



A COMPARISON OF THE EDB POSTGRES™ PLATFORM TO SELF-SUPPORTED PostgreSQL

THE KEY DIFFERENCES

Open source databases are becoming an increasingly popular option for organizations seeking high performance yet lower cost alternatives to legacy databases. According to a report by Gartner Research, by 2018, more than 70% of new in-house applications will be developed on an open-source DBMS. ¹

As organizations explore these new open source alternatives, PostgreSQL is becoming an increasingly popular choice. This popularity was reflected in DB-Engines.com (a leading source of database usage research) naming Postgres “DBMS of the Year” for 2017.

Like all database management systems, PostgreSQL requires additional enterprise tools and capabilities to ensure high availability at scale. These include additional tools for backup, disaster recovery, replication, monitoring and data migration.

To meet these needs, EnterpriseDB created the EDB Postgres™ Platform, which adds additional features, capabilities and support around the PostgreSQL database including:

- Increased security
- Enterprise-ready tools
- Oracle® compatibility features
- More deployment options
- Enhanced integration
- Enhanced DBA and Developer productivity
- PostgreSQL community leadership
- 24x7 “Follow-the-Sun” support

This document explains the key differences between PostgreSQL using the EDB Postgres Platform compared to self-supported PostgreSQL alone.

1 - GARTNER, EMERGING TECHNOLOGY TRENDS CREATE OPPORTUNITIES FOR DBMS COST OPTIMIZATION, PUBLISHED: 21 APRIL 2016

PostgreSQL

PostgreSQL is a relational database management system (DBMS) that has been an active open source project since 1996. It is one of the oldest and most stable open source projects as a result of the commitment of its members and its independence as a stand-alone community.

The community consists of a wide variety of individuals from many different backgrounds. Some are from PostgreSQL consulting businesses, software vendors, and large organizations who sponsor individuals to work on PostgreSQL. EnterpriseDB is one of the largest employers of PostgreSQL community members. Two of the five core PostgreSQL team members, seven committers and a much larger number of contributors are EDB employees.

Feature suggestions and code may be submitted by anyone, but the core team members and source code committers ultimately make the key decisions for each release. PostgreSQL has a reputation for high quality, standards-based code, extensive security features, excellent documentation, and application for a wide variety of workloads including unstructured data like JSON documents and key-value stores. New releases are published yearly, regular updates are provided quarterly and ad hoc security or maintenance updates are published as needed.

Users can freely download source code and build the database binaries themselves. If users prefer not to build the binaries themselves, they can either obtain free PostgreSQL from a number of packagers who include it in their software distribution (e.g. Red Hat Enterprise Linux) or from easy to use RPM or GUI installers.

The PostgreSQL software license is quite liberal and allows use without cost or signed agreements. It also allows modification and redistribution without the need to contribute changes back to the community.

The EDB Postgres Platform

The EDB Postgres Platform™ starts with the open source PostgreSQL database system and adds enterprise-class features and tools. These critical components are used by thousands of organizations worldwide to reliably operate Postgres at scale while minimizing risk.

The platform consists of two database options:

- PostgreSQL
- EDB Postgres Advanced Server (EPAS)

For readability, in the remainder of this paper, the term “Postgres” refers to either the PostgreSQL or EPAS databases. EPAS is a continuously synchronized commercial derivative database based on PostgreSQL, containing all the features and fixes of the open source database, plus added enterprise features for security, performance, and database compatibility for Oracle.

Why does EDB include two databases in its platform?

- Some companies mandate use of open source software
- To provide choice depending upon the needs of any given application or organization
- To provide additional enterprise features and capabilities not available in standard PostgreSQL

Surrounding the database in the EDB Postgres Platform are tools developed, maintained, packaged, distributed, supported and documented by EDB. These help organizations with management, monitoring, tuning, high availability, disaster recovery, and database replication needs. Enterprises also have access to data adapters for integrating other relational and non-relational databases. These powerful tools surrounding the database are one of the biggest differences between using self-supported PostgreSQL and the EDB Postgres Platform.

The EDB Postgres Platform also facilitates a wider array of database deployment options including bare metal, virtual

machines, containers, private cloud, public cloud, and hybrid cloud. In addition, EDB Postgres™ Ark DBaaS software deploys and manages databases in multiple public or private clouds while integrating into existing DevOps and ITSM environments.

And finally, the EDB Postgres Platform includes services such as 24x7 technical support, training and certification, along with a full menu of professional and remote DBA services.

THE ADVANTAGES

This section identifies the advantages of using the EDB Postgres Platform in comparison to operating a PostgreSQL database in a self-supported model (i.e., not partnered with a commercial entity).

1. INCREASED SECURITY

Enhanced Auditing

EPAS includes additional built-in auditing features that capture more detailed data about who did what and when in the database. Session tag auditing in EPAS allows developers to capture middle tier data that goes beyond the generic database user IDs in PostgreSQL that may be shared in connection pools.

EPAS can capture actual application level user IDs and log them for greater accuracy and compliance with data governance rules. The EDB auditing feature, which is easily enabled and configured via database parameters and related commands by authorized administrators, can be configured to write audit log entries to a location that is separate from the normal database system log. This allows access to the actual audit logs to be restricted to a smaller set of authorized users, enhancing the security of this sensitive information.

Password Policy Management

Password management is another major security

difference between PostgreSQL and EPAS. In direct response to a customer feature request (a top US-based credit card processor), EDB added database integrated password policy management capabilities for passwords. These govern password strength, lockout times, password lifetime, grace periods, and more. This allowed the organization to strengthen their PCI-DSS compliance.

SQL Injection Protection

In addition, self-supporting PostgreSQL users don't have ready-made protection against SQL injection attacks where malicious SQL code can be entered into application data entry fields that damages or destroys data. SQL/Protect has been a standard part of the EDB Postgres Platform for years and provides centralized DBA managed protection against a variety of SQL injection vectors, so organizations don't need to rely solely on individual developers to consistently write bullet-proof code.

Security Certifications

EPAS (the enterprise version of PostgreSQL in the EDB Postgres Platform) is the first PostgreSQL-based RDBMS to receive a Security Technical Implementation Guide (STIG) from the US Defense Information Systems Agency (DISA) after a year of collaboration. Read the memo at:

https://iasecontent.disa.mil/stigs/pdf/U_EDB_Postgres_Advanced_Server_9_STIG_V1_Memo.pdf

EPAS is also the only PostgreSQL-based distribution on Windows that has FIPS 140-2 certification (a U.S. government computer security standard used to approve cryptographic modules). Find the certificate at:

<https://csrc.nist.gov/Projects/Cryptographic-Module-Validation-Program/Certificate/3130>

The EDB Postgres Platform also has tools and techniques under development to help make Postgres databases compliant with the EU's General Data Protection Regulation (GDPR). While PostgreSQL offers a rock solid foundation of security, the EDB Postgres Platform has certain additional features that are not available in

self-supported PostgreSQL, making the EDB Postgres Platform a more efficient and effective fit for GDPR compliance. Further details on these features are described in an ebook, accessible from this link: <http://info.enterprisedb.com/GDPR-for-DBA.html>

Row Level Security (RLS)

RLS is useful for multi-tenant applications or simply where different user populations must be restricted from accessing each others data.

In direct response to customer requests, EDB added this feature to EPAS many years ahead of its addition to PostgreSQL, and did so in an Oracle-compatible fashion as Virtual Private Database. As a result, users did not have to wait as long for inclusion in PostgreSQL, and could work directly with EDB to get this new feature implemented in line with their time requirements.

In contrast with PostgreSQL's RLS implementation, any user with Oracle experience will find EPAS' RLS easy to use because it is compatible with Oracle and easier to maintain because it:

- Is policy based, and more flexible, in that it can be applied only if certain columns are accessed
- Can be leveraged against application users

2. ENTERPRISE-READY TOOLS

Enterprise Management Tools

The EDB Postgres Platform has a significant advantage over self-supported PostgreSQL when an organization explores essential tools around the database for high availability, backup and disaster recovery, replication and database management, monitoring, and tuning. The EDB Postgres Platform's tools are born directly out of the requirements of EDB's large enterprise customers, cover more use cases than community tools (specifically in heterogeneous replication and DBaaS), and provide a one stop shop for best of breed capabilities.

In contrast, a self-supporting PostgreSQL user relying solely on the "free" PostgreSQL ecosystem must navigate multiple independent projects for management, monitoring, tuning, replication, connection pooling, and high availability tools. There may also be multiple projects within a single tool domain (i.e. backup). This project dispersion presents a number of challenges for the self-supporting PostgreSQL user. First, one must spend time evaluating all the different options and choose the best tool. Second, upon the new release of any tool, it must be tested against the user's other tools. And third, testing, support, release schedules, and documentation will be different among each tool project, making general management across all the tools difficult.

EDB's tools work with both PostgreSQL and EPAS, making for easy operation and maintenance at installations where both databases are in use. EDB also deploys and tests its tools in an integrated fashion with every release and provides consistent support, release coordination, and documentation. Finally, at the time of this writing, the only Database-as-a-Service (DBaaS) available for deploying Postgres to private and public clouds in the PostgreSQL ecosystem is EDB's Ark DBaaS software.

EDB Postgres Enterprise Manager

pgAdmin is arguably the most popular tool for managing individual PostgreSQL databases. Postgres Enterprise Manager (following the EPAS model) is built upon and continuously synchronized with pgAdmin and adds the following features requested by EDB enterprise customers:

- Provides a single pane of glass for monitoring, alerting and general management of large numbers of Postgres databases
- 13 pre-designed dashboards
- Custom dashboards with drill down features, like time and data zooming, to bring critical data to light quickly
- Auto-discovery of managed servers that reduces setup time
- 225 pre-configured data probes put best practice monitoring and alerts to work immediately

- SMTP/SNMP support with prioritization and schedule-awareness alerts the right people with the right information at the right time
- Integration with Nagios for more advanced monitoring
- Custom probes monitor and alert specific instrumentation for unique application needs
- Custom alerts included for log files
- Index Advisor automatically finds and creates new indexes to optimize slow queries and integrates with the SQL Profiler
- Tuning Wizard gets the most performance out of hardware, operating system, and database workload profile
- Bulk operations against multiple databases at once
- Log Manager configures log settings in bulk, saving time and increasing standardization
- Log Analysis Expert reports log statistics on long running queries, unique queries, time spent running queries and locking statistics
- Session Wait Analysis dashboard simplifies performance diagnosis and pin-points bottlenecks
- Team Manager segregates access to managed servers by users and roles
- Capacity Manager forecasts storage needs for more efficient data center planning

Similarly to PostgreSQL, pgAdmin provides a great foundation upon which EDB has added a great variety of essential enterprise capabilities to make managing large scale installations of Postgres easy.

EDB Postgres Failover Manager

EDB Postgres Failover Manager creates highly available fault tolerant clusters, minimizing database downtime with health monitoring, failure detection/notification, downed node fencing (STONITH), and automatic failover. A simple and lightweight architecture protects against a variety of failure scenarios, while allowing end users to customize for specific application needs.

Key features include:

- Support for multi-server clusters for desired redundancy
- Architected with no single point of failure

- Easy to install, configure and use, resulting in quick implementation of high availability
- Configurable health monitoring and witness architecture prevent unnecessary or false failovers
- Configurable failure detection and failure actions support high availability requirements
- Prevents "split brain" scenarios guarding against potential data corruption
- Supports automatic failover to a standby replica for hands-free resolutions
- Postgres fast binary streaming replication keeps failover replicas up-to-date
- Administrative switchover and switchback provides complete cluster control
- Controlled switchover/switchback provides near zero downtime maintenance and testable configurations
- Pre and post promotion fencing scripts prevent suspect servers from re-joining a cluster, and causing possible data loss
- Fencing operations support virtual IPs and load balancers for maximum flexibility
- Automatic email notifications keep DBAs informed of all status/health changes

Configuration options make integration with existing environments easier. In addition to sending email notifications, EFM can trigger customer-supplied scripts to send notifications to real-time chat rooms or via SNMP. Additional trigger points allow DBAs to create a wider range of instructions to be executed in the event of a failure.

The PostgreSQL ecosystem offers a variety of options that are easy to find by an Internet search for "PostgreSQL automatic failover". Then the self-supporting PostgreSQL user needs to inventory and evaluate the feature differences between each one to be sure it will meet their specific needs.

EDB Postgres Backup and Recovery Tool (BART)

EDB Postgres Backup and Recovery Tool safeguards local and remote data by eliminating error prone scripting and manual tasks with simple commands, a command line

interface, centrally managed catalog, point-in-time recovery (PITR), retention policies, compression and other automated capabilities.

Key features include:

- Backup/restore of local and remote databases to protect all data
- Hot physical backups don't interrupt user operations
- Supports base backups and continuously-archived WAL segments for fine grained control over recovery timelines
- Block-based incremental backups speed up recovery, and reduce storage needs
- Compression reduces network traffic and needed storage space
- Centrally managed catalog provides easy reporting and recovery
- Retention Manager supports redundancy copies or recovery window policies reducing cleanup maintenance time
- Point-in-time recovery options minimize data recovery and recreation
- Automatic table space support reduces a DBA's workload
- Command line interface integrates with existing CRON jobs
- Friendly user-defined backup names

Again, the PostgreSQL ecosystem offers a variety of options discoverable with an Internet search for "PostgreSQL backup tools". A similar exercise as would be done for high availability tools also needs to be completed.

EDB Postgres Replication Server

EDB Postgres Replication Server provides solutions to a wide variety of data center issues in one package. It moves data in bulk or copies transactions in near real-time from one database to another. Support for different database types, flexible configuration options, custom data handling features and a graphical management interface allows DBAs to provide customized data movement/exchange solutions easily and quickly.

Currently available tools for self-supporting users in the PostgreSQL market lack the out-of-the-box capabilities present in EDB Postgres Replication Server.

EDB Postgres Replication Server can:

- Operate in single or multi-master modes
- Exchange data with Oracle and Microsoft® SQL Server to prevent buying extra licenses
- Exchange data with on-premises and cloud-based environments
- Offload reporting tasks from a master database, improving performance
- Copy data for backups and safekeeping
- Feed data to test the stability of a new database software version
- Seed new database applications with real time data
- Constantly move data to local or remote reporting servers, avoiding slow ETL processes
- Be used for testing application upgrades or staging systems with real data

General EDB Postgres Replication Server features include:

- Log-based binary replication engine keeps latency low and supports more replicas
- Publish - subscribe architecture supports modern distributed data centers
- Offline schema/snapshot replication or online continuous replication gives DBAs more options
- Table-based replication allows exchanging subsets of a database
- Row filters only replicate data needed by the application, reducing network traffic and storage
- Cascading configurations provide support for special needs
- Fast log based replication provides near instant replication
- A replication scheduler allows off-peak operations without impacting normal database use
- A replication history viewer provides easy confirmation and troubleshooting
- Automatic schema replication avoids maintenance headaches
- Table DDL replication keeps the system online during updates
- Graphical console and Command Line Interface (CLI)

EDB Postgres Replication Server also supports multi-master configurations with the following benefits:

- Improves high availability for write operations
- Deploys masters across geographies for consolidating data across locations
- Continuously synchronizes data, eliminating overnight batch jobs and ETL programming
- Automatic Conflict Detection prevents data corruption
- Multiple conflict resolution strategies address common multi-master problems
- Custom resolution strategies address unusual applications

Unlike the high availability and backup tools, an evaluation of replication tools can be quite extensive as they can be used to solve many different problems in a data architecture.

EDB Postgres Migration Tool Kit

The EDB Postgres Migration Tool Kit migrates Oracle, Microsoft SQL Server and MySQL data to PostgreSQL providing online/offline schema/data migration with flexible customization and fast parallel data loading.

Key features include:

- Automated schema/data migration - save time and effort
 - Online migration - migrate source database directly into PostgreSQL or Advanced Server, making it available for immediate use
- Offline migration - create DDL scripts that can be edited and run at a later scheduled time
- Multiple database support - migrate from Microsoft SQL Server, Sybase, or Oracle (including PL/SQL code) to PostgreSQL or EPAS
- Customize migrations - specify whether or not to include table constraints/triggers/indexes, manage bulk inserts, apply row filters, change data types inline, and specify subsets of schema objects to migrate or all
- Fast data migration - employ parallel data movement techniques, bypass logging for faster data loads or use native connectivity to source database

Many tools are also available in the PostgreSQL ecosystem, but none is capable of also migrating the database stored procedures and functions (PL/SQL) of the most widely used RDBMS, Oracle. Depending on the complexity of the application, this can save man-months of work in code re-writing and testing.

Performance Features

EDB has frequently led most of the performance efforts for the PostgreSQL community. In some cases, EDB has found that implementing performance features in PostgreSQL is more effective even for EPAS because it inherits all of PostgreSQL's features.

This was true for materialized views, index-only scans, linear read scaling up to 64 cores, and more recently, the addition of parallel query processing.

While self-supporting PostgreSQL users can enjoy some of EPAS's general performance features noted above, only subscribers to the EDB Postgres Platform will receive EDB's expertise in supporting, troubleshooting, and tuning these features in production deployments. Finally, the features listed below in the EDB Postgres Platform are not available to PostgreSQL adopted in an unsupported fashion.

Hash Indexed Partitions

For extremely large tables, the standard Range and List indexing of table partitions meets many user needs. However, performance can suffer when data access patterns for these tables involve frequent searching for exact matches or many concurrent inserts and/or updates. To solve this performance issue, EPAS allows indexing partitions with a hash index.

Query Hints

Optimizer hints are directives that can be embedded in certain SQL statements. These directives force the query optimizer to use (or not to use) a given type of query plan to retrieve data more efficiently (should one discover the optimizer is selecting a sub-optimal plan). These are extremely handy when developers know more about the nature of the data and record structures than the formal assumptions that are part of the optimizer. EPAS has a

wide variety of hints that can be applied to access methods, join orders, join plans, appends, and parallelism.

Bulk Data Loading

EDB*Loader is a high-performance tool for loading data in bulk and also supports parallel processing of tables.

Performance Analysis

EDB Postgres Advanced Server has the Dynamic Runtime Instrumentation Tools Architecture which monitors and records low level SQL session and system wait events. Diagnostic data is inserted to catalog views that can be queried and provide detailed statistics to help diagnose and fix performance bottlenecks.

Other important performance related features that are part of Postgres Enterprise Manager, not available to self-supporting PostgreSQL users using pgAdmin, include:

- SQL profiler analyzes large SQL workloads and helps identify poorly performing SQL
- Index adviser can also be employed to examine table structures and definitions, make recommendations for new indexes, and even deploy those indexes with the push of a button
- Postgres Expert can analyze the configuration files for databases en masse and make best practice recommendations for key performance and maintenance settings

3. ORACLE-COMPATIBILITY FEATURES

EPAS provides a number of important Oracle-like features. In fact, many EDB customers who don't use Oracle database products appreciate the additional features in EPAS. The Oracle compatibility helps them meet mission critical requirements.

Oracle Compatibility for Oracle Users

For organizations using Oracle database software or trying to get away from it, compatibility in the EDB Postgres Platform has much to offer:

- EPAS's database compatibility provides similar Oracle-like features so developers and operations staff can continue to leverage many of their existing Oracle skills
- EPAS understands and executes Oracle's PL/SQL commands natively without killing performance and without difficult-to-debug emulation, translation, or layers on other languages
- Much less re-writing of core business logic
- Support for code written for OCI and Proc*C
- Integrates into Oracle environments with heterogeneous replication or direct database links
- DBAs can leverage their experience with Oracle on EPAS - using Oracle compatible tools such as Password Profiles, Catalog Views, partitioning, SQL*Plus, SQL*Wrap, SQL*Loader, PL/SQL, Virtual Private Database and more

Oracle Compatibility for Non-Oracle Users

EDB's compatibility for Oracle has been in development since 2005. As mentioned above, many organizations use the EDB Postgres Platform simply because it has features (most added for Oracle compatibility) that are necessary for enterprise applications. These include:

- Database integrated password management for additional security
- Function Packages for developer productivity
- A resource manager to control and allocate CPU and I/O resources in the database
- Enhanced auditing up the application stack to track actual end user activity
- Procedural language obfuscation to protect sensitive code or algorithms in the database

There are many more features available, described in four indexed compatibility guides of over 1,100 pages, found at: <https://www.enterprisedb.com/resources/product-documentation>

Beware of "Me Too" Migrations

Many PostgreSQL consultancies and major public cloud providers advertise Oracle migrations to PostgreSQL using tools similar to what EDB provides. However, most noticeably absent from those solutions is the migration of PL/SQL, the Oracle programming language that encodes critical business logic and functions inside the database.

These “me too” migrations are for all intents and purposes complete rewrites of business logic—logic that may have thousands of hours of coding and testing behind them and years of flawless operation. EPAS Oracle compatibility natively understands and executes the original PL/SQL without emulation or translation layers and requires little rewriting of code. This dramatically reduces the risks of a migration and preserves staff knowledge and use of years of Oracle skills.

4. MORE DEPLOYMENT OPTIONS

Multiple Environments

Since PostgreSQL is in widespread use across the world in organizations of all types and sizes, it has been operated in today’s most popular environments. Both PostgreSQL and EPAS can be deployed in a variety of environments/deployment models including:

- Bare metal
- Virtual machines
- Containers
- Private cloud
- Public cloud
- Platform as a Service (PaaS)
- DBaaS

The advantage of the EDB Postgres Platform is twofold:

- Users have one organization to deal with for software acquisition, support maintenance, and training for all these environments and
- The EDB Postgres Platform includes DBaaS infrastructure software that can be used to deploy into private OpenStack clouds, or public Amazon and Microsoft clouds.

Many organizations will find themselves running two or three of these environments simultaneously even as they move off one to a new one. Running multiple environments presents unique challenges to provisioning, licensing, costs, management, monitoring, performance, security, high availability, replication, maintenance, scaling, data governance, and backup.

Architecting and re-architecting a database platform for best practices across all these challenges can take a lot of time and trial and error. Relying on a single platform from a single vendor is ultimately more efficient than dealing with multiple different organizations when using a self-supporting PostgreSQL strategy. EDB’s large customer base also provides a broad perspective of the circumstances most organizations will likely face and the best practice competencies needed.

DBaaS

EDB introduced the first database-as-a-service for PostgreSQL back in 2011 on Amazon EC2 called Postgres Plus Cloud Database. It introduced instant provisioning of single development instances or sophisticated production clusters with automatic connection pooling, load balancing, replication and read scale out. Automatic backup and cloning of entire databases in minutes was standard, as was self-healing clusters and automatic failover for high availability.

The self-service console allowed developers to create new databases and operations staff to create robust scalable production clusters of Postgres in minutes. This opened many new possibilities for cost savings, faster time to market, and more reliable systems.

Today, EDB Postgres Ark succeeds Postgres Plus Cloud Database and supports customized self-service database provisioning to both public and private clouds. A self-supporting PostgreSQL operation will be hard pressed to find or create such a DevOps oriented environment that allows choice amongst cloud vendors.

Emerging Technologies

EDB continues to work with customers and apply this knowledge of PostgreSQL to new environments and evolving requirements like containers, hybrid cloud, and Platform as a Service. These new capabilities have been

added to fully integrate and automate databases in custom DevOps environments based on Kubernetes and scripting languages like Chef and Puppet.

Whereas EDB's original DBaaS focused on instant satisfaction for developers and relied on operations staff to spin up new databases and production clusters, EDB Postgres Ark adds critical visibility and control features. These allow operations staff to customize and maintain software while complying with data governance responsibilities. EDB Postgres Ark features include:

- Multi-cloud support (Amazon, Microsoft and OpenStack)
- ReSTful API for PaaS Integration
- Customizable virtual machines
- Custom database templates
- Encryption

Organizations adopting a self-supporting PostgreSQL model simply won't get the specialized deployment tools available to an EDB Postgres Platform user.

5. ENHANCED INTEGRATION

The EDB Postgres Platform can integrate with other relational and non-relational databases in a variety of ways. Some are also possible with self-supported PostgreSQL, but would not come with the support and expertise provided by EDB.

Heterogeneous Database Replication

The EDB Postgres Platform provides a replication server that meets a wide variety of requirements. One unique characteristic is its ability to replicate between Postgres databases and Oracle or Microsoft SQL Server databases. For a single-master replication system, a variety of configurations are supported including replication:

- Between PostgreSQL and EDB Postgres Advanced Server databases (EPAS) (in either direction)
- From Oracle to PostgreSQL
- In either direction between Oracle and EPAS
- In either direction between Microsoft SQL Server and PostgreSQL
- In either direction between Microsoft SQL Server and EPAS

For multi-master replication, the participating database servers must be of the same type:

- PostgreSQL database servers
- PostgreSQL database servers and EDB Postgres Advanced Server (EPAS) operating in PostgreSQL compatible mode
- EPAS operating in PostgreSQL compatible mode
- EPAS operating in Oracle compatible mode

EDB Postgres Replication Server also allows table and record filtering, so there is no need to replicate an entire database from one location to the next—only the data that is needed. Performance is further enhanced by using the binary replication technology available in PostgreSQL.

Self-supported PostgreSQL users can find single and multi-master solutions in the PostgreSQL ecosystem, but none of them are ready to run and configure right out of the box like EDB Postgres Replication Server. They often require significant programming and setup specific to each installation. In addition, community replication solutions do not provide an easy-to-use graphical user interface management or a command line interface for scripting.

Oracle Database Links

The EDB Postgres Platform provides for a tighter coupling with Oracle databases through the use of database links. A database link is an object that allows a reference to a table or view in a remote Oracle database based on OCI within a DELETE, INSERT, SELECT or UPDATE command. This allows direct operations between EDB Postgres and Oracle databases without using additional software and hardware required by a solution like a standalone replication server.

Data Adapters

In addition to Oracle databases, there has been a small explosion of specialty and niche databases that now play crucial roles inside most large organizations. These include document and key-value based databases and more generalized large scale data stores like Hadoop. Consequently, a workhorse relational database system must integrate with and/or consolidate data from these sources.

Integration with non-relational data formats has been a priority for the EDB Postgres Platform. EDB has taken a leading role in the PostgreSQL community in the development of foreign data

wrappers (FDWs) based on the SQL/MED standard (SQL Management of External Data). EDB is the primary developer behind the community FDWs for MySQL, MongoDB and Hadoop, which serve as the foundation for the EDB Postgres Platform data adapters.

While EDB contributes the FDWs to the community via GitHub, the EDB Postgres Platform remains premier supplier of support and solution assistance for integrating the MongoDB JSON data, Hadoop analytics data and MySQL website data integrations.

6. ENHANCED PRODUCTIVITY

Developer Productivity

The EDB Postgres Platform provides features that improve productivity specifically for application developers that are not available in self-supported PostgreSQL environments.

Function Packages

One of the most important developer features in EPAS are its 21 predefined reusable Function Packages. These user expandable standard libraries replace a lot of extra coding with best practice code that speeds up the development process and reduces testing and debugging with over 200 functions for common programming tasks like:

- Database event notifications
- Data encryption
- Job scheduling
- Large object and raw data type manipulation
- Locking services
- Debugging output
- Inter-session communication
- SQL tuning
- Dynamic SQL
- Randomizing data
- Row level security and role setting
- Database metadata access
- Base64 encoding
- Reading/writing OS files
- HTTP calls
- Mail operations
- URL handling

Hierarchical Query

A timesaving and error reducing construct is the hierarchical query. This feature returns a result row composed of data from one or more rows in the same table based on a hierarchical relationship between the rows, such as manager to employee. This relieves the developer from much more complex coding.

Bulk Collect and Binds

Bulk Collect and Binds provide extremely fast processing for queries that perform a lot of switching between the SQL and Stored Procedure engines.

Bulk binding is used when a SQL statement is required to change a set of rows with a different set of values for each row. This technique is applied by the FORALL statement. The FORALL statement passes sets of values to the database server in collections. This eliminates the need to reiteratively invoke the same SQL statement with the different values.

Bulk collecting is used when a SQL statement returns multiple rows such as in a SELECT statement, FETCH statement, or when the RETURNING clause is used with insertions, updates, or deletions. The BULK COLLECT clause employs bulk collecting to return the result set in collections.

Specialized Data Types

EPAS also provides developers with a variety of enhanced data types necessary for many of today's modern programming tasks by offering new ways of collecting data and methods of organizing it. These include:

- **User defined object types** - allow the database to store real world object oriented representations of data, making development easier and quicker
- **Associative arrays** - a flexible data type that allows a column 'value' to hold one or more key-value pairs in an array
- **VARRAYS** - a flexible array data type for a column that stores a set number of elements that cannot be deleted and that have no gaps in the assignment of keys to values

- [Public/private synonyms](#) - a timesaving shorthand name that can be used to reference another fully qualified database object in a SQL statement
- [User defined exceptions](#) - gives developers more control over error reporting and error handling when processing sophisticated data

DBA Productivity

Whereas the previous features help those people tasked with creating applications, EPAS also provides special features to make people tasked with managing the day to day operations of databases more productive. These too are not available in a self-supported PostgreSQL environment.

Resource Manager

The EPAS Resource Manager allows a DBA to throttle CPU and I/O, assigning limits to different process groups. This protects high priority database processes like website purchase transactions from resource hungry low priority jobs that may be doing data loads, reporting or analysis. The ability to protect performance for certain production processes is extremely important for operational efficiency.

Enhanced Catalog Views

Another DBA feature in EPAS is a set of over 55 additional catalog views that profile all the objects and processing that occurs in the database. These additional views to those provided in PostgreSQL help DBA's understand their databases better, produce more detailed data processing reports, and save time in troubleshooting, security, and maintenance.

Dynamic Runtime Instrumentation Tools Architectures (DRITA)

Related to the extra catalog views are dynamic runtime instrumentation statistics gathered by EPAS. This feature allows a DBA to easily query catalog views populated with user selected statistics and determine the wait events that affect performance. EPAS instrumentation will record the number of times each event occurs as well as the time spent

waiting for it to complete on targeted processes. This type of fine grained analysis is critical to the fast diagnosis and troubleshooting of difficult performance problems.

Bulk Data Loading

EDB*Loader is often used for extremely large jobs. A feature that DBAs who use the EDB Postgres Platform really like is EDB*Loader's ability to record and save input data that have errors and continue processing the remaining data to completion. This allows DBAs to fix the errors one time all together and reload just those inputs. If limited to using the PostgreSQL COPY command, then DBAs must contend with the command aborting on error and possibly executing multiple cycles of fix and re-load.

7. POSTGRESQL COMMUNITY LEADERSHIP

PostgreSQL community participation with a commercial partner gives users deeper insights into creating solutions and allows them to actively influence the feature roadmap to address their specific requirements. Users of PostgreSQL will always be best served with a partner like EDB, as evidenced by the depth and breadth of EDB's community involvement described below.

In addition to helping the community thrive and grow, EDB's strong technical participation provides subscribers to the EDB Postgres Platform with more insight into and influence over the community feature roadmap. Self-supporting PostgreSQL users can always suggest database features to the community, but their voice is often difficult to hear over community members who represent thousands of users, have extensive experience in the community process, and possess the manpower to perform the work. This insight and influence extends to the tools included in the EDB Postgres Platform.

EDB Customer Driven Feature Contributions to PostgreSQL

EDB has developed major PostgreSQL product features including:

- Parallel query (PostgreSQL 9.6)
- Allowing old MVCC snapshots to be invalidated after a configurable timeout (PostgreSQL 9.6)
- synchronous_commit=remote_apply, allowing reliable read-scaling clusters (PostgreSQL 9.6)
- Lock improvements to increase scalability on multi-CPU machines (PostgreSQL 9.5, 9.4, 9.2)
- ALTER SYSTEM (PostgreSQL 9.4)
- Background worker processes that are dynamically started and terminated (PostgreSQL 9.4)
- Materialized views (PostgreSQL 9.3, 9.4)
- Greatly reduced System V shared memory requirements (PostgreSQL 9.3)
- Index-only scans (PostgreSQL 9.2)
- Original version of the JSON data type (PostgreSQL 9.2)
- Unlogged tables (PostgreSQL 9.1)
- Automatic sizing of the free space map (PostgreSQL 8.4)
- Visibility map (PostgreSQL 8.4)
- Heap Only Tuples (HOT) to accelerate space reuse and reduce MVCC overhead (PostgreSQL 8.3)

EDB Tool Contributions to the PostgreSQL Ecosystem

EDB contributes code and tools to selected PostgreSQL community ecosystem projects, such as:

- [hdfs_fdw](#) - a Postgres Foreign Data Wrapper for the Hadoop Distributed File System
- [mongo_fdw](#) - a Postgres Foreign Data Wrapper for BSON data
- [mysql_fdw](#) - a Postgres Foreign Data Wrapper for MySQL
- [pg_catcheck](#) - a simple tool for diagnosing system catalog corruption
- [pg_hibernate](#) - save and restore the Postgres shared-buffers contents across Postgres server restarts eliminating slow performance due to cache warmup cycles
- [pg_nosql_benchmark](#) - a tool for benchmarking Postgres (JSONB) and MongoDB (BSON)
- [pgAdmin](#) - the leading PostgreSQL development environment
- [pgPool](#) - a connection pooling and query routing tool
- [Postgres-XC](#) - a horizontal scaling fork of Postgres, whose major innovations are now being merged back into core Postgres

EDB PostgreSQL Community Activities

EDB creates and maintains installers to speed and ease the deployment of PostgreSQL for everyone:

- Creating and hosting freely available GUI binary installers for PostgreSQL
- Maintain freely available RPM installers for PostgreSQL

EDB supports the PostgreSQL community's ongoing needs for running an open source project:

- Hosting portions of the community's online infrastructure
- Testing of new features during the development process
- Working with hardware vendors to ensure PostgreSQL performs and scales well on their platforms
- Encouraging the growth of the PostgreSQL ecosystem with partnerships

8. FOLLOW THE SUN SUPPORT - 27X7

EDB Postgres Platform Support

The EDB Postgres Platform provides all the traditional software vendor services for PostgreSQL or EPAS, but with a pricing model that dramatically lowers the cost of these traditional services to the end user. EDB's services include:

- Live human-to-human support with Postgres experts backstopped by key community staff
- Support through all phases of the application lifecycle including evaluation/suitability, application development and testing, production deployment, and production operations
- 24x7 Service Level Agreements
- Software distribution services including the building and testing of GUI and RPM installers
- Software maintenance and testing including patches and emergency field fixes
- Technical and security alerts
- Professional Support Portal with a Knowledge Base
- Professional bug tracking and resolution tools
- Professional high quality documentation
- Technical videos
- Extended Lifetime Support options

Self-Supported PostgreSQL Support

In summary, what are the options for these types of services for self-supporting PostgreSQL users who opt not to use any kind of commercial partner?

In short, staff can monitor and ask questions of the PostgreSQL community mailing lists but without a guarantee of an answer, SLAs, or assurances of the level of competency of the responder. Staff will need to watch for updates, patches, and security issues, and probably spend a lot of time surfing the web and reading various blogs in search of answers that are close to meeting their issues. It is a plan that does not instill a lot of confidence in most IT or line of business managers, but certainly makes for interesting work for teams.

CONCLUSION

While this document highlights the major differences between self-supported PostgreSQL and the EnterpriseDB Postgres Platform, there may be topics not covered. EnterpriseDB staff is available to provide more detail or answer questions.

ABOUT EDB

EnterpriseDB (EDB) provides enterprises with the products, resources, and expertise required to confidently run large scale and highly available deployments of Postgres across on-premises, hybrid, multi-cloud, and DBaaS environments with reliable enterprise-class vendor support, services, and expertise.

The EDB Postgres™ Platform provides compatibility with native PL/SQL, the programming language that runs on Oracle database products, and includes automated migration tools to help save time and minimize uncertainty in the migration process.

EDB is an innovator and major contributor to the Postgres community, serving 20% of the Fortune 500 and 15% of Global 2000 companies worldwide.

EDB is based in the Bedford, MA with offices around the globe.

EnterpriseDB Corporation
34 Crosby Drive
Suite 201
Bedford, MA 01730

© Copyright EnterpriseDB Corporation 2018

EDB and EDB Postgres are trademarks of EnterpriseDB. All other marks, symbols, or words are owned by their respective owners.

A current list of EnterpriseDB trademarks is available on the web at "Copyright and trademark information" at www.enterprisedb.com/trademarks.

This document is current as of the initial date of publication and may be changed by EnterpriseDB at any time. THE INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT. EnterpriseDB products are warranted according to the terms and conditions of the agreements under which they are provided.

Call our nearest
location or email
sales@enterprisedb.com

US +1 781-357-3390 or 1-877-377-4352	UK +44-20 3740 6778	SWEDEN +46 844 683 476	JAPAN +81-50-5532-7038
GERMANY +49 322 210 97906	FRANCE +33 975 187 082	POLAND +48 223 079 848	KOREA +82-2-6007-2500
	NETHERLANDS +31 (0)20 808 0937	AUSTRALIA +61-2-8019-7055	INDIA +91-20-66449600/601

