High availability with postgresql and pacemaker

High Availability Clustering with Pacemaker**

High availability (HA) is a critical aspect of modern IT systems, ensuring continuous operation and data integrity in the face of hardware or software failures. Pacemaker is an open-source cluster resource manager that plays a vital role in creating HA clusters for various applications, including PostgreSQL.

Enabling High Availability in PostgreSQL

To enable HA in PostgreSQL, you can use Pacemaker in conjunction with the PostgreSQL HA software stack. This stack includes components like DBD and Patroni, which manage data replication and failover between multiple PostgreSQL instances.

Creating a HA Setup with Corosync Pacemaker

To create a HA setup with Corosync Pacemaker, follow these steps:

1. Install Corosync and Pacemaker on the cluster nodes.
2. Create a Corosync cluster and configure its parameters.
3. Add the PostgreSQL instances to the cluster and define the resource definitions.
4. Enable resource fencing and set up the quorum parameters.
5. Start and test the cluster to verify proper HA functionality.

Is PostgreSQL High Availability?

Yes, PostgreSQL can be made highly available through the use of HA clustering tools like Pacemaker and the PostgreSQL HA software stack. This enables seamless failover between cluster nodes, minimizing downtime and data loss.

Difference between Corosync and Pacemaker

Corosync is a group communication system used for building clusters. It provides membership management, message passing, and synchronization services. Pacemaker, on the other hand, is a cluster resource manager that builds upon Corosync and manages the availability and failover of cluster resources, including PostgreSQL instances.

Checking High Availability in Cluster

To check the HA status of a cluster, you can use commands like "pcs status" or "crm status" to display the current cluster state, resources, and failover history.

Best Practices for a Highly Available PostgreSQL

- Use a cluster resource manager like Pacemaker.
- Implement data replication and failover mechanisms.
- Configure resource fencing and quorum parameters.
- Monitor the cluster health and performance.
- Perform regular backups and disaster recovery testing.

Why Use PostgreSQL over MySQL?

PostgreSQL offers several advantages over MySQL, including:

- Higher reliability and data integrity.
- Advanced features like asynchronous replication, materialized views, and stored procedures.
- Better performance for complex queries and concurrent workloads.

Does PostgreSQL Have Automatic Failover?

Yes, PostgreSQL and the PostgreSQL HA stack can provide automatic failover mechanisms. When one instance fails, the standby instance is promoted as the primary, ensuring continuous operation.

Role of Corosync in Pacemaker

Corosync provides the underlying cluster communication and synchronization infrastructure for Pacemaker. It ensures reliable message delivery, member detection, and heartbeat monitoring, which are critical for maintaining cluster health and failover coordination.

Pacemaker in Linux

Pacemaker is a high availability clustering solution widely used in Linux distributions. It provides resource management, failover, and recovery capabilities to ensure the continuous availability of critical services and applications.

PCSD in Pacemaker

PCSD (Pacemaker Cluster Services Daemon) is a crucial component of Pacemaker that runs on each cluster node. It monitors and manages cluster resources, performs recovery actions, and coordinates with other PCSD instances within the cluster.

Making Postgres High Availability

To make Postgres high availability:

- Install Pacemaker and the PostgreSQL HA stack.
- Create a Corosync cluster and define the Postgres instances as resources.
- Configure data replication and failover mechanisms.
- Test the cluster and monitor its performance.

PostgreSQL Speed and Limitations

PostgreSQL is generally faster than MySQL for complex queries and concurrent workloads. However, MySQL can be faster for simple queries and large datasets. PostgreSQL's limitations include potential performance bottlenecks in some scenarios and a higher resource footprint compared to MySQL.

Configuring Corosync Pacemaker

To configure Corosync Pacemaker, you need to:

- Install and start the Corosync and Pacemaker services.
- Create a Corosync cluster and configure its options.
- Define resource definitions for the services or applications you want to make highly available.
- Enable resource fencing and configure quorum parameters.

Best Pacemaker

The best Pacemaker available is the latest stable version maintained by the Pacemaker community. It offers the most up-to-date features, security patches, and performance improvements.

Pacemaker Equivalent

Pacemaker's equivalent in the Windows ecosystem is Windows Server Failover Clustering (WSFC). Both Pacemaker and WSFC provide similar high availability capabilities for managing cluster resources and ensuring service continuity.

High Availability Pacemaker Corosync

High availability pacemaker Corosync refers to the combination of Pacemaker and Corosync to create highly available clusters. Corosync provides the underlying cluster communication and membership management, while Pacemaker manages the resources and failover mechanisms.

Ensuring High Availability

To ensure high availability, you need to:

- Implement redundancy at all levels (nodes, storage, network).
- Use HA clustering solutions like Pacemaker.
- Monitor the system health and take proactive steps to address issues.
- Perform regular backups and disaster recovery testing.

Calculating High Availability

High availability is typically calculated as a percentage of uptime, considering planned and unplanned downtime. For example, a 99.9% HA rate indicates an annual downtime of about 52.56 minutes.

PostgreSQL Cluster Nodes

PostgreSQL can support up to 1000 nodes in a cluster, but the optimal number depends on the workload, performance requirements, and hardware capabilities.

PostgreSQL Clustering

PostgreSQL supports clustering through features like replication and logical decoding, which allow data to be distributed across multiple servers for high availability and scalability.

Making PostgreSQL Faster

To make PostgreSQL faster:

- Optimize queries and indexes.
- Use appropriate hardware and configuration settings.
- Consider using a PostgreSQL extension like pg_hint_plan.
- Monitor performance and identify bottlenecks.

PostgreSQL Power

PostgreSQL is known for its:

- Open-source and free nature.
- Robust data integrity and ACID compliance.
- Advanced features and extensibility.
- Large community support.

PostgreSQL Speed vs. MySQL

PostgreSQL is typically slower than MySQL for simple queries and large datasets. However, it excels in complex queries and concurrent workloads, outperforming MySQL in these scenarios.

PostgreSQL Use Cases

PostgreSQL is best suited for applications that require:

- High data integrity and reliability.
- Advanced features like materialized views and stored procedures.
- Scalability and high concurrency.

PostgreSQL Asynchronous vs. Synchronous

PostgreSQL supports both synchronous and asynchronous replication. Synchronous replication ensures data is committed to the standby instance before acknowledging write operations, while asynchronous replication allows write operations to be acknowledged without waiting for the standby to commit.

PostgreSQL Concurrency

PostgreSQL handles concurrency through MVCC (Multi-Version Concurrency Control), which allows multiple transactions to access and modify data concurrently without data corruption.

Performing Failover in PostgreSQL

In PostgreSQL, failover is performed automatically by the HA stack. When one instance fails, the standby instance is promoted as the primary, and applications are reconnected to the new primary.

Purpose of High Availability Clustering

The purpose of high availability clustering is to ensure continuous operation and data integrity in the event of hardware or software failures. It allows for seamless failover between cluster nodes, minimizing downtime and data loss.

Using Pacemakers

Pacemakers are used to treat slow or irregular heart rhythms. They send electrical impulses to the heart to maintain a normal heart rate and rhythm.

High Availability Mode

High availability mode is a configuration setting that allows applications or services to maintain continuous operation even when underlying hardware or software components fail.

High Availability in Security

In security, high availability refers to measures taken to ensure the continued availability of critical systems and services, such as firewalls, intrusion detection systems, and authentication mechanisms.

Highly Available vs. Failover

Highly available systems provide continuous operation, while failover systems switch to a backup system in the event of a failure. Highly available systems typically employ multiple redundant components and fault-tolerant mechanisms.

Clustering vs. High Availability

Clustering involves grouping multiple servers to create a single logical system, while high availability focuses on ensuring that critical services and applications remain operational even in the face of failures.

Disadvantages of HA

Disadvantages of high availability systems include increased complexity, higher costs, and potential performance overhead.

High Availability Examples

Examples of high availability include redundant servers, load balancers, backup systems, and clustering solutions.

Servers for High Availability

The number of servers required for high availability depends on the desired level of redundancy and the application requirements. Typically, a minimum of two servers is used for high availability setups.

High Availability in SQL Server

In SQL Server, high availability can be achieved through various mechanisms, including Always On Failover Cluster Instances (FCIs), Database Mirroring, and Availability Groups.

Building a High Availability Cluster

To build a high availability cluster, you need to:

- Identify critical applications and services.
- Select an appropriate high availability solution.
- Implement redundancy and fault-tolerant mechanisms.
- Configure failover and recovery strategies.

Why High Availability

High availability is important to prevent downtime and data loss, ensure business continuity, and meet regulatory compliance requirements.

High Availability vs. Scalability

High availability focuses on preventing downtime, while scalability refers to the ability of a system to handle increased workload by adding additional resources.

Load Balancer vs. High Availability

Load balancers distribute traffic

instructor 7th edition study guide coaching handbook an action kit for trainers and managers 2004 sr evinrude e tec 4050 service manual new from monastery to hospital christian monasticism and the transformation of health care in late antiq viper alarm manual override mind hacking how to change your mind for good in 21 days 2004 yamaha f115tlrc outboard service repair maintenance manual factory chap 16 answer key pearson biology guide nec dt300 handset manual bobcat parts manuals the shining ones philip gardiner 1992 1997 honda cb750f2 service repair manual download 92 93 94 95 96 97 regional economic outlook may 2010 western hemisphere taking advantage of tailwinds world economic and financial surveys sanyo lcd22xr9da manual edexcel maths past papers gcse november 2013 the riddle of the rhine chemical strategy in peace and war research project lesson plans for first grade apush unit 2 test answers pogil activity 2 answers kunci jawaban advanced accounting fifth edition fujifilm fuji finepix a700 service manual repair guide panorama4thedition supersiteanswers leccion8 toxicpeopletoxic people10 waysofdealing withpeople whomakeyour lifemiserable ftcemath 612 studyguidemfds studyguide section1guided readingand reviewwhat aretaxeschapter 14answer top10 plusone globalhealthcaretrends investmentsopportunities beyondtomorrowatkins physicalchemistrysolutions manual6e sthilms 180repairmanual johndeere model650 manualnoticia bombacomputer applicationsinpharmaceutical researchand developmentmarcy diamondelite9010g smithmachinemanual sustainablebusinessand industrydesigningand operatingfor socialandenvironmental responsibilitymanagerialeconomics theoryapplications andcases 8theditioncbr125r workshopmanual1999 volvoowners manuaphilip ecgsemiconductormaster replacementguide guitarwiringmanuals gmservice manual97 jimmydictionary ofchristian loreandlegend inafixasm studymanualexam p16thedition eqshophpcp4025 partsmanualmy beforeand afterlife corporatefinanceexam questionsand solutions1996yamaha 8hp outboardservice repairmanualdungeons anddragons4th edition2011rogue serviceandrepair manualreformand regulationof propertyrights propertyrights inamerican historyfromthe colonialerato thepresentthe christianfoundationor scientificandreligious journalvolume ino9 september1880perkins 3cylinderdiesel enginemanual universityphysics 13thedition schoolscavenger huntclues jvcservice orquestions manual