada a base duplique um WFRE, defina o nome do projeto e configura a propriedade RoleName para **public, cliente1**. Também defina o nome de instância deste WFRE para o mesmo nome do schema CLIENTE1, ou algo que referencie ao novo cliente, este passo é importante, pois o Webrun usará o nome de instância para criar o link de cada sistema, e não somente a sigla do sistema, ou seja se a sigla do sistema for EXP e o nome de instância for CLIENTE1 o identificação do sistema será EXP\_CLIENTE1, isso evita com que sistemas de mesma sigla sejam “misturados”. Agora basta reiniciar o Webrun para atualizar a lista de sistemas.

Há outra vantagem nesta técnica, você pode criar “branches” de desenvolvimento, por exemplo duplicando o schema **public** criando um novo chamado **maker\_dev** e seguir grandes implementações no schema **maker\_dev** e fazer manutenções corretivas no schema de “estável” **public**.

## Anexo I – Script de separação das tabelas FR

O script abaixo move as tabelas FR para o schema **sistema**, adapte para refletir o nome que você escolheu, as seguintes tabelas continuarão no schema public: FR\_USUARIO, FR\_USUARIO\_GRUPO, FR\_GRUPO, FR\_PERMISSAO, FR\_SESSAO, FR\_USUARIO\_SISTEMA essas tabelas guardam informações específicas de cada implantação.

Antes de executar este script certifique-se que o schema **maker** foi criado.

```
--INICIO DO SCRIPT
ALTER TABLE FR_USUARIO SET SCHEMA sistema;
ALTER TABLE FR_USUARIO_GRUPO SET SCHEMA sistema;
ALTER TABLE FR_GRUPO SET SCHEMA sistema;
ALTER TABLE FR_PERMISSAO SET SCHEMA sistema;
ALTER TABLE FR_SESSAO SET SCHEMA sistema;
ALTER TABLE FR_USUARIO_SISTEMA SET SCHEMA sistema;
--FIM DO SCRIPT
```