

# PostgreSQL

[postgresql.org.br](http://postgresql.org.br)



[ivo.nascimento@ianntech.com.br](mailto:ivo.nascimento@ianntech.com.br)  
<http://www.ianntech.com.br>



# Cronograma

- 1) PostgreSQL e sua história.
- 2) Conectando-se ao servidor
- 3) Como é organizado os dados
- 4) Trabalhando com Objetos do banco de dados
- 5) Trabalhando com o Catálogo
- 6) Utilizando o Pgbench para teste de perf.
- 7) Trabalhando com o Contrib

# Um pouco de história...

O sistema gerenciador de banco de dados objeto-relacional hoje conhecido por PostgreSQL, é derivado do pacote POSTGRES escrito na Universidade da Califórnia em Berkeley. Com mais de uma década de desenvolvimento por trás, o PostgreSQL é atualmente o mais avançado banco de dados de código aberto disponível em qualquer lugar.

Michael Stonebraker patrocinado pela ARPA, ARO, NSF criou o Postgres.

A primeira apresentação foi em 1988.

Versão 2 liberada em 1990.

Versão 3 em 1991.

Andrew Yu e Jolly Chen adicionaram o interpretador da linguagem SQL e o produto se tornou Postgres95.

Em 1996 torna-se o postgresQL e chega a versão 6.0

A versão 7.0 2000-05-08.

A versão 7.4 em 2003-11-17.

A versão 8.0 em 2005-01-19.

# Conectando-se ao Servidor(login/senha it:it)

As máquinas utilizadas neste treinamento rodam o ubuntu 6.06 desktop(apesar da ubuntu disponibilizar o ubuntu server) em uma virtual machine onde estão instalados os recursos do postgresQL sendo.

- O servidor na versão 8.1(por que não 8.2?)
- A contrib
- O aplicativo PgAdmin
- O aplicativo PgVisio(beta)

# PostgreSQL(login/senha postgres:postgres)

## Aplicativos postgresQL

psql: aplicativo de terminal utilizado para acessar um servidor postgresQL.

Para usar:

**Psql -d [dbname] -Uusuario -h host**

pgAdmin: Aplicativo gráfico mantido pelo projeto para acessar o servidor postgresQL.

Para usar:

**clique no ícone do postgresQL no painel superior**

# Como é Organizado os dados

1.Estrutura de arquivo

2.Organização de Objetos

# Estrutura de arquivo

Diretório Principal:

`/etc/postgresql/8.1/main/`

Diretório de executáveis:

**`/usr/postgresql/lib/8.1/bin`**

diretorio de dados

**`/var/lib/postgresql/8.1/main`**

`/etc/postgresql/8.1/main/pgdata(link)`

diretorio de log

**`/var/log/postgresql/`**

`/etc/postgresql/8.1/main/log(link)`

# Subdiretórios de pgdata

- 1.base: bases de dados
- 2.blobal: data acessíveis para todo o cluster
- 3.pg\_clog: status das transações
- 4.pg\_subtrans: status de subtransações
- 5.pg\_multixact: status de transação para shared row lock
- 6.pg\_tblspc: links simbolicos de tablespace
- 7.pg\_xlog: arquivos do wal
- 8.pg\_twophase: status dos procedimentos preparados.

# Dica - Como melhorar a performance...

Coloque em discos diferentes pastas e arquivos que tenham bastante IO como:

log

indices

pg\_xlog

Em alguns casos pode proceder com links e em outros pode utilizar tablespaces.

# Organização dos Objetos

**Tablespace**

**schema**

**table**

**indice**

**stored procedure**

**operator**

**sequence**

**type**

**view**

# Trabalhando com Objeto

1.Tablespace

3.Sequence

5.Tipo de Dados

7.Domínio

# Tablespace

```
CREATE TABLESPACE  
tablespacename [ OWNER  
username ] LOCATION 'directory'
```

# Domain

```
CREATE DOMAIN name [ AS ]  
data_type  
[ DEFAULT expression ]  
[ constraint [ ... ] ]
```

# Sequence

```
CREATE [ TEMPORARY | TEMP ] SEQUENCE  
name [ INCREMENT [ BY ] increment ]  
[ MINVALUE minvalue | NO MINVALUE ]  
[ MAXVALUE maxvalue | NO MAXVALUE ]  
[ START [ WITH ] start ] [ CACHE cache ]  
[ [ NO ] CYCLE ]
```

# Type

```
CREATE TYPE name AS  
( attribute_name data_type [, ... ] )
```

# Localizando Objetos

# Monitorando o Servidor

pgbench

# Contrib

*1.DBLink*

*2.pgstattuple*

# Type

```
CREATE TYPE name (  
  INPUT = input_function,  
  OUTPUT = output_function  
  [ , RECEIVE = receive_function ]  
  [ , SEND = send_function ]  
  [ , ANALYZE = analyze_function ]  
  [ , INTERNALLENGTH = { internallength |  
  VARIABLE } ]  
  [ , PASSEDBYVALUE ]  
  [ , ALIGNMENT = alignment ]  
  [ , STORAGE = storage ]  
  [ , DEFAULT = default ]  
  [ , ELEMENT = element ]  
  [ , DELIMITER = delimiter ]
```