

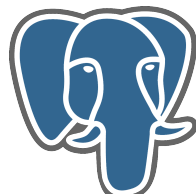


Postgres

Fabien Coelho, Claire Medrala

Mines Paris – PSL, MESR

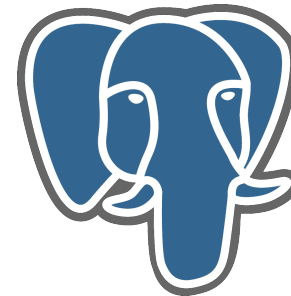
Décembre 2025



1 / 25



Postgres : base de donnée relationnelle-objet



Elephants can remember...

logiciel libre gratuit, source

architecture client-serveur

serveur portable **Unix** et Windows
Linux, *BSD, Solaris, AIX, MacOS

clients ■ texte, graphique, web...
■ Unix et Windows

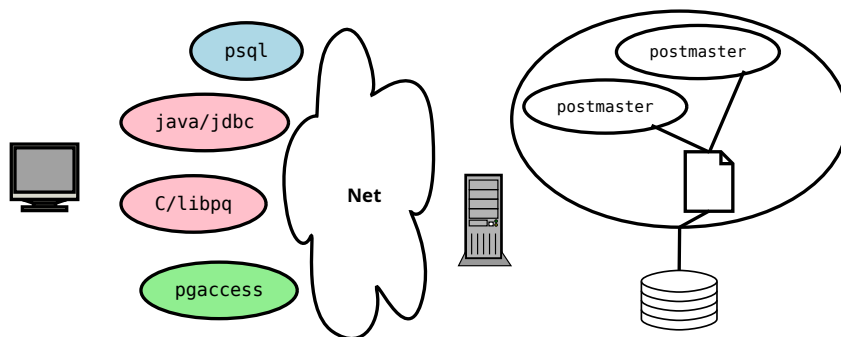
librairies applicatives jdbc odbc...

2 / 25



Postgres : clients et serveur

- accès par protocole réseau (TCP/IP) ou *socket* système



3 / 25



Michael Stonebraker

USA 1943-

1965 BSE Princeton

1967/71 MSc/PhD U. Michigan

1971 UC Berkeley

2001 MIT

2015 Turing Award 2014

Ingres 1976 1973-1979/1989
première **base relationnelle**

Postgres 1985-1993
recherche **base relationnelle-objet**

financements : DARPA, ARO, NSF, ESL Inc., commercialisation
projets : finances, astéroïdes, SIG...

Postgres v4.2 1993 fin du support de Berkeley... logiciel libre



4 / 25



Évolution comme logiciel libre. . .

Postgres

FC/CM

Éléphant

Stonebraker

Libre

Features

Hiérarchie

Interfaces

Docs

Utilisateurs

1995-1996 Postgres95 1.0 (Andrew Yu, Jolly Chen)
SQL, le nom *Postgres95* vieillit mal. . .

1997-2009 PostgreSQL 6.0 à 8.4
doc sub group trigger union. . . JDBC PL/pgSQL MVCC *hot backups*
intégrité référentielle, *write-ahead log*, jointure externe, i18n, schéma,
prepare, opts, `information_schema` Win, save point, PITR, tablespace, pgxs,
2PC, rôles, auto vacuum, performances, types, FTS, column perms. . .

2010-2019 PostgreSQL 9.0 à 12
replicationS (sync, async, cascade, logical), ORM optimisations, JSON*,
extensions. . .

2020-2025 PostgreSQL 13 à 18
requêtes parallèles, JIT, partitions, MERGE, generated columns. . .

5 / 25



Politique de gestion des versions (*versioning*)

Postgres

FC/CM

Éléphant

Stonebraker

Libre

Features

Hiérarchie

Interfaces

Docs

Utilisateurs

release 13 14 15 16 17 18 . . .

- progrès fonctionnels, **annuel**
- changement du format de stockage (*dump/restore*)
ou bien outil de migration `pg_upgrade`
- changement de système de numérotation entre 9 et 10

bugfix 13.23 14.20 15.15 16.11 17.7 18.1 . . .

- résolutions de bugs, tous les 2-4 mois
- même format de stockage, sauf si bug critique
- à suivre !

Support sur 5 ans, actuellement : 14 (nov 2024) – 18

6 / 25



Développement de Postgres

Postgres

FC/CM

Éléphant

Stonebraker

Libre

Features

Hiérarchie

Interfaces

Docs

Utilisateurs

Soutenu par des entreprises bénéficiant de postgres

core team 7 personnes, 7 entreprises, 2 continents
Peter Eisentraut, Andres Freund, Magnus Hagander, Jonathan Katz, Tom Lane, Bruce Momjian, Dave Page

committers 31 actifs sur un an
1 français (Michaël Paquier, VMware)

contributors professionnels ou amateurs

- major : 39 personnes, +10 entreprises, 4 continents
- others : +90 personnes. . .

01/2020 Microsoft rachète *Citus Data*

09/2020 EDB rachète 2ndQuadrant

7 / 25



Caractéristiques de Postgres

Postgres

FC/CM

Éléphant

Stonebraker

Libre

Features

Hiérarchie

Interfaces

Docs

Utilisateurs

relationnel langage SQL très complet, proche du standard. . .
vues, jointures, triggers, blob, sous requêtes, CTE. . .

intégrité référentielle (vérification des contraintes)

textes unicode et autres codages, localisation des opérateurs

optimiseur de requête avancé (décisionnel. . .)
indexation spaciale, textuelle, fonctionnelle, partielle. . .

extensions rules, fonctions, objets (héritage, champs multivalués. . .)
nombreuses fonctions, types, opérateurs disponibles, MySQL compat

transactions ACID, rollback, MVCC, WAL, savepoint, 2-phase commit

sécurité via SSL, Kerberos. . . accès, rôles, permissions fines

admin outils, sauvegardes. . .

8 / 25



Caractéristiques...

Postgres
FC/CM
Éléphant
Stonebraker
Libre
Features
Hiérarchie
Interfaces
Docs
Utilisateurs

Avancées

- PL** C SQL pgSQL pl py rb tcl PHP java js sh R scheme PSM V8...
- tablespace** gestion de espace et performance disques...
- duplication** synchrone ou asynchrone, logique ou base...
- postGIS** système d'information géographique
- PITR** reconstruction de l'état à une instant précis
- pooler** partage de connexions et de charge : pgbouncer, pgpool-II
- pgxn** *Postgres eXtension Network*

Manquantes

- transactions** autonomes ou nichées (?)
- multi-master** partage des écriture de la duplication
- sauvegarde** incrémentale, bas-niveau, à chaud (FS ?)



Interfaces

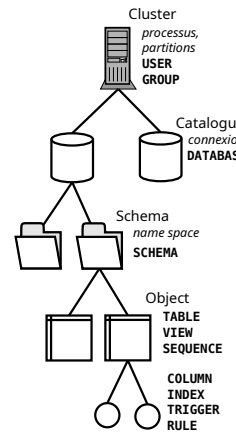
Postgres
FC/CM
Éléphant
Stonebraker
Libre
Features
Hiérarchie
Interfaces
Docs
Utilisateurs

- CLI** psql pgcli textuelle interactive *shell*
 - écriture de scripts, *pipe*...
- GUI** pgadmin4 DBeaver
- jupyter** notebook web, avec extension SQL...
- web** PhpPgAdmin, PostgreSQL Studio...
- libs** java/jdbc C/libpq C/odbc perl/dbi python/db
 - langages : java, C, C++, Perl, Python, PHP...
 - fonctions ou classes, spécifiques ou standardisées
- ORM** génération de classes
 - *MyBatis, Hibernate, SQLAlchemy*...
- ecpg** SQL *intégré* à C, traitement par préprocesseur



Hiérarchie des données

Postgres
FC/CM
Éléphant
Stonebraker
Libre
Features
Hiérarchie
Interfaces
Docs
Utilisateurs



- cluster** installation de la base
 - commandes initdb, tablespaces...
 - démarrage de postmaster avec pg_ctl
- catalogue** une base de donnée isolée
 - unité de connexion, d'accès...
- schema** groupe de tables, système ou appli.
 - espace de nommage, par défaut public
 - SET search_path TO public,pg_catalog;
- objets** relationnels
 - table view sequence...
 - ... column index trigger



Démonstration

Postgres
FC/CM
Éléphant
Stonebraker
Libre
Features
Hiérarchie
Interfaces
Docs
Utilisateurs

psql

- terminal textuel interactif à *la shell*... man psql
- commandes SQL et méta commandes \... : aide SQL \h, cmd \?
- fichier de configuration .psqlrc, historique...
- options : -h host, -U user, -l...
- configuration fichier .psqlrc

Commandes interactives

- \c ... connection à un catalogue
- \? aide sur les commandes insteractives
- \h ... aide sur les commandes SQL
- \d... description des tables, indexes, schéma...
- \q quitter psql



Illustration de pgadmin4, web/wsgi/python

Postgres

FC/CM

Éléphant

Stonebraker

Libre

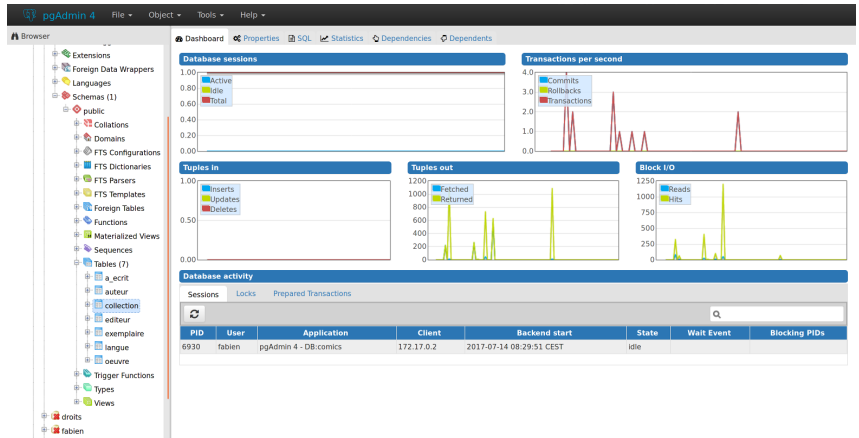
Features

Hiérarchie

Interfaces

Docs

Utilisateurs



13 / 25



PostgreSQL Studio – web/java

Postgres

FC/CM

Éléphant

Stonebraker

Libre

Features

Hiérarchie

Interfaces

Docs

Utilisateurs

The screenshot shows the PostgreSQL Studio web interface. The top navigation bar includes 'Schema', 'Columns', 'Indexes', 'Constraints', 'Triggers', 'Rules', 'Data', 'Stats', 'Script', and 'Security'. The main content area displays a table schema for 'ap' with columns: id, invnumber, transdate, entity_id, taxincluded, amount, netamount, paid, datepaid, and duedate. The table comment is 'Summary/header information for AP transactions and vendor invoices. Note that some constraints here are hard to'.

15 / 25



Illustration de DBeaver, GUI portable

Postgres

FC/CM

Éléphant

Stonebraker

Libre

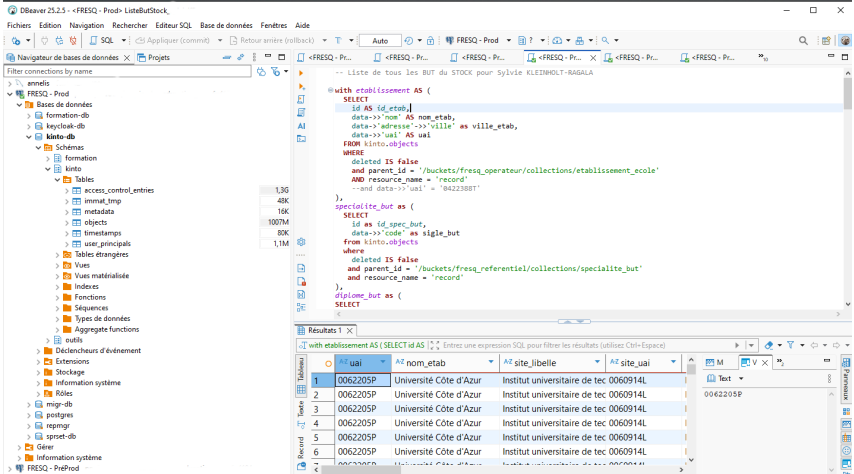
Features

Hiérarchie

Interfaces

Docs

Utilisateurs



14 / 25



Illustration de java/jdbc

Postgres

FC/CM

Éléphant

Stonebraker

Libre

Features

Hiérarchie

Interfaces

Docs

Utilisateurs

```
import java.sql.*;
public class JavaJdbc {
    static public void main(String[] args) throws Exception {
        Class.forName("org.postgresql.Driver");
        Connection c = DriverManager.getConnection(
            "jdbc:postgresql://localhost/comics",
            args[0], args[1]);
        Statement s = c.createStatement();
        ResultSet r =
            s.executeQuery("SELECT id, nom FROM auteurs");
        while (r.next())
            System.out.println(r.getInt(1)+"\t"+r.getString(2));
        r.close(); s.close(); c.close();
    }
}
```

16 / 25



Illustration de Python/DB

Postgres
FC/CM
Éléphant
Stonebraker
Libre
Features
Hiérarchie
Interfaces
Docs
Utilisateurs

```
#!/usr/bin/env python
import psycopg2 as pg
cnn = pg.connect("")
cur = cnn.cursor()
# query & result
cur.execute("SELECT firstname, lastname, email FROM Students")
for i in range(cur.rowcount):
    row = cur.fetchone()
    print(f"{i}: {row[0]} {row[1]} <{row[2]}>")
# cleanup
cur.close()
cnn.close()
```



Illustration de C/libpq

Postgres
FC/CM
Éléphant
Stonebraker
Libre
Features
Hiérarchie
Interfaces
Docs
Utilisateurs

```
#include <stdio.h>
#include <libpq-fe.h>
int main(int argc, char * argv[]) {
    PGconn * conn =
        PQconnectdb("host=/tmp dbname=comics user=coelho");
    PGresult * res =
        PQexec(conn, "SELECT nom FROM auteurs");
    int i, n = PQntuples(res);
    for (i=0; i<n; i++)
        fprintf(stdout, "nom=%s\n", PQgetvalue(res, i, 0));
    PQclear(res);
    PQfinish(conn);
    return 0;
}
```



Illustration de Python/aioSQL

Postgres
FC/CM
Éléphant
Stonebraker
Libre
Features
Hiérarchie
Interfaces
Docs
Utilisateurs

Fichier comics.sql

définition des requêtes en SQL

```
-- name: get-auteur(aid)
-- récupère les informations d'un auteur
SELECT nom, prenom, pseudo
FROM auteur
WHERE aid = :aid;
```

Fichier comics.py

utilisation des requêtes

```
#!/usr/bin/env python
import asyncio as sql
q = sql.from_path("comics.sql", "psycopg")
import psycopg2 as pg
c = pg.connect("dbname=comics")
print(f"info: {q.get_auteur(c, aid=3)}")
```

Résultat

exécution du script

```
info: [('Chabosy', 'Laurent', 'Lewis Trondheim')]
```



Illustration de C/ecpg

Postgres
FC/CM
Éléphant
Stonebraker
Libre
Features
Hiérarchie
Interfaces
Docs
Utilisateurs

```
#include <stdio.h>
int main(int argc, char * argv[])
{
    EXEC SQL BEGIN DECLARE SECTION;
    char nom[64];
    int id;
    EXEC SQL END DECLARE SECTION;
    id = atoi(argv[1]);
    EXEC SQL CONNECT TO comics;
    EXEC SQL SELECT nom INTO :nom FROM auteurs WHERE id=:id;
    fprintf(stdout, "id=%d nom=%s\n", id, nom);
    EXEC SQL DISCONNECT;
    return 0;
}
```



Illustration de PHP/Postgres

Postgres
FC/CM
Éléphant
Stonebraker
Libre
Features
Hiérarchie
Interfaces
Docs
Utilisateurs

```
<HTML><HEAD><TITLE>Auteurs de BD</TITLE></HEAD>
<BODY><H1>Auteurs de BD</H1>
<?
$conn = pg_pconnect(
    "dbname=comics user=calvin host=localhost");
$res = pg_exec($conn, "SELECT id, pseudo FROM auteurs");
for ($i = 0; $i < pg_numrows($res); $i++) {
    $id = pg_result($res, $i, 0);
    $pseudo = pg_result($res, $i, 1);
    print("id=$id, pseudo=$pseudo ");
}
?>
</BODY></HTML>
```



Documentation (html, pdf, epub) – +3300 pages A4

Postgres
FC/CM
Éléphant
Stonebraker
Libre
Features
Hiérarchie
Interfaces
Docs
Utilisateurs

<https://www.postgresql.org/docs/current/>

- 1 **tutoriel** survol rapide
- 2 **langage SQL** par le menu...
- 3 **administration** d'un serveur Postgres
- 4 **interfaces client** texte, librairies...
- 5 **programmation serveur** pour développement d'extensions
- 6 **références SQL**, clients, commandes admin
- 7 **internes** protocole, fonctionnement...
- 8 **appendixes**, en particulier notes de versions



Supports

Postgres
FC/CM
Éléphant
Stonebraker
Libre
Features
Hiérarchie
Interfaces
Docs
Utilisateurs

Dalibo <https://www.dalibo.com/formations>

- DEVPG PERF1 PERF2
- DBA1 DBA2 DBA3 HAPAT DMADM MIGORG
- SIG1 SIG2

Loxodata <https://www.loxodata.com/post/refcard2023/>

Cheat Sheet <https://www.postgresqltutorial.com/postgresql-cheat-sheet/>

Timescale <https://postgrescheatsheet.com/>



Déploiement de Postgres dans les administrations

Postgres
FC/CM
Éléphant
Stonebraker
Libre
Features
Hiérarchie
Interfaces
Docs
Utilisateurs

Circulaire 35837 de JM Ayrault du 19 septembre 2012

- PM/SGG/DISIC
- exemple +400 applications METL et MEDDE en 2013 :
 - 70% avec PostgreSQL, 3 To de données
 - 25% avec Oracle, 2 To de données
- Apache/Weblogic/Oracle → Apache/Tomcat/PostgreSQL
- déploiement de PostGIS : agriculture, intérieur, équipement, IGN
- guide d'usage et de migration vers PostgreSQL



Grandes entreprises actives à PostgreSQLFr

- Postgres
- FC/CM
- Eléphant
- Stonebraker
- Libre
- Features
- Hiérarchie
- Interfaces
- Docs
- Utilisateurs

<https://www.postgresql.fr/entreprises>