High Availability Database Solutions

for PostgreSQL & Postgres Plus

An EnterpriseDB White Paper

for DBAs, Application Developers and Enterprise Architects

November, 2008

http://www.enterprisedb.com

EnterpriseDB®
The Enterprise Postgres Company

Executive Summary

Competition and around the clock demand for information challenges IT to ensure mission critical databases are highly available (HA).

High Availability (HA) solutions enable IT to deliver measurable value in a variety of forms, including:

- Seamless failover for unplanned system component failures
- Disaster Recovery with minimal service interruption
- Optimized performance for both OLTP and Query/Reporting

This paper discusses the failover and redundancy capabilities of specific HA products that can be used in conjunction with PostgreSQL, Postgres Plus, and the Oracle compatible database Postgres Plus Advanced Server.

HA options for Postgres Plus provide enterprises with a range of optimized technology choices, for their IT infrastructures, including the replication of query intensive data from Oracle to Postgres Plus.

An in depth discussion targeted specifically to your organization's high availability requirements can be scheduled with an EnterpriseDB domain expert by sending an email to sales@enterprisedb.com.

High Availability

Today's high speed interconnected world requires IT to flawlessly deliver critical information to employees, customers and partners. For outward-facing applications in particular, system downtime is simply not an option. Competition is fierce and today's browsing public has zero tolerance for any business outages or delays. Furthermore, globally situated users demand continuous access to a wide range of information assets including those for self-service and business intelligence purposes. High Availability solutions enable organizations to maximize data accessibility, reliability, and performance across all tiers of the information architecture.

The primary goals of a High Availability infrastructure are to:

1. Minimize downtime during normal operations
2. Recover quickly while also maintaining integrity in a variety of disaster situations
3. Offload processes to low cost commodity hardware and software
4. Scale seamlessly to accommodate business growth

To achieve these goals, HA infrastructures are characterized by redundancy and failover features that reduce or eliminate potential points of failure.

Postgres Plus offers several failover and redundancy configurations utilizing a combination of open source and proprietary product solutions. These configurations include Active/Passive clustering using various cluster software solutions, Active/Active clustering using replication, and disaster recovery via standby databases.

All of these capabilities can be configured to minimize downtime. In fact, in a recent customer implementation, Postgres Plus Advanced Server, EnterpriseDB's Oracle compatible database, was installed in a Red Hat 5 Cluster Suite environment to achieve automated failover in the range of 20-30 seconds, thus significantly reducing overall database downtime.

.

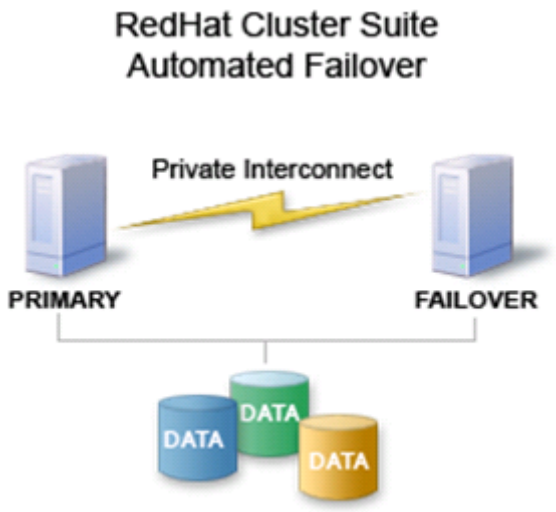

Figure 1
Active/Passive Cluster Configuration

HA configurations require a combination of investments that must balance business needs against implementation and management costs to decide what mixture is right for each class of applications. The most common Postgres Plus HA configurations are discussed in more detail below.

Active/Passive Failover Clusters

The basic components of an Active/Passive configuration, as shown in Figure 1 above, are two servers, shared storage, along with network and clustering software. The servers can be two physically separate machines or "slices" of a single machine (such as a VMware image or logical partition). One of the servers is considered the active 'in use' node while the other is the passive or 'idle' node.

The shared storage is connected to both servers so each server can, at the appropriate time, mount any of the file systems on the shared storage.

Some of the most popular cluster software available today includes products such as Red Hat Cluster, Veritas Cluster and Linux-HA. The cluster software is responsible for monitoring the health of the cluster and also for taking action when a failure is detected. The cluster software utilizes the inter-connect link to send a heartbeat signal between each server.

The continuous successful receipt and acknowledgement of this heartbeat means that nodes in the cluster are healthy and no action is needed. Failure to receive a heartbeat signal indicates a problem within the cluster and triggers a failover process. The failover process then shuts down the failed node, mounts the file systems on the passive node, and restarts the services on the passive node making it the new active node. In HA architectures, failover can complete in a matter of seconds.

In the above scenario, Postgres Plus runs on the active server. The cluster software is configured to start Postgres Plus' services on the passive node when the active node fails. Upon completion of this failover, the passive server becomes the active server and Postgres Plus is ready to continue database processing on this new active server.

This scenario illustrates single purpose resources in a high availability configuration. One server is used to meet the enterprise's normal needs while a second idle server is used for failure situations. As mentioned earlier, the entire failover process can take as little as 20-30 seconds, minimizing down time while also ensuring business continuity.

While idle resources in an Active/Passive architecture may be considered inefficient, they do guarantee that the organization's desired quality of service (QoS) will be maintained once the failover process is complete.

Active/Passive strategies make sense when high QoS is paramount to the enterprise's mission and budgets allow the needed resources for implementation.

Active/Active Failover Clusters

A common variation of the Active/Passive architecture is to utilize both servers in the cluster to handle more than one application database, thereby converting the Active/Passive cluster into an Active/Active cluster.

Consider the following scenario:

- Application F supports your Finance department and utilizes Postgres Plus as its database.
- Application H supports your Human Resources department and utilizes Postgres Plus Advanced Server for its database.
- You have a two node cluster using Red Hat Cluster for cluster management.
- During normal operating conditions, these two applications must have their own resources (i.e., they cannot contend for resources on a single machine).

You can configure an Active/Active cluster to provide high availability and failover for both applications in the following manner:

- Install the Finance database on Server F with its data files residing on the shared storage in a volume called Finance
- Install the HR database on Server H with its data files residing on the shared storage in a volume called HR.
- Server F mounts the Finance volume
- Server H mounts the HR volume
- Application F connects to Server F to conduct its business
- Application H connects to Server H to conduct its business
- The cluster software is configured to restart the Finance database on Server H if Server F fails
- The cluster software is configured to restart the HR database on Server F if Server H fails

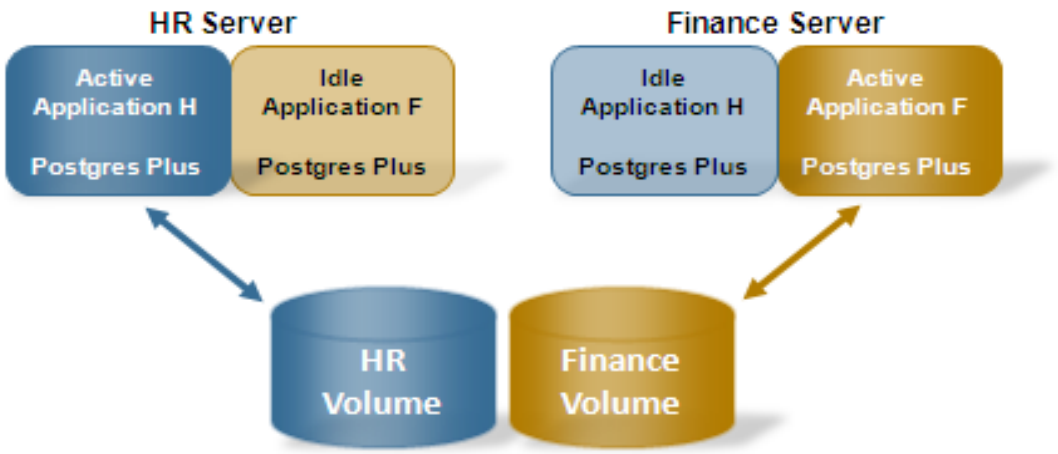

Figure 2
Active/Active Cluster Configuration

The scenario above illustrates a dual purpose high availability configuration where each server is active for its primary application and acts as a failover server for its secondary application.

In contrast, with an Active/Passive architecture, where the passive server is idle until it is activated by a failover process, Active/Active architectures make more efficient use of available computing resources. However, in failover states, multiple applications may contend for resources on the now-shared server.

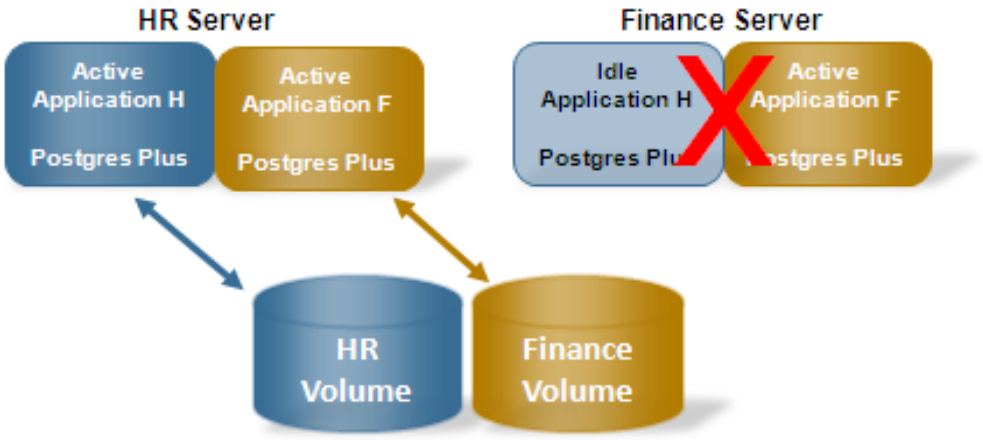

Figure 3
Active/Active Cluster in Failover Mode

Active/Active strategies make the most sense when computing resources must be economized and users can tolerate temporary performance degradation while a downed server can be brought back online.

Active/Active clustering is a proven, reliable and relatively inexpensive strategy that offers fast failover times that ensures operating continuity while reducing idle resources.

Disaster Recovery Using Standby Databases

The Postgres Plus product family offers multiple disaster recovery alternatives. One solution, database replication, is discussed in detail later in this document. Another solution is Postgres Plus' standby database capabilities.

In a standby database configuration, a production database ships its transaction logs to a remote site where the transactions are applied to a database running in recovery mode. The primary database runs in archive log mode. On a defined schedule, the transaction Write Ahead Logs (WAL) are copied to the remote location and applied to the standby database. Recovery mode simply means that the standby database is closed to user access so the WAL files can be applied. Once the log files are applied, a standby database is completely synchronized with the production database and can be brought online in the case of a failure.

The Postgres Plus pg_standby utility automates database updates via transaction log files and ensures the logs are applied to the remote database as soon as they become available.

In the event of a site failure, the standby database can be brought up in full production mode and applications can access the “new” production database with minimal disruption to business operations.

HA Architectures and Replication

Database replication is a strategy that maintains real time or near real time up to date copies of production data on different hosts, possibly in different physical locations. Database replication is a critical component of HA architectures because it can provide the data redundancy necessary for failover clusters, disaster recovery, as well as optimized OLTP and reporting performance.

The Postgres Plus product family provides several replication options as summarized below:

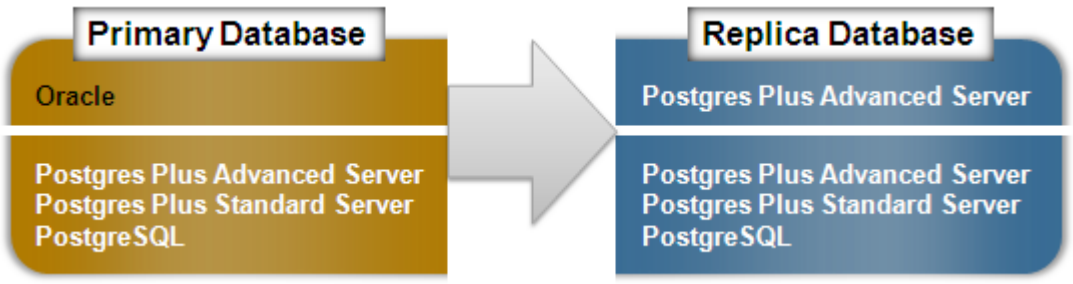

Figure 4
Postgres Replication Options

Postgres replication is transactional in nature and uses a push-pull architecture. All Postgres Plus replication configurations are trigger based. Let’s now examine some of the replication options available to you.

Postgres Plus to Postgres Plus Replication Using Slony

Postgres Plus to Postgres Plus replication uses an open source replication technology called Slony. Slony replication is asynchronous, trigger based, transactional replication. You define a master to N slave(s) replication relationship. The concept of cascading replication is also supported (i.e. the master node replicates data to a slave node, which in turn, replicates to other slave nodes).

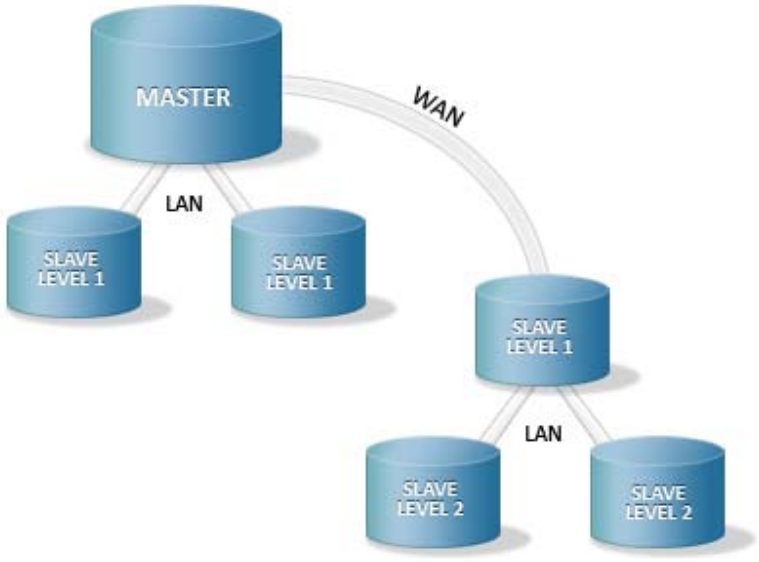

Figure 5
Postgres Plus Slony Replication

The Slony architecture is flexible and allows for replication of the entire database or a subset of the database tables. Graceful switchover (i.e., a slave becomes the master and the master becomes a slave) is supported, providing an effective means of performing OS or database upgrades without downtime.

If the master node fails, one can perform a failover where a slave becomes the master and will begin replicating data to all of the other slaves. A failover differs from a switchover in that the failed node is removed from the replication cluster completely. In order to bring that node back into the replication cluster, one must re-instantiate the node using the *add node/subscriber* capabilities of Slony.

Oracle to Postgres Plus Replication Using Postgres Plus Replication Server

Postgres Plus Replication Server provides bidirectional replication services between Postgres Plus and Oracle. Like Slony replication, Postgres Plus is an asynchronous, transactional, trigger based replication solution.

Postgres Plus Replication Server is based on a publish and subscribe architecture. The source database publishes changes to a replication server. Subscribers can subscribe to the data they wish to receive, and the replication server ensures that every subscriber receives the data it

needs. In the figure below, it is noteworthy that all communications between the management console and the pub/sub tier use RMI, and all interactions with the database tier use JDBC.

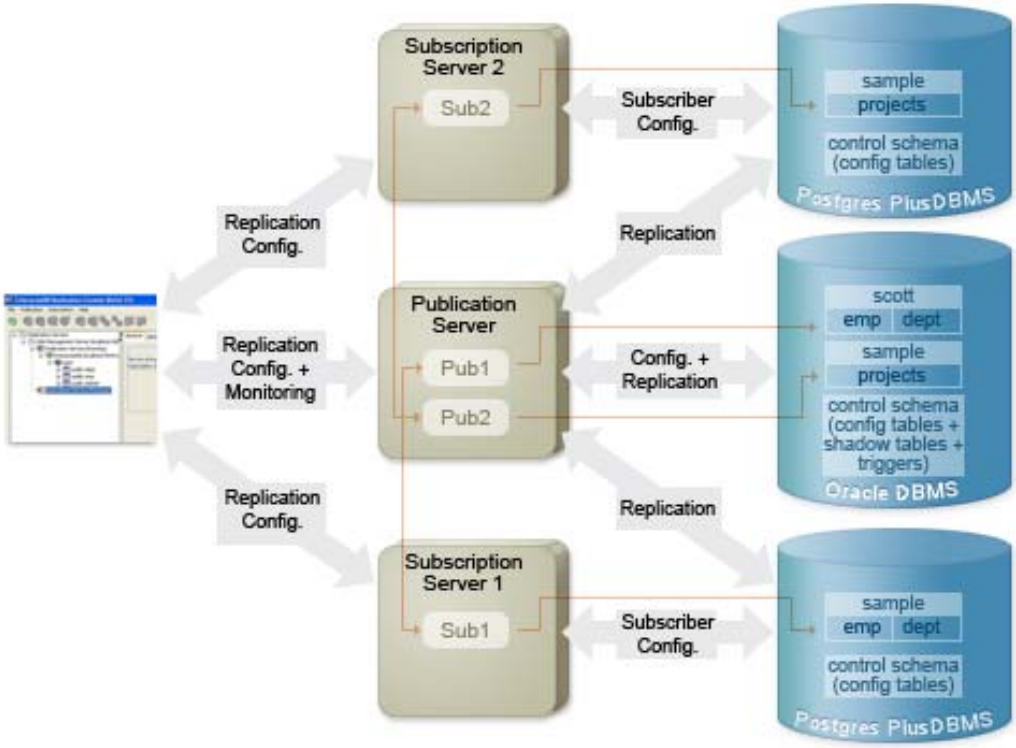

Figure 6
Oracle and Postgres Plus Advanced Server Replication

Oracle to Postgres Plus replication not only serves failover and disaster recovery needs, but it is also a highly effective strategy for balancing transactional and reporting workloads. The key benefits of using Oracle to Postgres Plus replication include:

- Production System Performance. Query-intensive reporting applications executing off a replicated database do not degrade the performance of the primary OLTP systems.

- Database Optimization. OLTP and reporting databases can be independently tuned to optimally support their now independent transactional and inquiry workloads.

- Problem Isolation. Databases are better isolated from common application problems. For example, an ad hoc runaway query will not bring your OLTP database down.

Many organizations choose to replicate from Oracle to Postgres Plus to achieve significant cost savings (see the FTD case study below). Although they continue to run their transactional systems on Oracle, they find it much more cost effective to run their reporting applications on Postgres Plus. And since Postgres Plus includes a robust suite of Oracle-compatible data types, views, interfaces, and tools, the organization's DBAs and application developers experience minimal learning curves in adopting Postgres solutions.

FTD Field Study

FTD, the international florist icon, was faced with a serious dilemma. Their reporting systems were severely impacting their ability to process online orders. FTD originally considered using Oracle for the reporting system but, due to the high cost and lack of support/attention given to them by Oracle, they chose to use Postgres Plus Advanced Server instead.

In a matter of weeks, FTD had implemented the Oracle to Postgres Plus Advanced Server replication and performed extensive stress testing of the system.

FTD was able to quickly and easily integrate their reporting technology, Oracle Reports, with Advanced Server. As a result, FTD's analysts and users could begin using the new Advanced Server database with no additional learning.

The system went live in time for one of FTD's peak ordering seasons and was a huge success. Now they are looking for other areas to utilize Postgres Plus Advanced Server. In addition, they saved over $100,000 by pursuing this strategy.

Additional HA Options for PostgreSQL and Postgres Plus

There are several other high availability failover and replication product options for PostgreSQL and the Postgres Plus product family. They include both open source and commercially available products.

pgpool and pgpool II

pgpool and pgpool II are both open source technologies that provide additional HA features. pgpool provides connection management and connection pooling capabilities that can be used to achieve connection failover in the HA configurations discussed earlier.

pgpool II combines pgpool's connection management, pooling and failover capabilities with a replication facility. There are two modes to replication support in pgpool II:

Master/Slave mode – this mode relies on other replication technology, such as Slony, to handle the actual database replication. In master/slave mode, pgpool II routes all DML statements to the master node and, if desired, load balances queries across all nodes in the configuration. pgpool II initiates failover processes if a node fails; thus, providing automated failure recovery.

Replication mode – in this mode, pgpool II acts as the replication agent. When a transaction is issued, pgpool II replicates that transaction to each node in the configuration; thus, maintaining multiple copies of the database. If one node fails, pgpool II allows processing to continue as long as there is at least one other surviving node in the configuration.

PGCluster

PGCluster is another open source HA solution. It provides an Active/Active replication based HA solution and includes load balancing functionality as well.

PGCluster is comprised of 3 primary components:

- Load Balancer Server
- ClusterDB Server
- Replication Server

Using these components, one can configure PGCluster as a load balancing cluster or as a high availability cluster as shown in the diagram below.

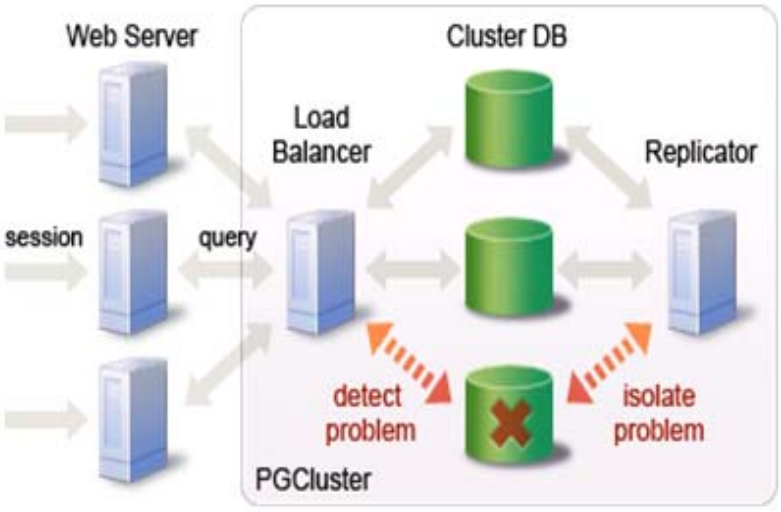

Figure 7

Continuent's UniCluster and Sequoia

UniCluster is a commercial product that provides a virtual database view of multiple PostgreSQL or Postgres Plus databases. UniCluster replicates all transactions to every node in the cluster; thus, ensuring continuity in the event of a node failure.

Sequoia is the open source version of UniCluster. Both solutions provide replication, load balancing and failover capabilities.

Bucardo

Bucardo is a multi-master replication system for PostgreSQL. It is an asynchronous, trigger based technology that allows multiple masters and multiple slave configurations. Bucardo is based on Perl and provides the ability to do conflict detection, resolution and exception handling via custom Perl routines.

The Bucardo architecture is comprised of a Perl daemon and master Bucardo database that contains all the information about the databases involved in the replication configuration.

Conclusion

Given the critical need for highly available architectures in today's enterprise information systems, it is no surprise that a variety of High Availability options providing failover and redundancy co-exist in the market today.

Numerous HA product options and solutions are available for the Postgres Plus family of databases. Whether you are looking for fast failover and high availability of an application through Active/Passive clustering, or a HA technology that also optimizes reporting capabilities via replication, there is a Postgres based solution for you.

You can choose a single Postgres compatible technology or a combination of solutions to best satisfy your business needs and high availability requirements. In particular, some Postgres replication options provide Oracle installations with economic as well as technical choices that can significantly benefit your enterprise's budget.

As Postgres experts, EnterpriseDB can help you evaluate your needs, define an appropriate HA architecture, and select the technologies that best meet your requirements. Our experts can also help you implement the selected infrastructure components and provide your team with ongoing assistance. We offer both Replication Setup and High Availability Setup packaged services, and can tailor any service offering to meet your specific objectives.

For more information about these services, please visit:

http://www.enterprisedb.com/tservices/professional_services.do

About EnterpriseDB

EnterpriseDB is the leading provider of enterprise class products and services based on PostgreSQL, the world's most advanced open source database. The company's Postgres Plus products are ideally suited for transaction-intensive applications requiring superior performance, massive scalability, and compatibility with proprietary database products. EnterpriseDB has offices in North America, Europe, and Asia. The company was founded in 2004 and is headquartered in Edison, N.J. For more information, please call +1-732-331-1300 or visit www.enterprisedb.com.