



Redundância com PostgreSQL

Por: Gustavo G. Pereira



1: Introdução à Tolerância a Falhas

Já vimos nas aulas




2: Conceitos Chave em Tolerância a Falhas

- Tipos de Falhas em Banco de Dados
 - Falhas de Hardware
 - Falhas de Software
 - Desastres Naturais
- Estratégias de Tolerância a Falhas
 - Redundância
 - Replicação




3: Tolerância a Falhas no PostgreSQL


- Replicação em PostgreSQL
 - Streaming Replication
 - Logical Replication
- Failover e Switchover
 - Definições e diferenças
 - Como o PostgreSQL gerencia esses processos




3.1: Replicação em PostgreSQL: Replicação em geral refere-se ao processo de criar e manter cópias de dados em um ou mais servidores para fins de redundância, escalabilidade ou backup. No PostgreSQL, a replicação pode ser realizada de várias maneiras, incluindo Streaming Replication e Logical Replication.



3.2: Streaming Replication: É uma forma de replicação nativa suportada pelo PostgreSQL, onde os dados são replicados diretamente do servidor primário para um ou mais servidores secundários em tempo real, através do envio contínuo de registros de transações (WAL - Write Ahead Logs).



3.3: Logical Replication: É outro método de replicação nativo no PostgreSQL, que replica os dados baseados em instruções de alto nível (como INSERT, UPDATE, DELETE) em vez de replicar os registros de transação brutos. Isso permite maior flexibilidade e granularidade na replicação, mas também pode ter um desempenho ligeiramente inferior ao Streaming Replication.



3.4: Failover e Switchover: Ambos os termos referem-se à troca de um servidor primário por um servidor secundário em caso de falha ou para manutenção planejada. A diferença entre eles é que o failover é automático e ocorre quando o sistema detecta uma falha no servidor primário, enquanto o switchover é uma operação manual, onde um administrador de banco de dados realiza a troca de forma controlada e planejada.



4: Práticas e Ferramentas de Suporte

- Ferramentas de Monitoramento
 - PGAdmin, Prometheus, etc.
- Automação de Failover
 - Patroni, Pgpool-II, etc.
- Backups e Recuperação
 - Tipos de backup (completo, incremental)
 - Estratégias de recuperação



5: Estudos de Caso e Melhores Práticas

- Estudos de Caso
 - Exemplos reais de implementação de tolerância a falhas usando PostgreSQL
- Melhores Práticas
 - Manutenção preventiva
 - Testes regulares de recuperação
 - Documentação e treinamento de equipe



6: Uma breve explicação sobre o Docker

- O que é?
- Para que serve?



<https://github.com/ietxaniz/pg-logrepl/blob/main/LICENSE>

<https://github.com/eremeykin/pg-primary-replica>

<https://www.postgresql.org/docs/14/wal-intro.html>