



PostgreSQL Administration

PostgreSQL is a sophisticated open source relational database management system. PostgreSQL supports the structured query language (SQL) with a wide variety of data types and support for: triggers, a number of embedded programming languages, and full transactional processing.

Objectives

Instructor led training course on the administration of the PostgreSQL Relational Database Management System. You will be able to develop and maintain powerful PostgreSQL servers.

Duration

2 days

Audience

Database administrators, developers, IT support staff.

Pre-Requisites

SQL course or equivalent knowledge.

Linux Essentials course or experience with using the Linux command line.

Outline

- Introduction to PostgreSQL
 - A Brief History of PostgreSQL
 - Getting Started
 - Installation
 - Architectural Fundamentals
 - Using the psql terminal based client
- PostgreSQL SQL Language specifics
 - Creating a Database
 - Accessing a Database
 - Inheritance
- Server Setup and Operation
 - The PostgreSQL User Account
 - Creating a Database Cluster
 - Starting the Database Server
 - Server Start-up Failures
 -

- Client Connection Problems
- Shutting Down the Server
- Server Configuration
 - Setting Parameters
 - File Locations
- Security
 - Preventing Server Spoofing
 - Encryption Options
 - Connections and Authentication
 - Connection Settings
 - Security and Authentication
 - The pg_hba.conf file
 - Username maps
 - Authentication methods
 - Trust authentication
 - Password authentication
 - Authentication problems
 - Database Roles
 - Role Attributes
 - Privileges
 -

Role Membership

- Resource Consumption
 - Memory
 - Kernel Resource Usage
 - Cost-Based Vacuum Delay
 - Background Writer
 - Asynchronous Behavior
 - Write Ahead Log
 - Settings
 - Checkpoints
 - Archiving
- Error Reporting and Logging
 - Where To Log
 - When To Log
 - Message severity levels
 - What To Log
 - Using CSV-Format Log Output
 - Run-Time Statistics
 - Query and Index Statistics Collector
 - Statistics Monitoring
-

Vacuum

- Automatic Vacuuming



Backup and Restore

- SQL Dump
- Restoring the dump
- Using pg_dumpall
- Handling large databases
- Use compressed dumps.
- Use split.
- Use pg_dump's custom dump format.
- File System Level Backup
- Continuous Archiving and Point-In-Time Recovery (PITR)
- Setting up WAL archiving
- Making a Base Backup
- Recovering using a Continuous Archive Backup
- Recovery Settings
- Timelines
- Tips and Examples
- Standalone hot backups
- archive_command scripts
-

- Record-based Log Shipping

-

- Incrementally Updated Backups

-

- Migration Between Releases

-

- Monitoring Database Activity

-

- Standard Unix Tools

-

- The Statistics Collector

-

- Statistics Collection Configuration

-

- Viewing Collected Statistics

-

- Viewing Locks

-

- Dynamic Tracing

-

- Compiling for Dynamic Tracing

-

- Built-in Probes

-

- Using Probes

-

- Defining New Probes

-

- Monitoring Disk Usage

-

- Determining Disk Usage

-

- Disk Full Failure