Most PostgreSQL tools are classified under the following categories...

* Backup And Recovery Tools
* Monitoring Tools
* Logical and Trigger Based Replication Tools
* Multimaster Replication Tools
* High Availability And Failover Tools
* Connection Pooling Tools
* Table Partitioning Tools
* Migration Tools

**Backup And Recovery Tools**

**Barman**

Barman (Backup and Recovery Manager) is an open-source administration tool for disaster recovery of PostgreSQL servers written in Python.It allows your organisation to perform remote backups of multiple servers in business critical environments and help DBAs during the recovery phase.Barman’s most wanted features include backup catalogs, incremental backup, retention policies, remote recovery, archiving and compression of WAL files and backups.Barman is written and maintained by PostgreSQL professionals at 2ndQuadrant.

**EDB BART**

EDB Backup and Recovery Tool (BART) is a key component of an enterprise-level Postgres-based data management strategy. BART implements retention policies and point-in-time recovery requirements for large-scale Postgres deployments. Bart 2.0 VERSION provides block-level incremental backups.

**PgBackRest**

pgBackRest aims to be a simple, reliable backup and restore system that can seamlessly scale up to the largest databases and workloads. Instead of relying on traditional backup tools like tar and rsync, pgBackRest implements all backup features internally and uses a custom protocol for communicating with remote systems. Removing reliance on tar and rsync allows for better solutions to database-specific backup challenges. The custom remote protocol allows for more flexibility and limits the types of connections that are required to perform a backup which increases security.

**Monitoring Tools**

**POWA**

PoWA is PostgreSQL Workload Analyzer that gathers performance stats and provides real-time charts and graphs to help monitor and tune your PostgreSQL servers.It is similar to Oracle AWR or SQL Server MDW.

**PgCluu**

pgCluu is a PostgreSQL performances monitoring and auditing tool.View reports of all statistics collected from your PostgreSQL databases cluster. pgCluu will show you the entire information of the PostgreSQL Cluster and the system utilization.

**Pgwatch2**

Pgwatch2 is one of the easiest monitoring tools to be used for PostgreSQL monitoring.It is based on Grafana and offers out of the box monitoring for PostgreSQL databases. Due to the use of containers, pgwatch 2 can be installed in minutes without having to worry about dependencies and complex installation procedures. It is pre-configured. All it needs is a database connection and you are ready to go.

**Logical and Trigger Based Replication Tools**

**pgLogical**

pglogical is a logical replication system implemented entirely as a PostgreSQL extension. Fully integrated, it requires no triggers or external programs. This alternative to physical replication is a highly efficient method of replicating data using a publish/subscribe model for selective replication.

**Slony-I**

Slony-I is a "master to multiple slaves" replication system for PostgreSQL supporting cascading replication. The big picture for the development of Slony-I is that it is a master-slave replication system that includes all features and capabilities needed to replicate large databases to a reasonably limited number of slave systems.

Slony-I is a system designed for use at data centers and backup sites, where the normal mode of operation is that all nodes are available.

**Bucardo**

Bucardo is an asynchronous PostgreSQL replication system, allowing for both multi-master and multi-slave operations. It was developed at Backcountry.com by Jon Jensen and Greg Sabino.

**Multimaster Replication Tools**

**BDR**

Bi-Directional Replication for PostgreSQL (Postgres-BDR, or BDR) is the first open source multi-master replication system for PostgreSQL to reach full production status. Developed by 2ndQuadrant, BDR is specifically designed for use in geographically distributed clusters, using highly efficient asynchronous logical replication, supporting anything from 2 to more than 48 nodes in a distributed database.

**High Availability And Failover Tools**

**Repmgr**

repmgr is an open-source tool suite for managing replication and failover in a cluster of PostgreSQL servers. It enhances PostgreSQL's built-in hot-standby capabilities with tools to set up standby servers, monitor replication, and perform administrative tasks such as failover or manual switchover operations.repmgr was designed by 2ndQuadrant.

**PAF**

PostgreSQL Automatic Failover is a new OCF resource Agent dedicated to PostgreSQL. Its original wish is to keep a clear limit between the Pacemaker administration and the PostgreSQL one, to keep things simple, documented and yet powerful. Once your PostgreSQL cluster is built using internal streaming replication, PAF is able to expose to Pacemaker what is the current status of the PostgreSQL instance on each node: master, slave, stopped, catching up, etc. Should a failure occur on the master, Pacemaker will try to recover it by default. Should the failure be non-recoverable, PAF allows the slaves to be able to elect the best of them (the closest one to the old master) and promote it as the new master.

**Patroni**

Patroni is a template for you to create your own customized, high-availability solution using Python and - for maximum accessibility - a distributed configuration store like ZooKeeper, etcd or Consul. Database engineers, DBAs, DevOps engineers, and SREs who are looking to quickly deploy HA PostgreSQL in the datacenter-or anywhere else-will hopefully find it useful.

**Stolon**

stolon is a cloud native PostgreSQL manager for PostgreSQL high availability. It's cloud native because it'll let you keep an high available PostgreSQL inside your containers (kubernetes integration) but also on every other kind of infrastructure (cloud IaaS, old style infrastructures etc.).

**Connection Pooling Tools**

**PgBouncer**

This is a very popular connection pool written by Skype developers in 2007. The project has been maintained by various developers in subsequent years, but its role of lowering the cost of connecting to PostgreSQL has never changed.PgBouncer allows PostgreSQL to interact with orders of magnitude more clients than is otherwise possible because its connection overhead is much lower. Instead of huge libraries, accounting for temporary tables, query results, and other expensive resources, it essentially just tracks each client connection in a queue. Then, based on configuration settings, it creates several PostgreSQL connections and assigns them to the connections on a first-come, first-served basis.

**PgPool-II**

The next pooling resource we will explore is named pgpool-II, but we'll refer to it simply as pgpool. This is another popular connection proxy, but it predates PgBouncer by almost a year, having been available since late 2006. The scope of pgpool is also much larger, providing functionality such as query-based replication, connection pooling, load balancing, parallel-query, and more. One feature pgpool exposes, is server pooling. If we have two PostgreSQL servers, we can make use of a virtual IPaddress so that clients need not modify configuration files when we switch the primary database server. However, in order to move the IP address between servers, it must first be removed from one server and recreated on the other. This disconnects all active clients and causes a small disruption in availability.However, pgpool can pool servers so that the active primary server is hidden from database clients. We can promote the secondary within pgpool, and it will handle failover internally. From the application or client's perspective, the database was never offline.

**PostgreSQL Table Partitioning Tools**

**Pg\_Partman**

pg\_partman is an extension to create and manage both time-based and serial-based table partition sets. Sub-partitoning is also supported. Child table & trigger function creation is all managed by the extension itself. Tables with existing data can also have their data partitioned in easily managed smaller batches. Optional retention policy can automatically drop partitions no longer needed. A background worker (BGW) process is included to automatically run partition maintenance without the need of an external scheduler (cron, etc) in most cases.

**pg\_Pathman**

The pg\_pathman is a Postgres Pro extension that provides an optimized partitioning solution for large and distributed databases. Using pg\_pathman, you can partition large databases without downtime, speed up query execution for partitioned tables, manage existing partitions and add new partitions on the fly, add foreign tables as partitions, and join partitioned tables for read and write operations.

**Migration Tools**

**Ora2pg**

Ora2Pg is a free tool used to migrate an Oracle or MySQL database to a PostgreSQL compatible schema. It connects your Oracle database, scans it automatically and extracts its structure or data, it then generates SQL scripts that you can load into your PostgreSQL database. Ora2Pg can be used from reverse engineering Oracle database to huge enterprise database migration or simply to replicate some Oracle data into a PostgreSQL database. It is really easy to use and doesn't need any Oracle database knowledge than providing the parameters needed to connect to the Oracle database.