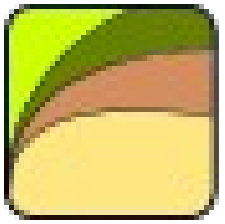


atmantree.com

En caso de dudas escriba a:
info@atmantree.com

Alta Disponibilidad con PostgreSQL

Introducción

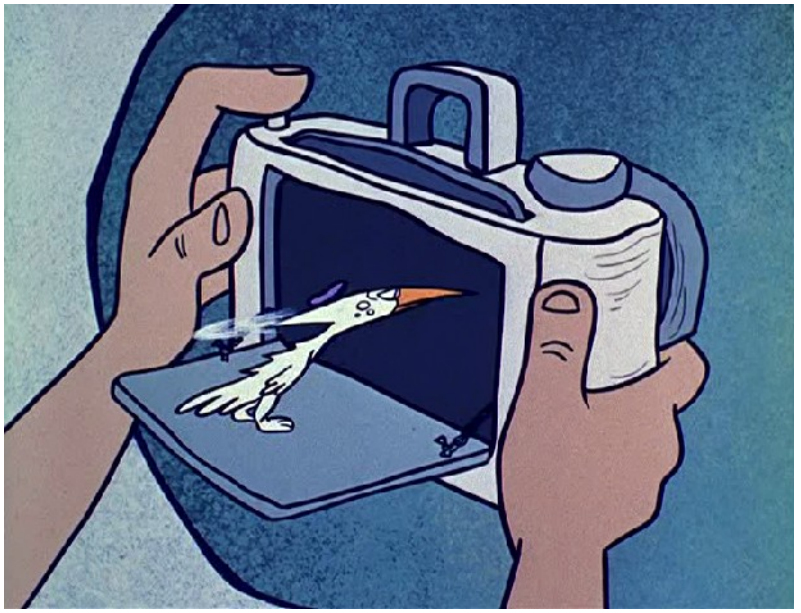
Agenda

- Bases de Datos
- PostgreSQL
- Alta Disponibilidad (HA)
- Términos
  - Replicación
  - Métodos
  - Balance de Cargas
  - Consultas Distribuidas
  - Divergencia
  - ACID
  - Tolerancia a Fallos
  - Síncrona vs. Asíncrona
  - Diligente vs. Relajada
  - Particionado de Datos
  - Cluster
  - Grid
  - Discos Compartidos
  - Shared-Nothing
- Alternativas
- Programas

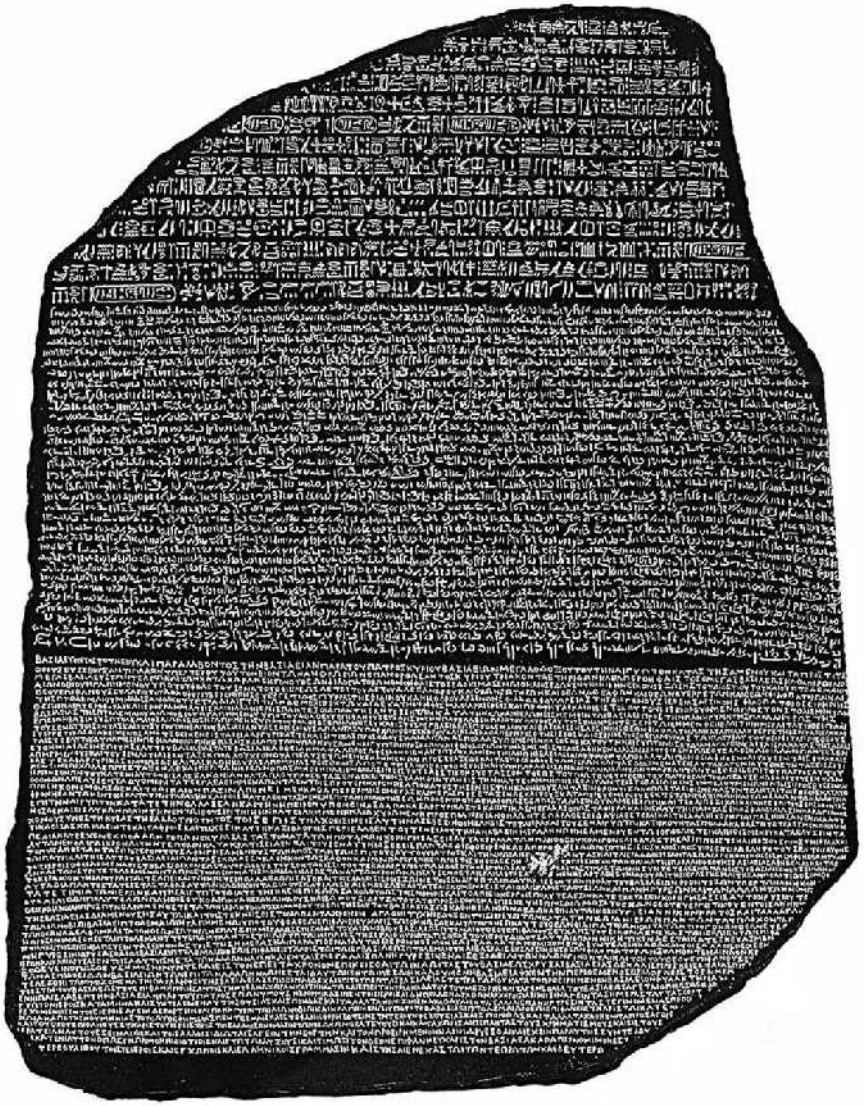
Bases de Datos

Para entender cómo funciona algo lo mejor es imaginar cómo lo harían los Picapiedras

Bases de Datos



Sistema de Digitalización



Almacenamiento de Calidad

Bases de Datos

Páginas de Memoria

Datacenter

Bases de Datos

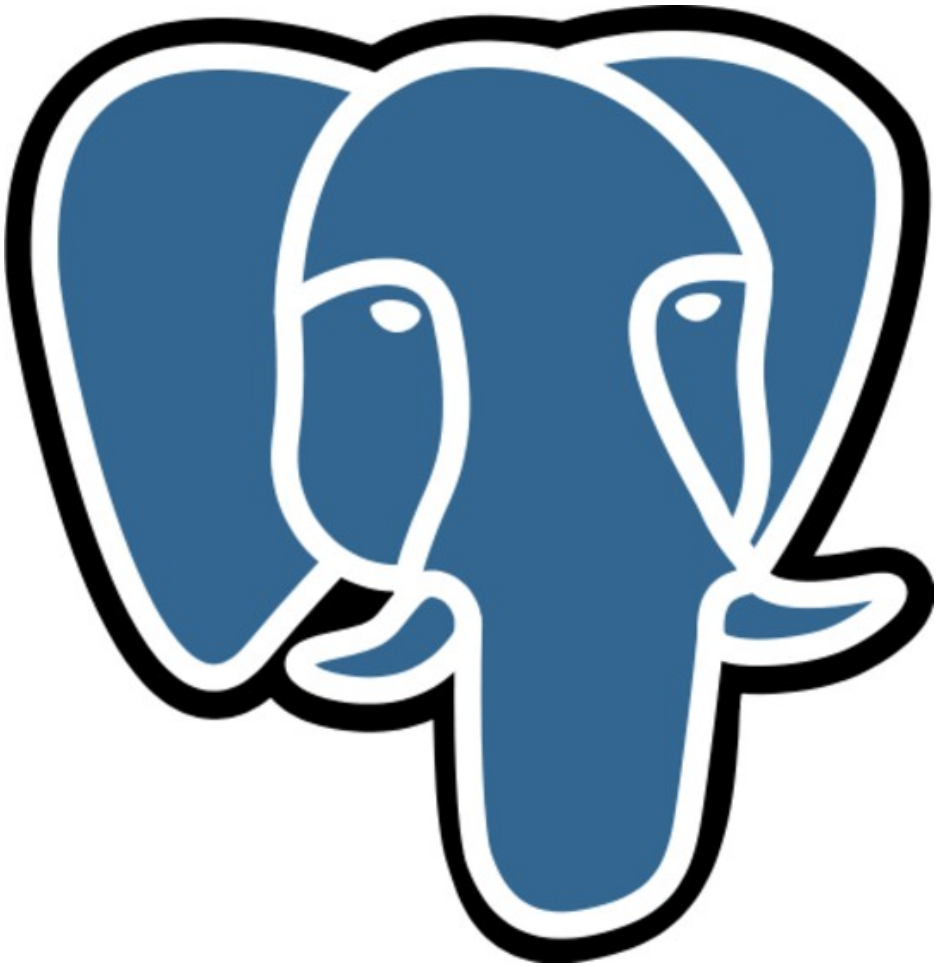
Caída del Sistema

Bases de Datos

Regresamos en Breve con Los Picapiedras

PostgreSQL

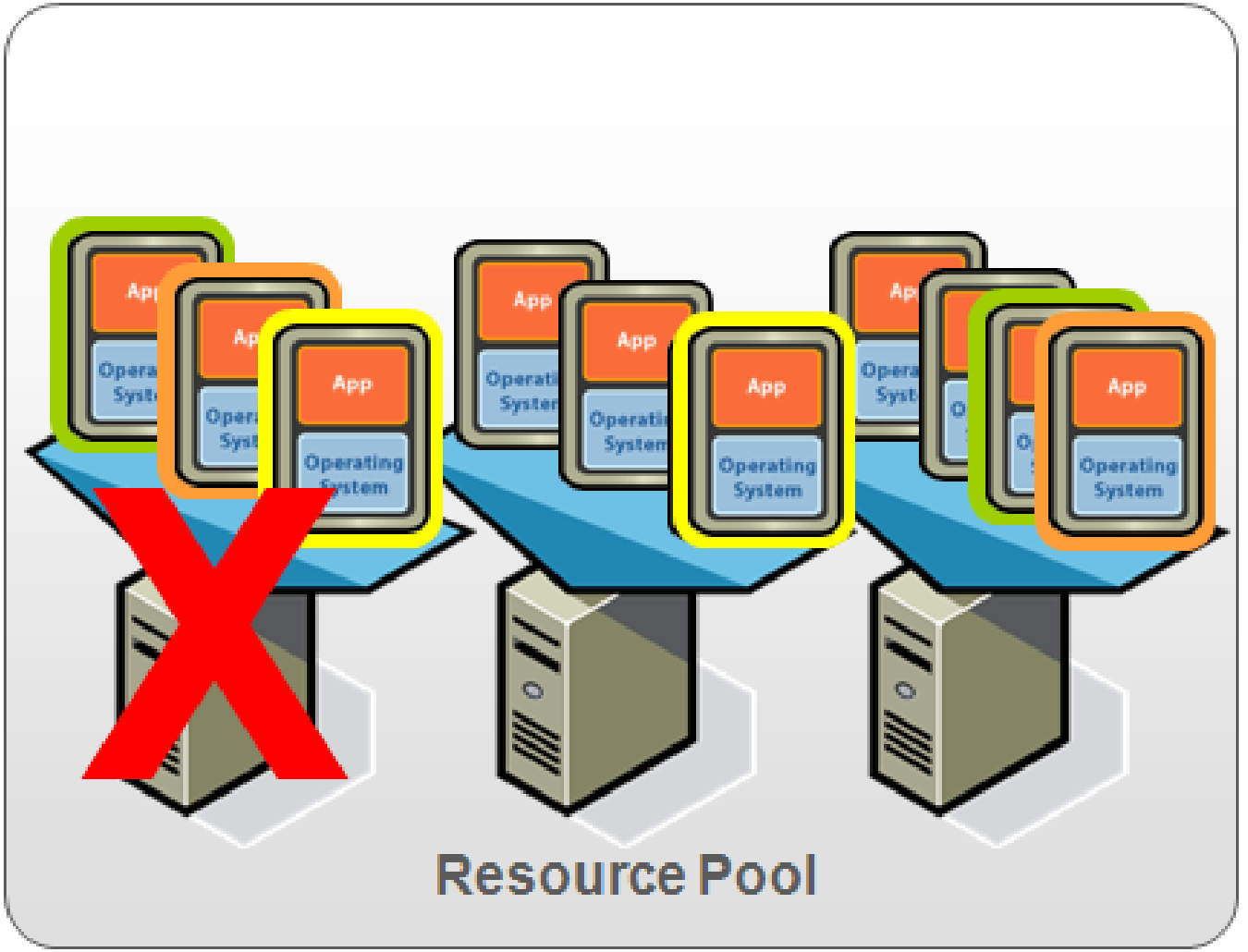
PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con código fuente disponible libremente. Utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema.



PostgreSQL

- Nace en la Universidad de Berkeley en 1977
- Sinónimo de:
  - Estabilidad
  - Potencia
  - Robustez
  - Facilidad de administración
  - Estándares

Alta Disponibilidad



Alta Disponibilidad

Según Wikipedia:

"**Alta disponibilidad** (High availability) es un protocolo de diseño del sistema y su implementación asociada que asegura un cierto grado absoluto de continuidad operacional durante un período de medición dado. Disponibilidad se refiere a la habilidad de la comunidad de usuarios para acceder al sistema, someter nuevos trabajos, actualizar o alterar trabajos existentes o recoger los resultados de trabajos previos. Si un usuario no puede acceder al sistema se dice que está no disponible. El término tiempo de inactividad (downtime) es usado para definir cuándo el sistema no está disponible"

Alta Disponibilidad

- Viene definida por el resultado, no por la estrategia a implementar.
- Se mide en términos de “Tiempo al Aire” y “Tiempo Fuera”.
- El enemigo a vencer son las caídas de servicio y tiempos de recuperación.
- A la Alta Disponibilidad se le suma en las Bases de Datos todos los aspectos de Calidad relacionadas con los servicios de este tipo.

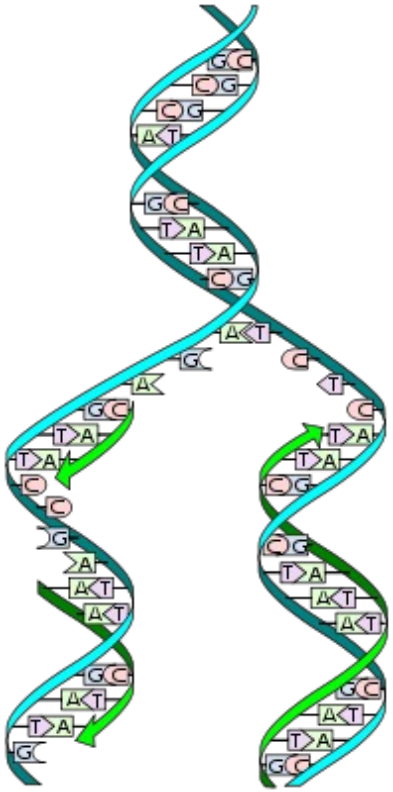
Alta Disponibilidad

- Recuperar un sistema toma su tiempo
- La estrategia por excelencia es la redundancia de recursos
- En los recursos redundantes hacemos copias (Replicación)

Algunos Términos y Definiciones de Alta Disponibilidad

Replicación

¿Qué entendemos por Replicación?

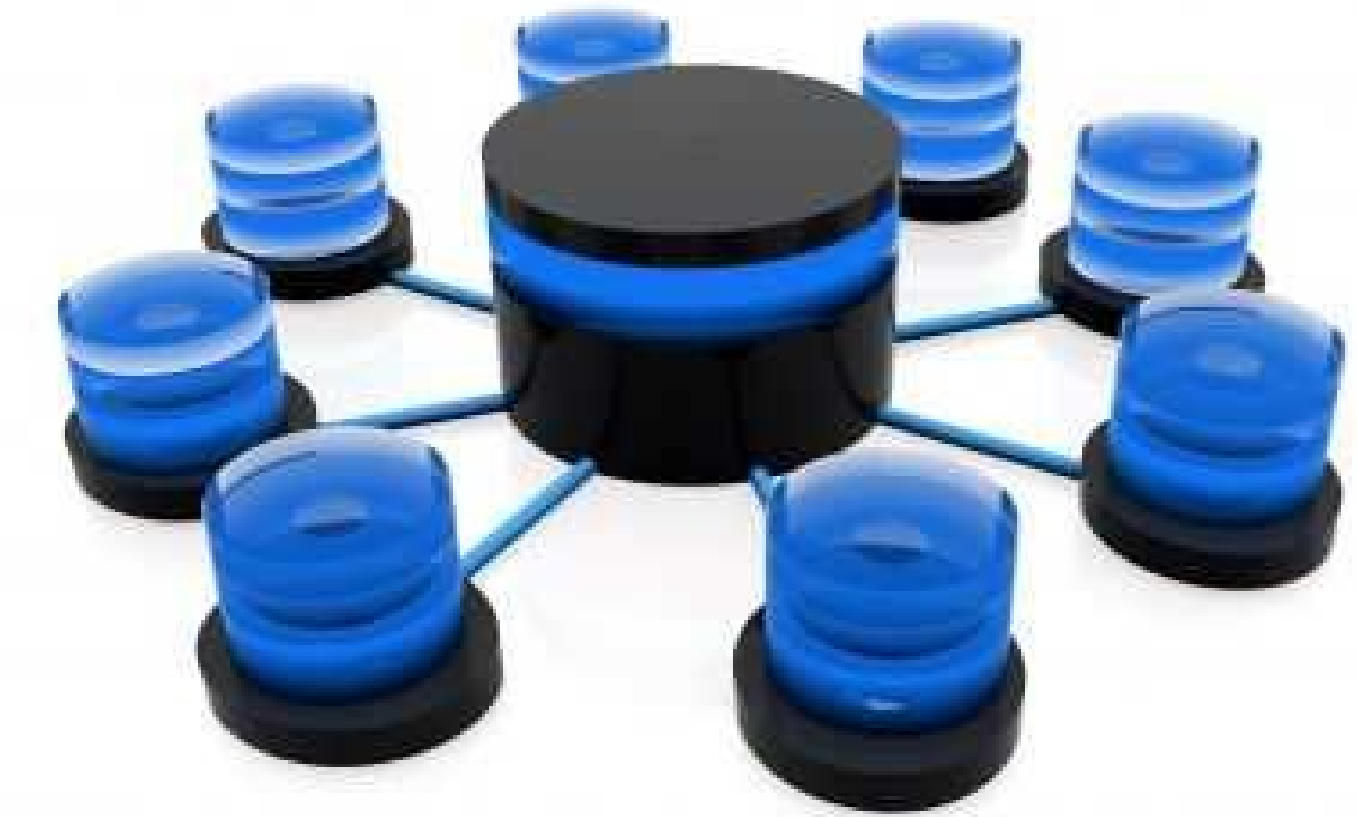


Replicación

Para los sistemas de base de datos, la replicación es un proceso de intercambio de datos de transacciones para garantizar la coherencia entre los nodos de la base de datos redundantes. Esto mejora la tolerancia a fallos, lo que conduce a una mayor confiabilidad del sistema en general. La replicación de bases de datos también se puede llamar bases de datos distribuidas, especialmente cuando se combina con otras características interesantes.

Replicación

En Bases de Datos



Métodos de Replicación

Replicación de Transacciones

- Copia cada una de las transacciones ejecutadas en el sistema donde se encuentran conectados los usuarios hacia un conjunto de bases de datos secundarias.
- Estas bases de datos secundarias recibe los cambios propagados por la base de datos primaria a través de un canal de comunicación.
- En esta técnica existe el posible inconveniente de las diferencias generadas por funciones no determinísticas como now() o random().

Métodos de Replicación

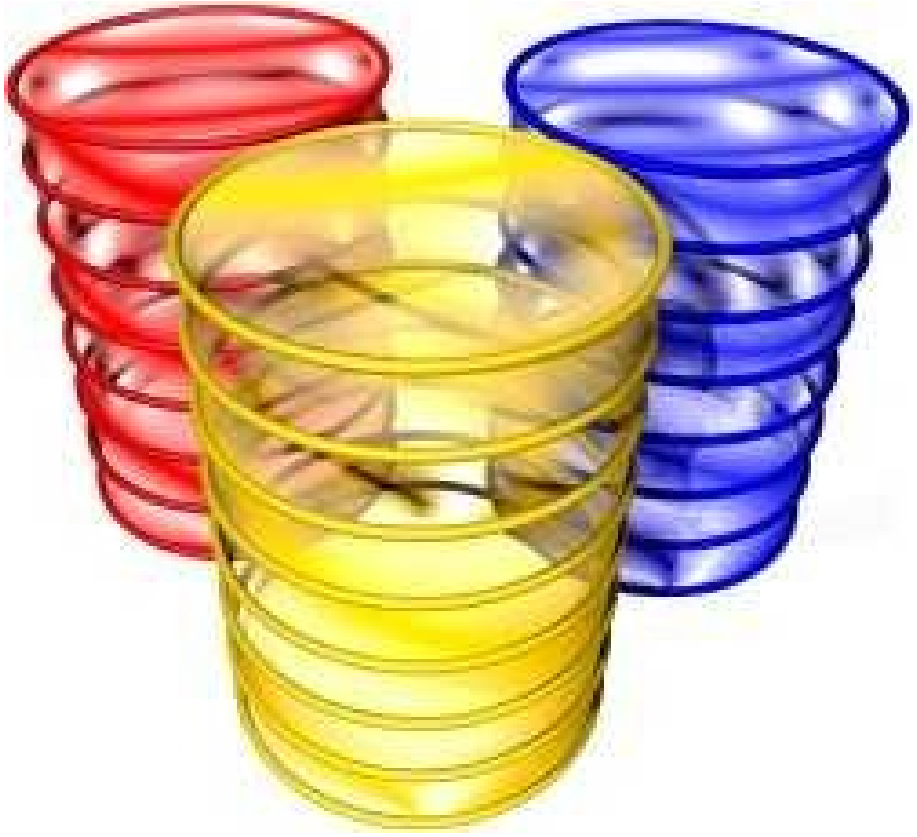
Replicación por Bitácoras (logs)

- Otro método comúnmente utilizado es el envío de logs (bitácoras de la base de datos).
- Este método presenta el problema que muchas veces dichas bitácoras no siempre son pensadas como elementos de intercambio para efectos de replicación.
- Este método generalmente se delega para porciones de la base de datos o ventanas de mantenimiento y sincronizaciones menores de bases de datos.

Métodos de Replicación

Replicación con Formatos Específicos

- Estos casos generalmente conllevan a configuraciones adicionales.



Balance de Cargas

No es lo mismo esto..

Balance de Cargas

.. que esto.

Balance de Cargas

- Generalmente asociado al concepto de replicación está el Balance de Carga para mejorar el desempeño de lectura.
- Algunas soluciones para High Availability incorporan balanceadores de carga dentro de sus características esenciales. Otros reposan esta responsabilidad sobre programas altamente dependientes de la plataforma de Sistema Operativo o programas de Terceros.

Consultas Distribuidas

Es prácticamente otro nombre que recibe el Balance de Cargas. Dentro de la documentación de PostgreSQL se le conoce como “Multi-Server Parallel Query Execution”

Divergencia

Encuentre las diferencias..

Divergencia

Encuentre las diferencias..

Divergencia

- Mantener la coherencia de los datos a través de múltiples nodos replicados es un proceso costoso debido a la latencia de red.
- Es así como algunos sistemas esquivan el costo por latencia permitiendo que los nodos diverjan levemente, lo que significa que es posible que se realicen transacciones en conflicto.
- Para retomar bases de datos coherentes y consistentes, estos conflictos deben ser resueltos cuanto antes de forma automática o manual.

Divergencia

- Estos conflictos rompen las condiciones ACID de los sistemas gestores de bases de datos, en cuyo caso las aplicaciones reposen bajo sistemas con divergencia deben estar al tanto e implementar sistemas de revisión.

ACID

- Acrónimo de los términos:
  - Atomicity
  - Consistency
  - Isolation
  - Durability
- Descrito en la norma ISO/IEC 10026-1: 1992 sección 4.

ACID

- Atomicity (Atomicidad) es la propiedad que asegura que la operación se ha realizado o no, y por lo tanto ante un fallo del sistema no puede quedar a medias.
- Consistency (Integridad) Es la propiedad que asegura que sólo se empieza aquello que se puede acabar. Por lo tanto se ejecutan aquellas operaciones que no van a romper las reglas y directrices de integridad de la base de datos

ACID

- Isolation (Aislamiento) es la propiedad que asegura que una operación no puede afectar a otras. Esto asegura que la realización de dos transacciones sobre la misma información sean independientes y no generen ningún tipo de error.
- Durability (Durabilidad) es la propiedad que asegura que una vez realizada la operación, ésta persistirá y no se podrá deshacer aunque falle el sistema.

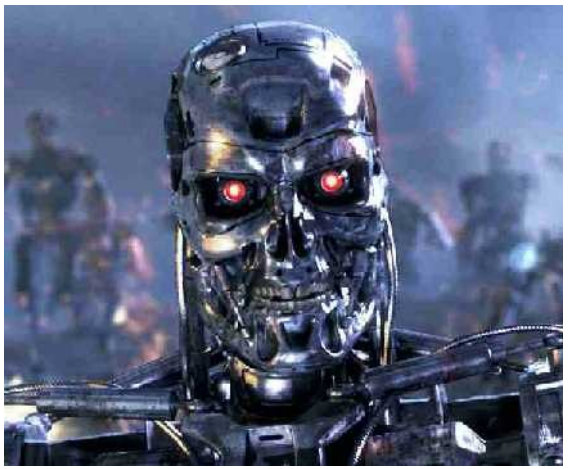
Tolerancia a Fallos

Ejemplos de Tolerancia y Recuperación de Fallos

Tolerancia a Fallos

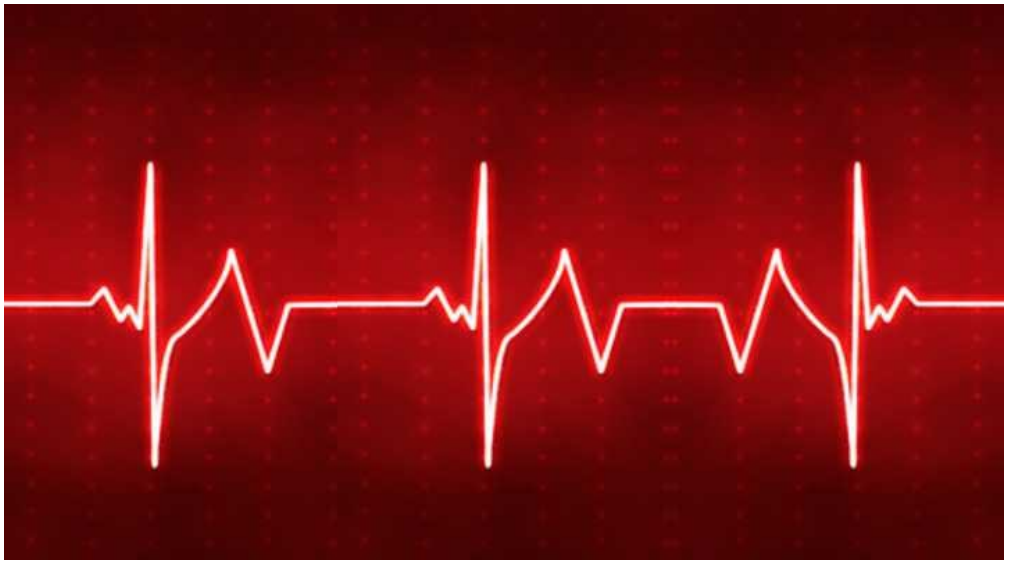
Venido de los términos en inglés Failover y Switchover, es la técnica que permite el cambio de forma automática entre nodos redundantes de un arreglo de computadores (o instancias de bases de datos).

Se prevé su uso para salvaguardar los servicios provistos por servidores y aplicaciones de la interrupción de forma anormal de los servicios de red, servidores y aplicaciones instalados en los nodos activos de un arreglo.



Tolerancia a Fallos

A nivel de servidores se configura igualmente validaciones de "Hearthbeat" (incluso en zonas geográficas distantes) para mantener sincronizados dos o más computadores, de forma de que si hay un fallo exista la posibilidad de hacer "hot" switch entre servidores y evitar el tiempo fuera (downtime).



Tolerancia a Fallos

- Algunos sistemas, de forma intencional, no realizan el failover completamente automatizado.
- mantienen un tiempo fuera de "solo lectura" hasta la intervención de un operario.
- Un tipo de failover automático con confirmación humana.
- Busca evaluar el impacto de la caída y elementos a recuperar al levantar los servicios.

Replicación Síncrona

Se realiza tan pronto como se realiza una transacción la misma se ejecuta en todos los nodos. Esto es muy costoso en términos de latencia y la cantidad de mensajes que se enviarán, pero evita la divergencia

Replicación Asíncrona

Los nodos del 'cluster' pueden aplicar los datos de transacciones en cualquier momento posterior, por lo que los nodos pueden servir diferentes simultáneamente distinta data.

Replicación Diligente

Normalmente en la documentación de Bases de Datos se utilizan como sinónimos de replicaciones Síncronas y Asíncronas. En principio toda replicación diligente o entusiasta es síncrona, pues se busca tener una copia exacta e indéntica en ca

Replicación Relajada

En el caso de las replicaciones relajadas el proceso de divergencia es controlado, es decir, se establecen un conjunto de reglas y parámetros que permitan una divergencia leve que no afecte el funcionamiento del proceso general.

Particionado de Datos

Particionado Vertical

Similar al proceso de normalización separa en tablas con menor cantidad de columnas los datos de una misma entidad.

Particionado Horizontal

Permite almacenar registros de diferentes tablas.

Particionado de Datos

- En sistemas Multi Maestro puede utilizarse para distribuir la trozos de data a través de los nodos.
- El uso de particionado requiere del uso de consultas distribuidas.
- Reduce la carga total de su “cluster”

Cluster

Cluster

- En general la definición de “Cluster” o “Clustering” es bastante vaga en la documentación
- Suele referirse al agrupamiento de servidores según la técnica que lo emplee.
- A cada servidor se le conoce como nodo.

Grid

- Nombre comercial para “Cluster”
- Normalmente montado sobre redes de conexiones redundantes, o no dirigidas o con interfaces sobre la Internet según el autor que se consulte.

Replicación de Discos

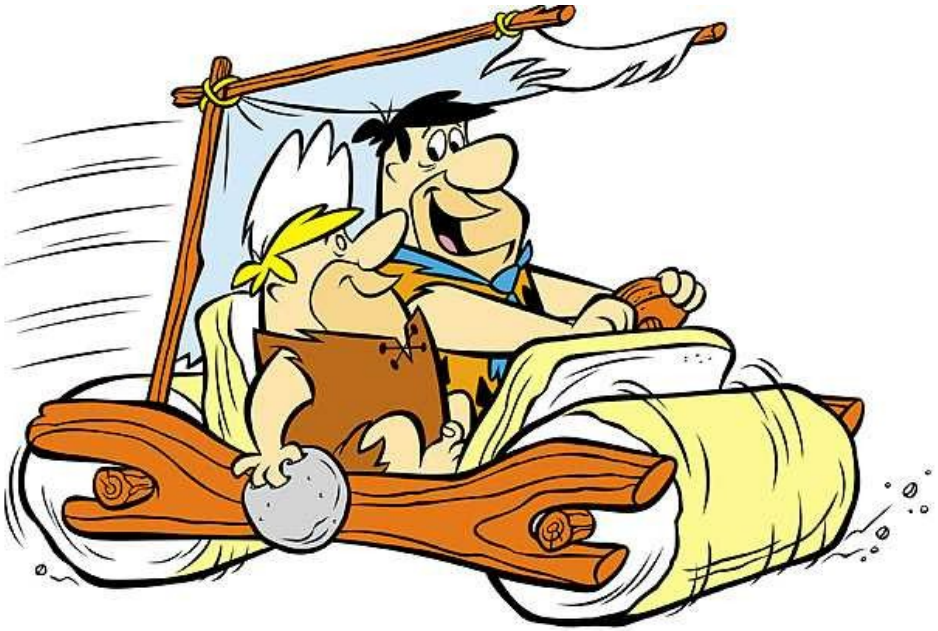
Forma donde se delega la replicación a componentes del sistema de archivo que se replican.

Puede ser una solución simple y elegante para abordar el failover por disco.

Shared-Nothing

- Arquitectura de Computación Distribuida donde se segmentan elementos de los servicios a servidores individuales (o conjuntos pequeños de servidores en cluster) que no comparten nada entre si.
- Ampliamente utilizado para atender aplicaciones de alto volumen de consultas y transacciones separadas por ámbitos

¿Y cómo hacemos todo esto con PostgreSQL?



¿Y cómo hacemos todo esto con PostgreSQL?

Alternativas de HA con PostgreSQL

Caracteristica	Disco Compartido	Caliente/Tibio en espera usando PITR	Trigger-Based Master-Standby Replication	Disparador basado en Maestro de espera	Declaración de middleware basado en la replicación	Replicación Asíncrona Multimaster	Replicación Síncrona Multimaster
Implementación más común	NAS	DRBD	PITR	SLONY	pgpool-II	Bucardo	
método de comunicación	Disco compartido	bloques de disco	WAL	Filas de tablas	SQL	Filas de tablas	Filas de tablas y filas bloqueadas
No requiere hardware especial		•	•	•	•	•	•
Permite multiples servidores maestros					•	•	•
No sobrecarga el servidor maestro	•		•		•		
No espera varios servidores	•		•	•		•	
Ante un fallo del maestro nunca se pierden datos	•	•			•		•
En modo de spera acepta consultas de sólo lectura			Solo calor	•	•	•	
Granularidad por tabla				•		•	
No es necesario la resolución de conflictos	•	•	•	•			•

Programas para HA con PostgreSQL

Programa	Licencia	Madurez	Metodo de Replicación	Sincronización	Pool de Conexión	Balanceo de Carga	Particionamiento de Consultas
PGCluster	BSD	Ver los detalles en la pagina web	Maestro-Maestro	Síncrono	NO	SI	NO
Pgpool-I	BSD	Estable		Síncrono	SI	SI	NO
Pgpool-II	BSD	Liberada recientemente		Síncrono	SI	SI	SI
Slony-I	BSD	Estable	Maestro-Esclavo	Asíncrono	NO	NO	NO
Bucardo	BSD	Estable	Maestro-Maestro Maestro-Esclavo	Asíncrono	NO	NO	NO
Londiste	BSD	Estable	Maestro-Esclavo	Asíncrono	NO	NO	NO
Mammoth	BSD	Estable	Maestro-Esclavo	Asíncrono	NO	NO	NO
Rubyrep	MIT	Liberada recientemente	Maestro-Maestro Maestro-Esclavo	Asíncrono	NO	NO	NO

Veamos ahora cómo
trabaja pgpool

Créditos

@carlosgr_arahat
@lennincaro
@leninmhs
@deglip
@gregoria126
@cnti

Fuentes:
documentación pgpool;
documentación proyecto postgresql-r;
Wikipedia