EDB Postgres Advanced Server vs. Oracle® Enterprise

A Technical Comparison of
EDB Postgres Advanced Server
and Oracle® Enterprise

COMPARATIVE ANALYSIS
DECEMBER 2020

EDB

EDBPOSTGRES.COM

Contents

Introduction

Organizations are increasingly choosing EDB Postgres Advanced Server as a standard Relational Database Management System for new and existing applications.

EDB Postgres Advanced Server provides the performance, security, and manageability features and capabilities required to power the majority of enterprise workloads. An open source based development model reduces costs and liberates money spent on expensive proprietary databases to be used in developing new applications of innovation.

This potential to free up dollars in core IT is especially true for organizations using Oracle® because EDB Postgres Advanced Server is also compatible with Oracle.

EDB Postgres' compatibility allows it to:

- Be used as a substitute for Oracle for new applications
- Migrate many existing Oracle apps preserving investments in PL/SQL business logic
- Complement and coexist with their existing Oracle infrastructure
- Leverage existing staff skills using a new database.

Finally, the cost savings using EDB Postgres can become extremely large and impossible to ignore when deploying to virtual environments or the cloud when one compares EDB's and Oracle's pricing models for those environments.

This guide is intended to help you evaluate EDB Postgres' capabilities
and identify the workloads and applications where
EDB Postgres Advance Server can be used in place of Oracle.

In the pages that follow, you will find:

- A comparison of various aspects of Oracle Enterprise Edition from Oracle with EDB Postgres Advanced Server
- An emphasis on the issues of greatest interest to EDB prospects and customers as communicated to us since EDB's founding in 2004.
- A limited compilation of some options and tools used in the database or with the database software in common deployments.

Also note that the following information is not intended to be:

- A competitive comparison of all of Oracle's or all of EDB Postgres' capabilities and business practices.
- A comparison of capabilities specific to any one version of Oracle. It's also important to note that EDB's database compatibility features are driven specifically by customer requests, which span many versions of Oracle.
- Product documentation. This information does not reflect Oracle or EDB's product documentation. It also does not include all of EDB Postgres Advanced Server's compatibility features—only the most popular ones. For a comprehensive list of features and official documentation, refer to the information links below.
- A Total Cost of Ownership calculator. For actual pricing determinations and comparisons, readers are advised to contact EDB.

Compatibility with Oracle

Database administrators and application developers commonly ask, “Which version of Oracle are you compatible with?” EDB has developed database compatibility for Oracle based on popular features across many versions of Oracle. EDB’s goal has always been to create a critical mass of compatibility for the most popular features regardless of Oracle version to enable EDB Postgres Advanced Server to support Oracle workloads and provide end users significant cost savings for a large portion, or in some cases all, of their Oracle footprint. In selecting new features for every software release, EDB focuses on the most popular features whose value to customers meets one or more of the following criteria:

- **Reduced Technical Risk:** This refers to objects or code created in Oracle that can be migrated and executed “as is” against or inside an EDB Postgres Advanced Server database and behave or produce the identical result as they would in Oracle.

- **Reduced Re-training Risk:** This means that knowledge, skills, and tools most frequently used with Oracle can also be used with EDB Postgres Advanced Server significantly reduce the learning curve needed to be productive quickly in either creating new applications or migrating old ones.

- **Reduced Integration Risk:** This means that EDB Postgres Advanced Server databases and applications can integrate well with existing Oracle infrastructure and non-database software that will be retained or cannot be changed for the foreseeable future.

Tables Legend	
Yes/No	Denotes whether the feature or characteristic is supported in the database.
✓	The feature operates in a manner compatible with Oracle allowing users to continue using and/or migrate their existing Oracle skills, program code or data.
EDB Postgres Advanced Server	EDB’s database with compatibilty for Oracle and additional enterprise features for security and performance is built upon PostgreSQL and continuously merges changes with every major, minor and security release.
Yes ✓	Denotes whether the feature or characteristic is supported in the database, and that the feature operates in a manner compatible with Oracle allowing users to continue using and/or migrate their existing Oracle skills, program code or data.

General Capabilities

There are a few foundational details prospective users should understand straight away when comparing Oracle's database with the EDB Postgres Advanced Server (EPAS) database.

Both are mature, enterprise-class object-relational databases that meet the industry standards for atomicity, consistency, isolation, and durability (ACID) compliance. It's also important to note the products were both developed from the same seminal IBM research on System R, and designed to solve many of the same problems and so there is a great deal of similarity between the database programs.

General/Capabilities	Oracle Enterprise	EDB Postgres Advanced Server
Design Origin	Commerical implementation based on IBM's original research for System R.	Academic implementation (UC Berkeley) based on IBM's original research for System R.
Continuous Development Since	1979	PostgreSQL development started in 1986. EPAS development started in 2004. EPAS is based on PostgreSQL and continuously merged.
Object Relational Database	Yes	Yes
Processing Architecture	Process Based and Thread Based	Process Based
Full ACID Compliance	Yes	Yes
Multi-Version Concurrency Control	Yes	Yes
Multi-tenant Architecture	Yes	Yes
Automatic Workload Management	Yes	No
Enterprise Database Management	Oracle Enterprise Manager	EDB Postgres Advanced Server
Multi-Core Support	Yes	Yes
Write Ahead Durability	Redo Logs	Write Ahead Log
Disk Read Buffering	Yes	Yes

Terminology

For all the work that has gone into making SQL a standard, there are still differences in nomenclature used in many SQL based products. Some of the more important and perhaps non-obvious differences are noted below.

Terminology	Oracle Enterprise	EDB Postgres Advanced Server
Table or Index	Table or Index	Table, Index or Relation
Row	Row	Row or Tuple
Column	Column	Column or Attribute
Data Block	Data Block	Page - When Block is on Disk
Page	Page	Buffer - When Block is in Memory

Each instance of EDB Postgres Advanced Server is referred to as a “cluster”. A cluster is a collection of databases that is managed by a single program instance, and is comprised of a data directory that contains all data and configuration files and can be referred to in two ways: by location of the data directory or by port number. A single server can have many program installations and you can create multiple clusters using the command: initdb.

Capacities

Some of the first questions raised when considering a new database involve capacity. DBAs and developers need to understand whether a new solution has the capacity to support existing application data designs, workloads and anticipated growth. Applying the capacity of a new solution to an organization's workloads and future applications means understanding how it supports data across multiple structures within the database.

Capacities	Oracle Enterprise	EDB Postgres Advanced Server
Max. Database Size	Unlimited	Unlimited
Max. Table Size	4 GB x Block Size	32 TB
Max. Row Size	4 TB	1.6 TB
Max. Field Size	For BLOB: (4 GB - 1) x DB _BLOCK_SIZE Initialization Parameter	1 GB
Max. Rows per Table	Unlimited	Unlimited
Max. Columns per Table	1000	250 - 1600 Depending on Column types
Max. Indexes per Table	Unlimited	Unlimited

Tables and Partitioning

The range of constructs within the database and how much flexibility DBAs have in organizing these structures can impact performance as well as maintenance and other operational requirements. The ability to partition a database improves performance, for example. Organizing data into distinct structures and distributing them across the infrastructure also improves manageability, availability, and load balancing. Materialized views allow DBAs to replace slow, resource-intensive runtime queries, complex joins, or lengthy scans of data with simple, faster reads from pre-joined pre-sorted and stored results.

Tables and Partitioning	Oracle Enterprise	EDB Postgres Advanced Server
Temporary Tables	Yes	Yes
Views	Yes	Yes
Materialized Views	Yes	Yes
Partitioning	Yes	Yes ✓
Partition by Range	Yes	Yes ✓
Partition by Hash	Yes	Yes ✓
Partition by List	Yes	Yes ✓
Sub-Partitioning	Yes	Yes ✓
Interval Partitioning	Yes	No
Partitioned Indexes	Yes	No
ANSI Constraints	Yes	Yes
Tablespaces	Yes	Yes
Index Organized Tables	Yes	Can cluster a table by an index providing similar performance boosts when reading data from a pre-sorted structure.

Data Types

Data types provide various ways for a DBMS to define, implement, and use information within the system and put constraints on how data is interpreted by the database when multiple data types are in use. EDB Postgres Advanced Server has very strong compatibility with Oracle data types and is highly extensible allowing it to very quickly support new and emerging data types and workloads as they become popular.

Data Types	Oracle Enterprise	EDB Postgres Advanced Server
Type System	Static + Dynamic (through ANYDATA)	Static
Integer	NUMBER	NUMBER ✓, DEC , NUMERIC, SMALLINT (16-bit), INT, BINARY_INTEGER, PLS_INTEGER, INTEGER (32-bit), BIGINT (64 bit)
Floating Point	BINARY_FLOAT, BINARY_DOUBLE	BINARY_FLOAT ✓, BINARY_DOUBLE ✓, FLOAT, REAL (32-bit), DOUBLE PRECISION (64-bit)
Decimal	NUMBER	NUMBER ✓, DEC, DECIMAL, NUMERIC
String	CHAR, VARCHAR2, CLOB, NCLOB, NVARCHAR2, NCHAR, LONG (deprecated)	CHAR ✓, VARCHAR ✓, CLOB ✓, NCLOB ✓, NVARCHAR2 ✓, NCHAR, CHARACTER, TEXT, CHAR, VARYING, CHARACTER VARYING, VARCHAR
Binary	BLOB, RAW, LONG RAW (deprecated), BFILE	BLOB ✓, RAW ✓, LONG RAW ✓, BYTEA (no compatible type for BFILE)
Date/Time	DATE, TIMESTAMP (with/without TIMEZONE), INTERVAL	DATE ✓ TIMESTAMP (with/without TIMEZONE), INTERVAL ✓, TIME (with/without TIMEZONE)
Boolean	Not Available	BOOLEAN
ROWID	ROWID	ROWID
XMLTYPE	XMLTYPE	XMLTYPE
Key-Value	Requires NSWLDB which is a separate database program.	Yes, is integrated into the core database
JSON	Use VARCHAR2, CLOB, and BLOB with is_json check constraint.	JSON and fast binary JSONB with 58 JSON operators, functions and relational json converters
Spatial / Geospatial	Yes	Yes
Other	IMAGE, AUDIO, VIDEO, DICOM	ENUM, POINT, LINE, LSEG, BOX, PATH, POLYGON, CIRCLE, CIDR, INET, MACADDR, BIT, UUID, XML, arrays, composites, ranges, custom
Data Domains	Yes	Yes

Indexes

In order to provide optimal performance for the wide range of supported data types and new workloads utilizing those data types, the database must also support a wide variety of indexes. EDB Postgres Advanced Server is somewhat unique in this regard, especially its GiST index which allows for easy development of specialized indexes for new data types.

Indexes	Oracle Enterprise	EDB Postgres Advanced Server
B-Tree	Yes	Yes
Hash	Yes	Yes
Expressions	Yes	Yes
Partial	Yes	Yes
Reverse	Yes	Yes A functional index can be used to reverse the order of a field
Bitmap	Yes	use GIN Index
Block Range Index	Yes	Yes
GiST Easy creation of specialized indexes.	No	Yes
GIN Custom inverted indexes.	No	No
K-Nearest-Neighbor	Yes Using the package DMBS_DATA_ MINING and Spatial option.	No
Full Text Search	Yes	Yes
Spatial	Yes	Yes - using free PostGIS extension

SQL Capabilties

Postgres Advanced Server strongly conforms to the ANSI-SQL:2008 standard. It also has Transactional DDL which supports backing out even large changes to DDL, such as table creation. While you can't recover from an add/drop on a database or tablespace, all other catalog operations are reversible. This feature is often used for protection when doing complicated work like schema upgrades. If you put all such changes into a transaction block, you can make sure they all apply atomically or not at all. This drastically lowers the possibility that the database will be corrupted by a typo or other such error in the schema change, which is particularly important when you're modifying multiple related tables where a mistake might destroy the relational key.

SQL Capabilites	Oracle Enterprise	EDB Postgres Advanced Server
Union	Yes	Yes ✓
Intersect	Yes	Yes ✓
Except	Yes	Yes ✓
Inner Joins	Yes	Yes ✓
Outer Joins	Yes	Yes ✓
Inner Selects	Yes	Yes ✓
Merge Joins	Yes	Yes ✓
Common Table Expressions	No	Yes
Windowing Functions	No	Yes
Parallel Query	Yes	Yes
Query Hints	Yes	Yes ✓
Transactional DDL	Yes	Yes
Alter Session	Yes	Yes
Dynamic SQL	Yes	Yes

SQL Extensions

Oracle has a number of SQL extensions that are very popular with Oracle users. While not standard to the SQL language they provide a lot of utility and convenience to DBAs and developers. EDB Postgres Advanced Server supports those most desired by EDB customers.

SQL Extensions	Oracle Enterprise	EDB Postgres Advanced Server
DUAL	Yes	Yes ✓
DECODE	Yes	Yes ✓
ROWNUM	Yes	Yes ✓
SYSDATE	Yes	Yes ✓
SYSTIMESTAMP	Yes	Yes ✓
NVL, NVL2	Yes	Yes ✓

High Availability

Mission-critical workloads must remain operational at all times and have little tolerance for even planned downtime for maintenance. This demand for high availability requires solutions that provide high speed replication and redundancy to eliminate single points of failure in the system; failure detection and automated failover to ensure systems continue to function in the event of a breakdown in the system; and data and system recovery to assist DBAs following failure events.

High Availability	Oracle Enterprise	EDB Postgres Advanced Server
Data Guard	Yes	Yes Using Streaming Replication and Log Shipping
Active Data Guard	Yes	Yes Using Streaming Replication, Log Shipping, Cluster Health Monitoring, Failover and Replica Reads.
Flashback Query	Yes	No
Flashback Table, Database and Transaction Query	Yes	No
Backup and Recovery Tools	Yes	Yes
Point in Time Recovery	Yes	Yes

Performance and Scalability

Database operations may be optimized through various means to deliver higher performance. Connection pooling, for example, refers to a common way of maintaining open connections to the database for applications that repeatedly make requests as opposed to having to create new connections each time. Data replication can increase performance by making information simultaneously available to multiple end-user applications. These performance enhancements can be achieved through database enhancements and various external complementary solutions.

Performance/Scalability	Oracle Enterprise	EDB Postgres Advanced Server
Connection Pooling	Yes	Yes
Real Application Clusters (RAC)	Yes A shared everything architecture for Performance, High Availability and Read scaling.	No A shared nothing architecture. High Availability is achieved with EDB Failover Manager or active/passive clustering. Read scaling is achieved with Replication
In-Memory Database	Yes	No
Multi-Master Read/Write Solution	Advanced Replication, Streams and GoldenGate	EDB Replication Server and native Postgres Bi-Directional Replication
Columnar Store	Yes Using In-Memory Column Store	Yes Using cstore Foreign Data Wrapper
CPU and I/O Resource Limits	Yes	Yes

Security

Database security encompasses many dimensions from secure connections to password management to access control to physical data encryption to auditing and more. Among open source and commercial databases, EDB Postgres Advanced Server is among the most secure and contains extensive support for PCI DSS.

Security	Oracle Enterprise	EDB Postgres Advanced Server
Authentication Systems Support	Yes LDAP, SSL, RADIUS, PAM, Kerberos, GSSAPI, SSPI	Yes LDAP, SSL, RADIUS, PAM, Kerberos, GSSAPI, SSPI
DB Connection Encryption	Yes	Yes
DB Connection White Lists	Yes – Using before connect triggers	Yes
DB Connection Black Lists	Yes – Using before connect triggers	Yes
Profiles for Passwords	Yes	Yes ✓
Server Code Obfuscation	Yes	Yes
ANSI Standard SQL GRANT/REVOKE	Yes	Yes
Column Level Permissions	Yes	Yes
User/Group/Role Support	Yes	Yes ✓
Virtual Private Database	Yes	Yes
View Security Barriers	No	Yes
Data Masking	Yes	No
Data Redaction	Yes	Yes ✓
Real Application Security	Yes	Only DBMS_RLS functionality
Database Vault	Yes	No
Audit Vault and Database Firewall	Yes	Database Firewall Only (SQL/Protect)
Advanced Security	Yes	Multiple options available (See Appendix A)
Fine Grained Auditing	Yes	Yes ✓– Using VPD policies to insert audit trail into an audit log upon access
Data Encryption Toolkit	Yes	Yes ✓

Integration

Today's data centers commonly consist of one or more relational and many non-relational database solutions deployed to handle specific workloads based on data type and application. Relational databases utilize a range of mechanisms for connecting to other like and dissimilar database solutions across the infrastructure in order to connect data from multiple sources and create a cohesive data fabric. In some cases, the database is engineered with specific capabilities that enhance data integration. Database vendors also develop adaptors that enable their database to connect with other vendor solutions.

Integration	Oracle Enterprise	EDB Postgres Advanced Server
Database Links	Yes	Yes ✓
Native Asynchronous Log-Based Replication	Yes	Yes
Native Synchronous Log-Based Replication	Yes	Yes
Session Based Synchronous Replication*	No	Yes
Distributed Transactions	Yes	No
Distributed Queries	Yes	Yes
Integration with: SQL Server, Sybase, Hadoop, MongoDB, MySQL	Database Gateway Database Gateway Oracle Data Integrator Golden Gate Oracle Data Integer	EDB Replication Server Not available Hadoop Data Adapter MongoDB Data Adapter MySQL Data Adapter

*It is possible, and often useful, to have some transactions commit synchronously and others asynchronously depending on the session connected to the database.

Application Development

Databases are a foundation of today's data-driven enterprise and applications are increasingly data intensive. Vendors in turn work to continually enhance their database solutions to support the needs of application developers, who seek the flexibility to make choices and simple ways for executing complex tasks. For example, databases that can provide support for multiple server-side languages for triggers, functions, and stored procedures give developers the option to choose their language for both client, middle tier and database server programming. Object oriented capabilities like user-defined object types allow the database to store real world representations of data thus making development easier, quicker, and more understandable.

Application Development	Oracle Enterprise	EDB Postgres Advanced Server
IDE	SQL Developer	EDB Postgres Advanced Server
Database Server Programming Language	PL/SQL (Block Structured Language)	SPL (PL/SQL Compatible) (Block Structured Language)
Additional Programming Languages for Database Server Stored Procedures, Triggers and Functions	Java	PL/pgSQL (PostgreSQL's Procedural Language) PL/Java C, C++ PL/Perl, Python PL/Tcl
Java Support	No	Yes
JDBC Support	Yes	Yes
ODBC Support	Yes	Yes
.NET Support	Yes	Yes
PL/SQL Debugger	SQL Developer	EDB Postgres Advanced Server
Stored Procedures	Yes	Yes ✓
Named Parameter Notation for Stored Procedures	Yes	Yes ✓

Application Development, cont.

Application Development	Oracle Enterprise	EDB Postgres Advanced Server
Triggers	Yes	Yes ✓
REF Cursors	Yes	Yes ✓
Implicit/Explicit Cursors	Yes	Yes ✓
Anonymous Blocks	Yes	Yes ✓
Bulk Collect/Bind	Yes	Yes ✓
Associative Arrays	Yes	Yes ✓
Nested Tables	Yes	Yes ✓
VARRAYS	Yes	Yes ✓
Hierarchical Queries	Yes	Yes
Parallel Query	Yes	Yes ✓
PL/SQL Supplied Packages	Yes	Yes (See Appendix B)
PRAGMA RESTRICT_REFERENCES	Yes	Yes ✓
PRAGMA EXCEPTION_INIT	Yes	Yes ✓
PRAGMA AUTONOMOUS_TRANSACTION	Yes	Yes ✓
User Defined Functions	Yes	Yes
User Defined Objects	Yes	Yes
User Defined Exceptions	Yes	Yes ✓

Big and Unstructured Data

Databases are a foundation of today's data-driven enterprise and applications are increasingly data intensive. Vendors in turn work to continually enhance their database solutions to support the needs of application developers, who seek the flexibility to make choices and simple ways for executing complex tasks. For example, databases that can provide support for multiple server-side languages for triggers, functions, and stored procedures give developers the option to choose their language for both client, middle tier and database server programming. Object oriented capabilities like user-defined object types allow the database to store real world representations of data thus making development easier, quicker, and more understandable.

Big/Unstructured Data	Oracle Enterprise	EDB Postgres Advanced Server
Spatial/Location/Graph	Yes	Yes
JSON Support	Yes Text Based	Yes Text- and High Performance Binary-Based
Key-Value Store	NoSQLDB	Yes
Support for XML Namespaces, DOM, XQuery, SQL/XML, and XSLT	XML DB	No
Compression (Tables, Files, Network, and Backups)	Yes	No
Partitioning	Yes	Yes
Hadoop Integration	Yes ETL via Data Integrator Application Adapter for Hadoop	Yes Real-time Join with Relational Data with Hadoop Data Adapter
MongoDB Integration	Yes Golden Gate Adapter	Yes Read/Write/Join with MongoDB Data Adapter
Cube, Rollup and Grouping Sets	Yes	Yes
Transportable Cross-Platform Table Spaces	Yes	No
Full Text Search	Yes	Yes
Advanced Compression	Yes	No

Management

Large enterprises have large database deployments often into the hundreds and beyond. DBAs require tools for maintaining these data farms easily and quickly and for performing operations in bulk across multiple databases. Customizable graphical consoles with a full compliment of features for monitoring, tuning, managing, and alerting are paramount to DBAs performing the basics of their responsibilities.

Management encompasses both the capabilities within the database that support the DBA in their operational tasks and tools external to the database as well.

Management	Oracle Enterprise	EDB Postgres Advanced Server
CLI	SQL*Plus	EDB*Plus ✓
Bulk Data Loader	SQL*Loader	EDB*Loader ✓
Enterprise Management	Oracle Enterprise Manager	EDB Postgres Advanced Server
System Catalog Views	Yes	Yes ✓ (See Appendic C)
Point-in-Time Recovery (PITR)	Yes	Yes
Online Backup	Yes	Yes
Online Reorganization	Yes	No
Automatic Memory Management	Yes	No
Automatic Storage Management	Yes	No
Automatic Undo Management	Yes	Yes
Diagnostics Package	Yes	Yes
Tuning Package	Yes	Tuning Wizard, Index Advisor, Postgres Expert in PEM
SQL Query Profiler	Yes	Yes in PEM

Incompatibilities

There are a number of incompatibilities between Oracle and EDB Postgres Advanced Server that are either not yet addressed or worth noting because of their frequent use.

Incompatibilities	Oracle Enterprise	EDB Postgres Advanced Server
MERGE	Yes	UPSERT
Advanced Queuing	Yes	Yes
Nested Procedures/Functions	Yes	Yes
Pipelined Functions	No	No
Empty String vs Null	Empty string = NULL	Empty string = !NULL
Casting	Performs many implicit data type conversions such as a number to a string	Requires you to cast a datatype to the other datatype or an error is produced

Deployment Options

With the advance of private, public, and private clouds and virtualization, the range of database deployment options has increased for end users. The following provides a snapshot of the available deployment options for Oracle Enterprise and EDB Postgres Enterprise.

Deployment Options	Oracle Enterprise	EDB Postgres Advanced Server
On-Premises Hardware	Intel AMD IBM Power Sun ultraSPARC	Intel AMD IBM Power
On-Premises Virtual	Yes With restrictions	Yes
On-Premises Cloud Private Cloud	Oracle Cloud on an Oracle stack including Infrastructure and Platform (middleware and database)	Consumption based solution in HPE GreenLake Database with EDB Postgres.
Public Cloud	Oracle Cloud on a proprietary stack including Infrastructure and Platform (middleware and database)	Cloud Database Service - EDB managed database service on AWS
Public Cloud - self managed	Yes	Yes
Hybrid Cloud	Yes	Yes
Containers	Yes	Yes

APPENDIX A:

Data Encryption Options

The following data encryption options offer different levels and granularity of protection depending on the needs of the application.

Using pgcrypto

- Postgres contrib module
- Applied to selected table columns
- Cannot search or index encrypted fields
- Encryption must be applied at table creation, so advance planning is required
- The application must handle the encryption/ decryption so that exchanges with the database remain encrypted
- DBAs cannot see data in clear

Using DBMS_CRYPTO

- Oracle compatible wrapper around pgcrypto with same features and limitations
- Supports multiple cipher algorithms
- DES, 3DES, AES and AES128
- MD4, MD5 and SHA-1 hash functions
- Generate cryptographically strong random values

Using Disk Partition Encryption

- File system disk partition is encrypted / decrypted by the OS
- Protects all files in the database partition including temporary files
- Data is decrypted when read from the filesystem. This allows DBAs to see the data – so have roles and permissions locked down
- Transparent to application developers e.g. Red Hat Enterprise Linux supports Linux Unified Key Setup-on-disk-format (LUKS)

Using File System Level

- Individual files or directories are encrypted by the file system
- Requires file-based key management
- Individual management of encrypted files e.g. incremental backups even in encrypted form
- Access control can be enforced by use of public-key cryptography
- Cryptographic keys are only held in memory while the file that is decrypted by them is held open
- Transparent to application developers e.g. eCryptfs for Linux (http://ecryptfs.org/)

APPENDIX B:

EDB Postgres Advanced Server Compatible Package Support

EDB focuses on the most popular functions within packages. Hence for some packages not all Oracle functions may be supported. For specific details refer to the EDB Postgres Advanced Server documentation.

Package Name	Package Description
DBMS_ALERT	Functions that allow asynchronous notification of database events via an Alert. Using this package and triggers, an application can notify itself whenever values of interest in the database are changed.
DBMS_AQ	Database-integrated asynchronous message queuing provides a flexible mechanism for integrating applications across the enterprise by communicating activities and exchanging a variety of information payloads.
DBMS_CRYPTO	Provides functions to encrypt and decrypt stored data.
DBMS_JOB	Has been replaced by DBMS_SCHEDULER but included for compatibility with older Oracle applications.
DBMS_LOB	Functions that allow access to and manipulation of Large Object values.
DBMS_LOCK	Provides a function interface to Lock Management services.
DBMS_OUTPUT	Allows the sending of messages from stored procedures, packages, and triggers for application or debugging use.
DBMS_PIPE	Functions that allow two or more sessions in the same database instance to communicate with one another.
DBMS_PROFILER	Provides functions to profile stored procedure workloads and identify performance bottlenecks.
DBMS_SCHEDULER	Job scheduler functions for creating and executing unattended repetitive tasks inside the database.

APPENDIX B: continued

Package Name	Package Description
DBMS_SQL	Permits the use of dynamic SQL in procedures to allow applications to run SQL statements with unknown parameters (such as table name) until runtime.
DBMS_RANDOM	Useful functions to generate random text, numeric and date values.
DBMS_REDACT	Redaction prevents a user from seeing all or portions of sensitive data.
DBMS_RLS	Implements row level security functions in the database blocking users from seeing each other's data in the same application.
DBMS_SESSION	Functions with the ability to enable and disable roles.
DBMS_UTILITY	A collection of functions for getting information about various runtime operations and meta data from the database.
UTL_ENCODE	Functions to perform Base64 encoding and decoding of data intended for transport between hosts.
UTL_FILE	Allows database procedures to read and write operating system text files in an I/O stream fashion.
UTL_HTTP	Functions that provide the ability to make HTTP calls to access information on web servers.
UTL_MAIL	Provides functions to create, manage, and send email from the database including attachments, CC, and BCC.
UTL_RAW	Functions supporting the manipulation of raw data types.
UTL_SMTP	Provides functions for sending mail via SMTP according to the RFC821 specification.
UTL_URL	Functions for escaping and "un-escaping" URL strings.

APPENDIX C:

EDB Postgres Advanced Server Compatible Catalog Views

EDB Postgres Advanced Server provides over 70 Oracle Catalog Views that provide information about database objects in a manner compatible with the Oracle data dictionary views.

ALL_ALL_TABLES
ALL_CONS_COLUMNS
ALL_CONSTRAINTS
ALL_DB_LINKS
ALL_IND_COLUMNS
ALL_INDEXES
ALL_JOBS
ALL_OBJECTS
ALL_PART_KEY_COLUMNS
ALL_PART_TABLES
ALL_POLICIES
ALL_SEQUENCES
ALL_SOURCE
ALL_SUBPART_KEY_COLUMNS
ALL_SYNONYMS
ALL_TAB_COLUMNS
ALL_TAB_PARTITIONS
ALL_TAB_SUBPARTITIONS
ALL_TABLES
ALL_TRIGGERS
ALL_TYPES
ALL_USERS
ALL_VIEW_COLUMNS
ALL_VIEWS
DBA_ALL_TABLES
DBA_CONS_COLUMNS
DBA_CONSTRAINTS
DBA_DB_LINKS
DBA_IND_COLUMNS
DBA_INDEXES
DBA_JOBS
DBA_OBJECTS
DBA_PART_KEY_COLUMNS
DBA_PART_TABLES
DBA_POLICIES
DBA_ROLE_PRIVS
DBA_ROLES
DBA_SEQUENCES
DBA_SOURCE
DBA_SUBPART_KEY_COLUMNS
DBA_SYNONYMS
DBA_TAB_COLUMNS
DBA_TAB_PARTITIONS
DBA_TAB_SUBPARTITIONS
DBA_TABLES
DBA_TRIGGERS
DBA_TYPES
DBA_USERS
DBA_VIEW_COLUMNS
DBA_VIEWS
USER_ALL_TABLES
USER_CONS_COLUMNS
USER_CONSTRAINTS
USER_DB_LINKS
USER_IND_COLUMNS
USER_INDEXES
USER_JOBS
USER_OBJECTS
USER_PART_KEY_COLUMNS
USER_PART_TABLES
USER_POLICIES
USER_ROLE_PRIVS
USER_SEQUENCES
USER_SOURCE
USER_SUBPART_KEY_COLUMNS
USER_SYNONYMS
USER_TAB_COLUMNS
USER_TAB_PARTITIONS
USER_TAB_SUBPARTITIONS
USER_TABLES
USER_TRIGGERS
USER_TYPES
USER_USERS0
USER_VIEW_COLUMNS
USER_VIEWS
V$VERSION
PRODUCT_COMPONENT_VERSION

About EDB

PostgreSQL is increasingly the database of choice for organizations looking to boost innovation and accelerate business. EDB's enterprise-class software extends PostgreSQL, helping our customers get the most out of it both on premises and in the cloud. And our 24x7 global support, professional services, and training help our customers control risk, manage costs, and scale efficiently.

With 16 offices worldwide, EDB serves over 4,000 customers, including leading financial services, government, media and communications, and information technology organizations.

To learn about PostgreSQL for people, teams, and enterprises, visit edbpostgres.com.

EDB Postgres Advanced Server vs. Oracle® Enterprise

A Technical Comparison of
EDB Postgres Advanced Server
and Oracle® Enterprise

ALL PRODUCT, PRODUCT SPECIFICATIONS AND DATA ARE SUBJECT TO CHANGE WITHOUT NOTICE TO IMPROVE RELIABILITY, FUNCTION OR DESIGN OR OTHERWISE.

EDB Corporation, its affiliates, agents, and employees, and all persons acting on its or their behalf (collectively, "EDB"), disclaim any and all liability for any errors, inaccuracies or incompleteness contained in any datasheet or in any other disclosure relating to any product.

EDB makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose or that it will continue to produce a particular product.

Statements regarding the suitability of products for certain types of applications are based on EDB's knowledge of typical requirements that are often placed on EDB products in generic applications. Such statements are not binding statements about the suitability of products for a particular application. It is the customer's responsibility to validate that a particular product with the properties described in the product specification is suitable for use in a particular application.