

Acesso ao PostgreSQL

Prof. Ronaldo Castro de Oliveira
FACOM – UFU

PostgreSQL

- ▶ É um SGBD que incorpora o modelo relacional para o banco de dados e suporta a linguagem SQL como linguagem de consulta padrão.
- ▶ Executa em diversas plataformas
 - ▶ Unix-like: FreeBSD; Linux; Mac OS X
 - ▶ Windows
- ▶ É um software livre com código-fonte aberto.



Histórico do PostgreSQL

- ▶ Um banco de dados relacional chamado Ingres foi desenvolvido entre 1977-1985 na Universidade da Califórnia em Berkley,
- ▶ Seu código foi usado para a construção do primeiro SGBD comercial
 - ▶ Hoje o Ingres é um software livre
- ▶ Em 1986 um novo projeto na UCB foi iniciado: o Postgres, que utilizou várias idéias do Ingres.
- ▶ Em 1994 o Postgres passou receber suporte à **SQL**, mudando assim sua denominação para Postgres95
- ▶ Em 1996 recebeu uma nova denominação: PostgreSQL e foi aberto à desenvolvedores na rede.
- ▶ Em 2005: *PostgreSQL 8.0* com facilidades de instalação em Windows



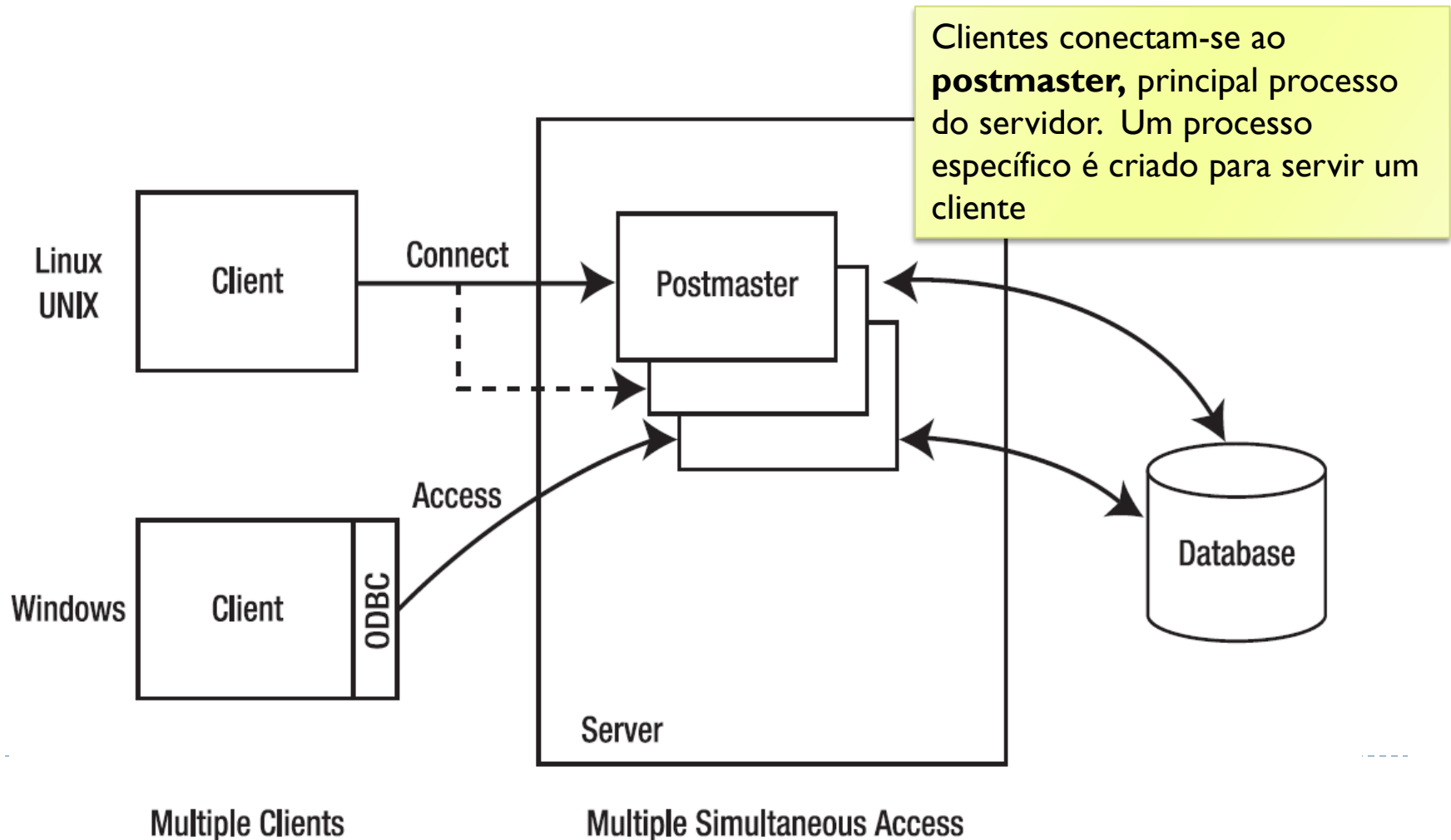
Arquitetura do PostgreSQL

- ▶ Assim como vários SGBDs comerciais, o PostgreSQL pode ser usado em um ambiente **Cliente/Servidor**
- ▶ Existe um processo servidor que é executado em uma única máquina
- ▶ Os aplicativos que necessitam acessar os dados devem fazer isso via o processo servidor
- ▶ Os clientes não conseguem acessar o dado diretamente, mesmo quando estão rodando no mesmo computador que o servidor



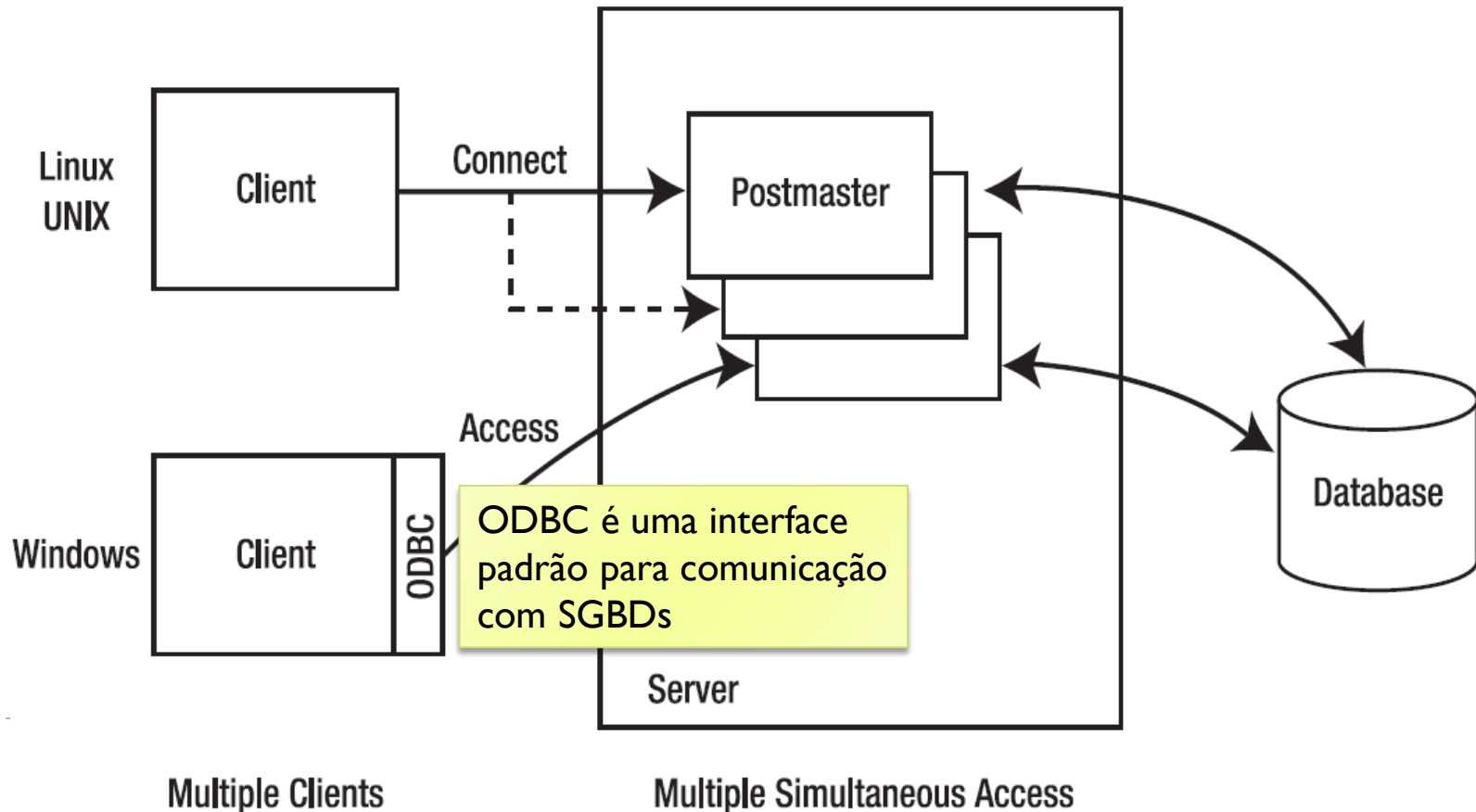
Arquitetura Cliente/Servidor

- ▶ É possível que o cliente rode Windows e o servidor seja Unix
- ▶ Acesso simultâneo/ Necessária rede TCP/IP



Arquitetura Cliente/Servidor

- ▶ ODBC – *Open Database connectivity* permite que vários aplicativos usem o BD, como por exemplo o Excel/Access



Acessando aos dados em PostgreSQL

- ▶ Aplicativo em linha de comando: **psql**
 - ▶ É possível executar consultas SQL interativamente
- ▶ Aplicação gráfica: **pgAdmin III**
 - ▶ Utilizaremos o pgAdmin III neste curso
- ▶ Servidor web: apache rodando **phpPgAdmin**
- ▶ Criação de aplicativos clientes utilizando bibliotecas de acesso ao BD
- ▶ Acesso indireto via ODBC ou JDBC (Java Database Connectivity)



Mais informações

<http://www.postgresql.org/docs/8.1/interactive/app-postmaster.html>

postmaster

Name

postmaster -- PostgreSQL multiuser database server

Synopsis

postmaster [-A 0 | 1] [-B *nbuffers*] [-c *name=value*] [-d *debug-level*] [-D *datadir*] [-F] [-h *hostname*] [-i] [-k *directory*] [-l] [-N *max-connections*] [-o *extra-options*] [-p *port*] [-S] [--name=*value*] [-n | -s]

Description

postmaster is the PostgreSQL multiuser database server. In order for a client application to access a database it connects (over a network or locally) to a running postmaster. The postmaster then starts a separate server process ("[postgres](#)") to handle the connection. The postmaster also manages the communication among server processes.

Obs: both postmaster and postgres are in fact the same program (postgres.exe)



Criando Instância do PostgreSQL em Nuvem

- ▶ Acessar o servidor em nuvem:

- ▶ <https://www.elephantsql.com/>

- ▶ Solicitar a criação de uma conta gratuita clicando em:

Get a managed database today

- ▶ Escolher a Instância Tiny Turtle (gratuita)

- ▶ Clicar em **Try now for FREE**

- ▶ Criar uma conta no ElepahntSql

- ▶ Clicar em **Don't have an account? [Sign up](#)**


- ▶ Fazer o login da conta criada

TINY TURTLE

Shared high performance server

20 MB data

5 concurrent connections



FREE
GET STARTED

Try now for FREE

Criando Instância do PostgreSQL em Nuvem


► Selecione o plano de acesso:

Select a plan and name - Step 1 of 4

Name	<input type="text" value="TesteAulaBD"/>	Plan
Plan	<input type="text" value="Tiny Turtle (Free)"/>	
Tags	<input type="text" value="BD"/>	

Tags are used to separate your instances between projects. This is primarily used in the project listing view for easier navigation and access control.

Tags allow admins to [manage team members access](#) to different groups of instances.



Tiny Turtle

► Selecione a região onde irá rodar sua instância:

Select a region and data center - Step 2 of 4


Data center	<input type="text" value="SA-East-1 (Sao Paulo)"/>
-------------	--



Criando Instância do PostgreSQL em Nuvem

► Revise e Crie a sua instância do PostgreSQL:

Plan



Tiny Turtle

Total: Free

Name:	TesteAulaBD
Provider:	Amazon Web Services
Region:	SA-East-1 (Sao Paulo)
Tags:	BD

Cancel

Create instance

Criando Instância do PostgreSQL em Nuvem






- ▶ Verificar se a instância foi criada:

Instances					+ Create New Instance
BD					
Name	Host	Plan	Datacenter	Actions	
TesteAulaBD	kesavan	Tiny Turtle	Amazon Web Services SA-East-1 (Sao Paulo)	Edit	



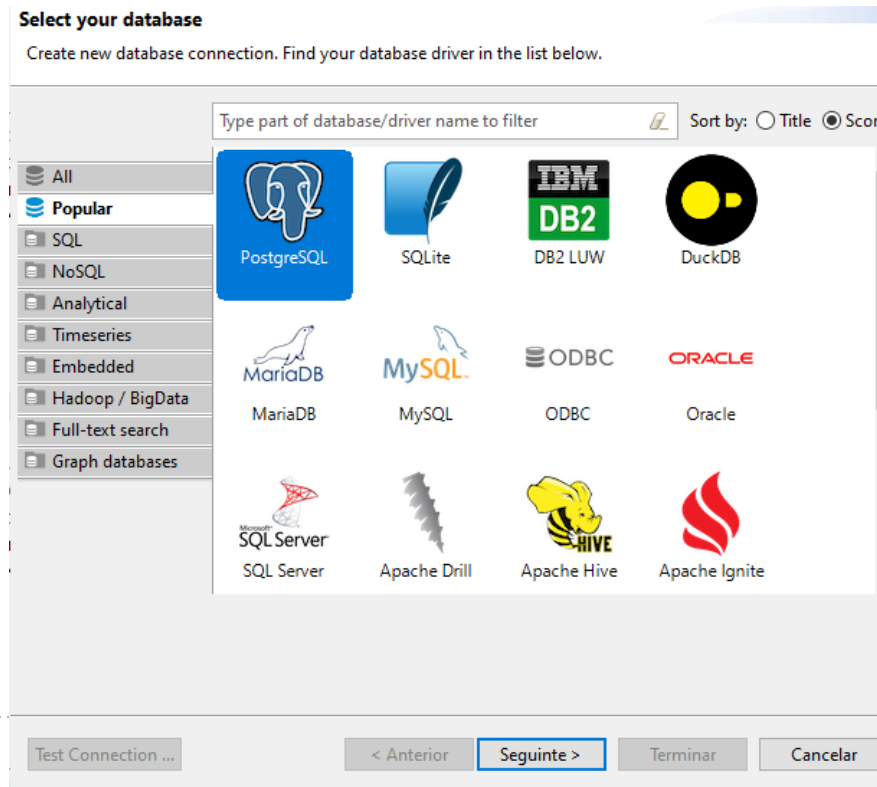
Criando Instância do PostgreSQL em Nuvem

- ▶ Clicar no nome da instância criada para ver características e informações do servidor, usuário e senha de acesso.

Server	kesavan.db.elephantsql.com (kesavan-01)	
Region	amazon-web-services::sa-east-1	
Created at	2022-06-27 17:06 UTC+00:00	
User & Default database	bxnhuezb	<button>Reset</button>
Password	lcf-JN...  	
URL	postgres://bxnhuezb:***@kesavan.db.elephantsql.com/bxnhuezb  	

Configurando o Dbeaver para conectar ao BD criado na nuvem

- ▶ Abra o Dbeaver e clique em:
 - ▶ Banco de dados → 
- ▶ Escolha o Banco de Dados PostgreSQL e clique em no botão Seguinte



Configurando o Dbeaver para conectar ao BD criado na nuvem

- Defina o servidor, nome do Banco de Dados, usuário e senha que foram gerados pelo ElephantSql

Connect to a database

Configurações de conexão
PostgreSQL connection settings

PostgreSQL

Main PostgreSQL Driver properties SSH Proxy SSL

Server

Host: Port:

Database:

Authentication

Authentication:

Username:

Password: ☒ Save password locally

Advanced

User role: Local Client:

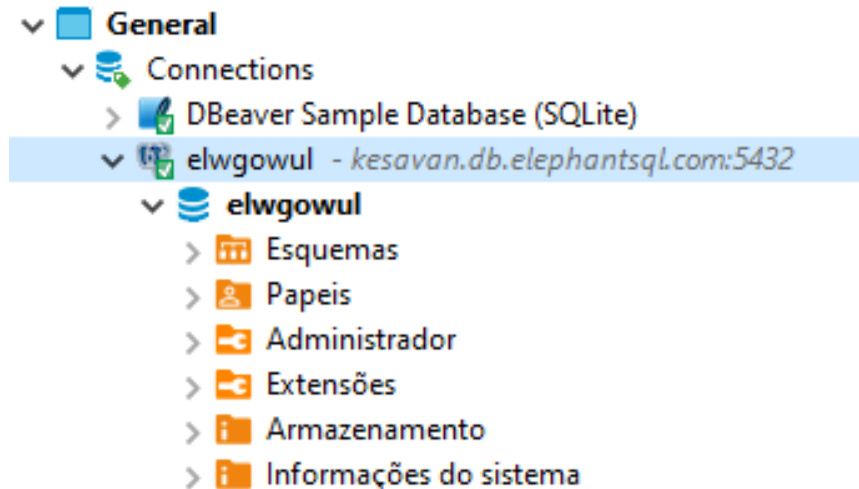
i You can use variables in connection parameters.

Driver name: PostgreSQL

Clicar em Terminar

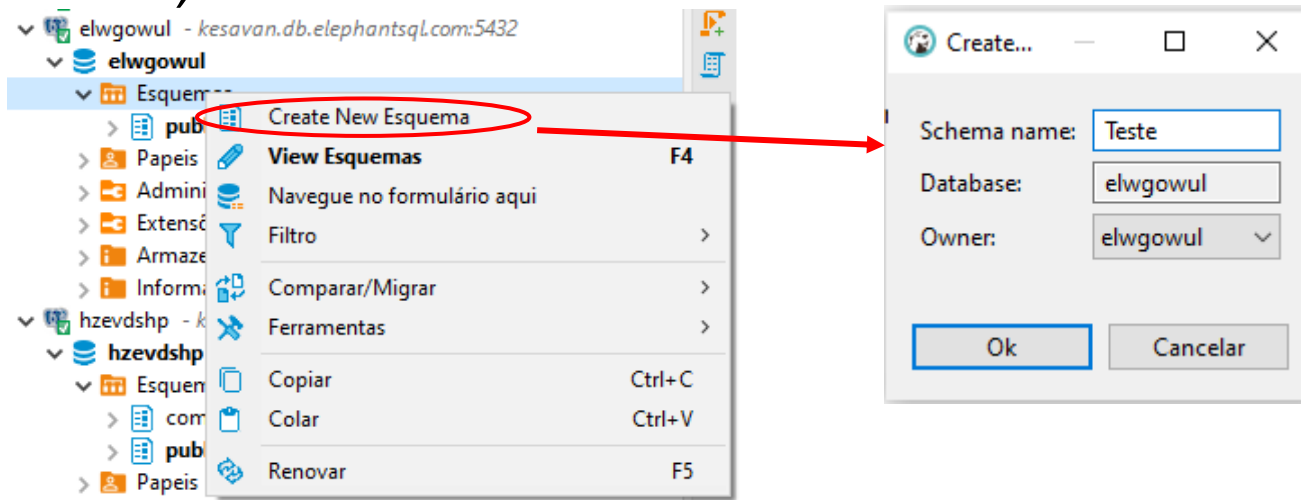
Configurando o Dbeaver para conectar ao BD criado na nuvem

- ▶ Após a execução dos passos anteriores o Banco de Dados criado no ElephantSql já estará conectado ao Dbeaver.

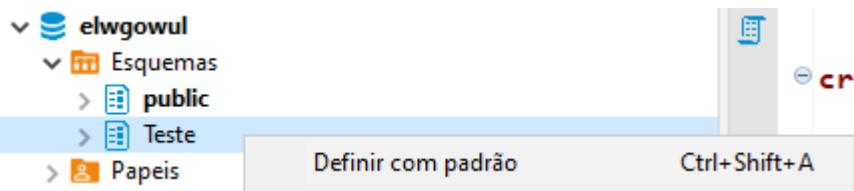


Configurando o Dbeaver para conectar ao BD criado na nuvem

- ▶ Para utilizar o Banco de Dados crie um novo esquema dentro deste banco de dados e coloque este esquema como padrão de execução (botão direito em cima de Esquemas)



- ▶ Colocando como padrão de execução (botão direito no Esquema criado:



Esquemas

- ▶ A organização do banco de dados em esquemas é útil pois
 - ▶ Permite o acesso do usuário aos objetos de qualquer esquema do banco de dados que ele está conectado (diferente do caso de diferentes banco de dados)
 - ▶ Permite organizar os objetos do banco de dados em grupos lógicos, facilitando o gerenciamento
 - ▶ Permite vários usuários utilizarem um mesmo banco de dados sem que um interfira no outro.
- ▶ There is nothing special about the public schema except that it exists by default. It can be dropped, too.



Apagando os esquemas

-- criando esquemas

create schema *universidade*;

create schema *farmacia*;

create schema *oficina*;

-- usando os esquemas

set search_path to *universidade*;

set search_path to *farmacia*;

set search_path to *oficina*;

-- apagando **todas** as tabelas e o esquema

drop schema *cinema* **cascade**;

drop schema *grupo01* **cascade**;

drop schema *public* **cascade**;

