



## Retour d'expérience PostgreSQL

Domaine Solutions et Urbanisation

Jeudi 24 septembre 2015



Emmanuel REMY  
@manuremy

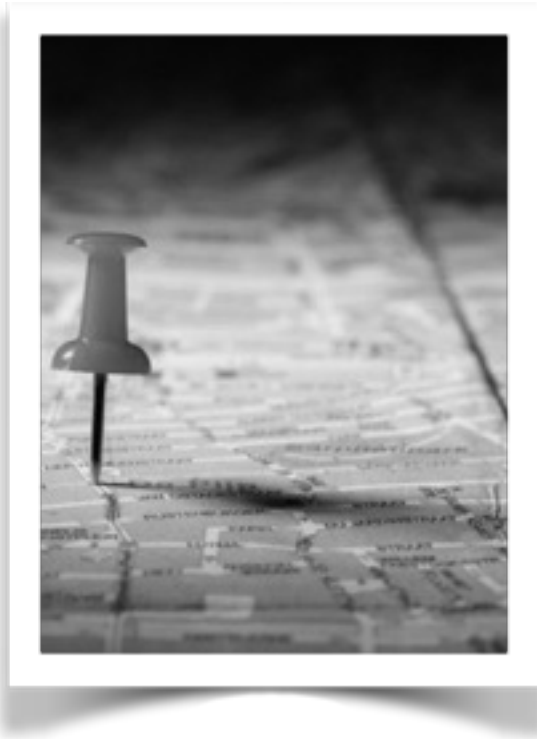
# Points abordés

---



PostgreSQL





## Les enjeux

- **Fragmentation de notre porte-feuilles de bases -> réorganiser, simplifier, urbaniser**
- Volonté de rester connecter avec les réalités des devs... **mais dans un cadre bancaire**
- Maîtrise des évolutions du métier (le **#Digital** est partout !)
- Cadre réglementaire toujours plus présent -> DWH
- ~~Demain~~ Aujourd'hui le BigData ?





# Ce serait trop simple

- **Lourdes contraintes de sécurité (PSSI)**
- L'impact des institutions (PILCOP, BCE)
- Une petite DSI (65 pers, 3 DBA)
- Des budgets raisonnables, sans être monstrueux
- Des contraintes liées aux progiciels
- Les avis des Consultants





## Une DSI facilitatrice

- **A l'écoute** des idées et des tendances
- Culture de l'expérimentation via des **Labs** et de la Veille
- Culture du partage en interne
- **Agilité pour rapidement convaincre**
- Un **SI producteur**
- **Business Case** éligible
- **Concrétiser** l'usage de la donnée dans le temps



# Notre cible

---



ORACLE®



ORACLE®



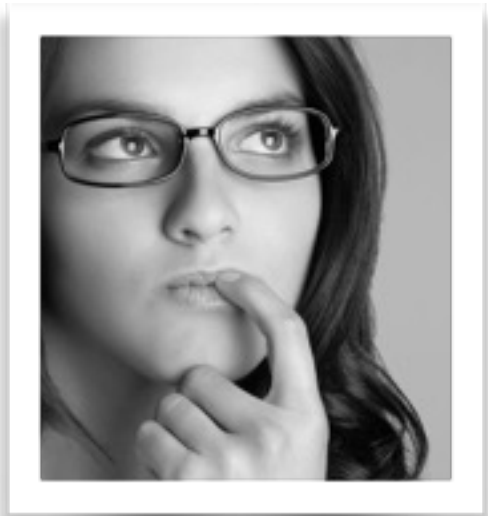
PostgreSQL



## Ne pas se perdre

- **Prendre le pouls de la communauté, être ambassadeur en interne**
- Inventaire technique, technologique, PG
- Démarches vers les Editeurs de progiciels
- Acquisition d'un socle de compétences -> Formations
- Architecture, sécurité, prototype
- **Se faire conseiller**
- Go au bon moment (1 an après)





# Etre accompagné

- **Préconisations par rapport à notre existant**
- Sauvegardes/Restauration
- Performances (dont chargements)
- Réplication ou Haute dispo ?
- ESX, datacenter, cloud ?
- Load Balancing (on fait quoi de notre f5 ?)
- Supervision
- Chiffrement, pistes d'audit PSSI
- Migrations depuis Oracle et MySQL
- Capacités OLAP (partitionnement ?)







## Attentions

- **Aujourd'hui Couchbase nous sert à...**
- **Les « on dit »: Requêtes trop lentes dans PG**
- **« On fait du NoSQL »**
- Simplicité d'accès en REST (cela a changé)
- Interface Web de consultation
- Architecture cluster
- Traitement des flux JSON en JS
- API interne CASDEN existante
- Fonctionnalités *Time lo live*





... mais

- Faible volumétrie
- Type JSONB dans PG
- Indexation GIN
- Transactions ACID
- Requêtes natives sur un flux natif JSON (mais nouveaux opérateurs à maîtriser)
- Requêtes complexes en profitant du SQL (CTE, Window etc...)
- **+ moteur PLV8 pour développer des compléments**



# Cas de Couchbase



```
create table my_docs(  
  id serial primary key,  
  body jsonb not null,  
  search tsvector,  
  created_at timestamptz not null default now(),  
  updated_at timestamptz not null default now()  
)
```

```
create function create_document_table(name varchar, out boolean)  
as $$  
var sql = "create table " + name + "(" +  
  "id serial primary key," +  
  "body jsonb not null," +  
  "search tsvector," +  
  "created_at timestamptz default now() not null," +  
  "updated_at timestamptz default now() not null);";  
  
plv8.execute(sql);  
plv8.execute("create index idx_" + name + " on docs using GIN(body  
jsonb_path_ops)");  
plv8.execute("create index idx_" + name + "_search on docs using  
GIN(search)");  
return true;  
$$ language plv8;
```

```
with items as (  
  select body -> 'id' as invoice_id,  
  (body ->> 'completed_at')::timestamptz as date,  
  jsonb_array_elements(body -> 'items') as sale_items  
  from sales  
)  
), fact as (  
  select invoice_id,  
  date_part('quarter', date) as quarter,  
  date_part('year', date) as year,  
  date_part('month', date) as month,  
  date_part('day', date) as day,  
  x.*  
  from items, jsonb_to_record(sale_items) as x(  
    sku varchar(50),  
    vendor varchar(255),  
    price int,  
    quantity int  
  )  
)  
select * from fact;
```





## Aucun regret... pour nous !

- Cela ne peut s'inscrire QUE dans un cadre de migrations plus conséquentes (infra, archi, Oracle, MySQL)
- Gros travail d'accompagnement des devs pour les architectes applicatifs
- **Le PSSI est heureux**
- On a pu capitaliser sur JS
- On étend la démarche à... XML (par exemple pour des flux Banque de France)
- **Ce n'est qu'une étape**



# PostgreSQL

---



Question (pas plus d'une) ?

