

# Accélérez vos données : Maîtrisez la Capture de Changement en Temps Réel

**Benjamin Djidi**  
Co-fondateur, CEO  
Popsink  
[ben@popsink.com](mailto:ben@popsink.com)



**Stéphane Lambin**  
Directeur Service et  
Innovation - Synergy  
[slambin@synergy.fr](mailto:slambin@synergy.fr)



Pourquoi ?

**Moderniser** les capacités data  
de manière **robuste, évolutive,**  
et **sécurisée** - en minimisant  
l'impact de la Data sur les  
les systèmes de production.

Technologie

# Capture de Changements

La botte secrète pour une ingestion consistante et robuste entre systèmes hétérogènes: **répliquer les changements.**

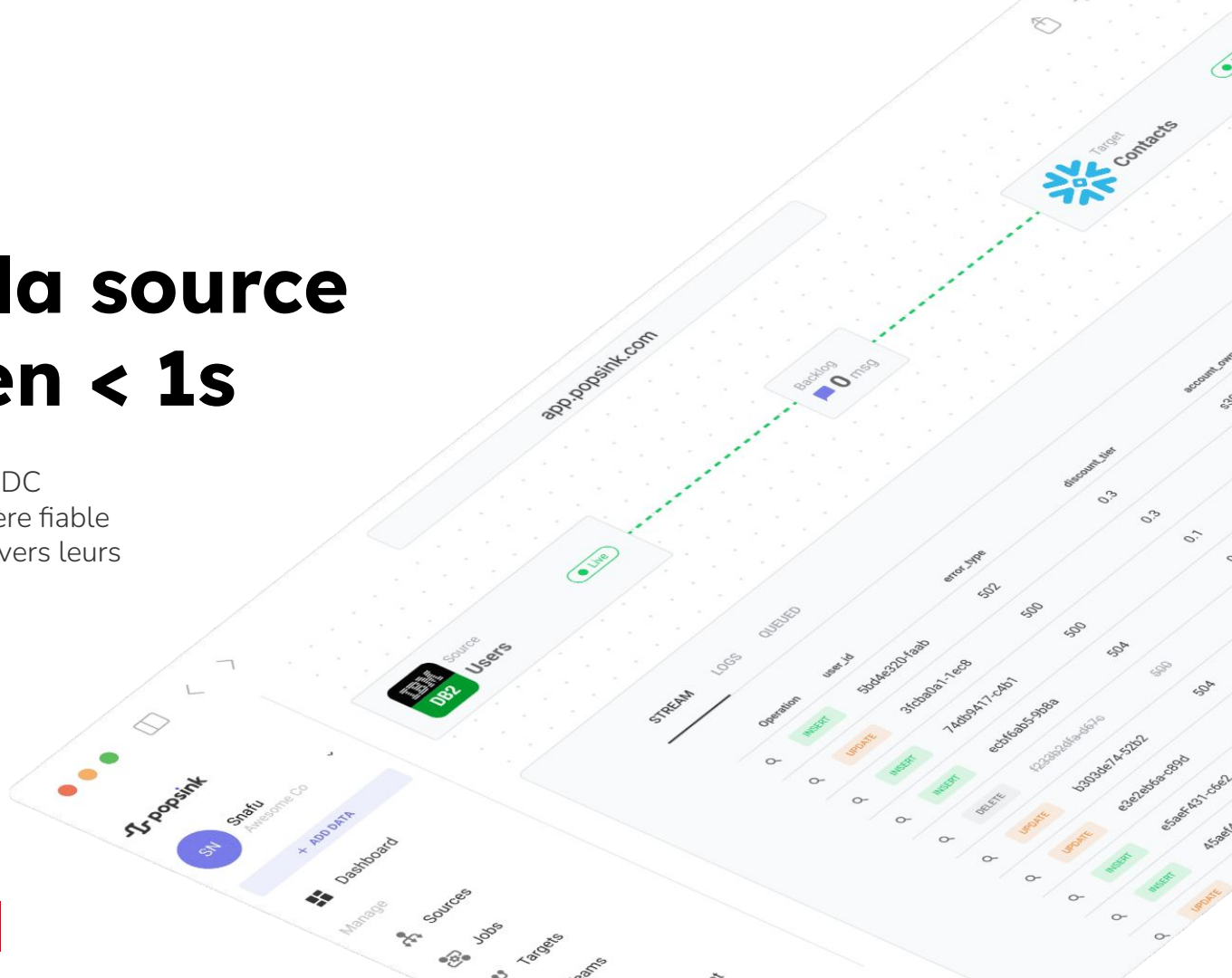
STREAM					LOGS	QUEUED
	Operation	user_id	error_type	disc		
Q	INSERT	5bd4e320-faab	502	0.3		
Q	UPDATE	3fcba0a1-1ec8	500	0.3		
Q	INSERT	74db9417-c4b1	500	0.1		
Q	INSERT	ecbf6ab5-9b8a	504	0.1		
Q	DELETE	f233b2dfa-d67e	500	0.3		
Q	UPDATE	b303de74-52b2	504	0.4		
Q	UPDATE	e3e2eb6a-c89d	502	0.1		
Q	INSERT	e5aeF431-c6e2	502	0.3		
Q	INSERT	45aef445-b476	500	0.5		
Q	UPDATE	84aef331-c5b8	500	0.3		

Modernisation

# Passer de la source à la data en < 1s

Les pipelines incrémentaux en CDC déplacent vos données de manière fiable et temps réel - de leurs sources vers leurs destinations.

- ✓ Réplication sans perte
- ✓ Récupération Automatique
- ✓ Compatible Mainframe
- ✓ Intégrations ERP & CRM

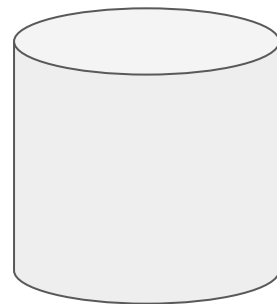


# Méthode Historique

SOURCE

USER_ID (PK)	FIRST_NAME	LAST_NAME
3	Paul	Dupont
2	Jane	Doe
1	Foo	Bar
0	Axel	Thevenot

DESTINATION



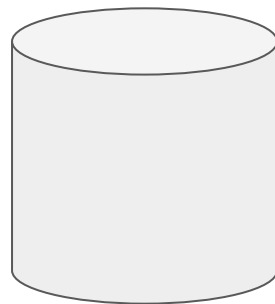
SOURCE

USER_ID (PK)	FIRST_NAME	LAST_NAME
3	Paul	Dupont
2	Jane	Doe
1	Foo	Bar
0	Axel	Thevenot

```
SELECT *  
FROM <table>  
WHERE <clause>
```



DESTINATION



SOURCE

USER_ID (PK)	FIRST_NAME	LAST_NAME
3	Paul	Dupont
2	Jane	Doe
1	Foo	Bar
0	Axel	Thevenot

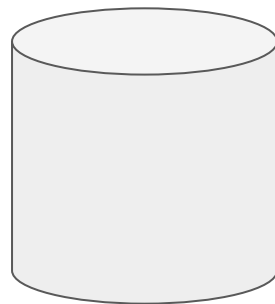
```
SELECT *  
FROM <table>  
WHERE <clause>
```



```
INSERT INTO  
UPSERT  
MERGE
```



DESTINATION



# Limite 1 : DELETE

USER_ID (PK)	FIRST_NAME	LAST_NAME
3	Paul	Dupont
2	Jane	Doe
1	Foo	Bar
0	Axel	Thevenot

INSERT INTO  
UPSERT  
MERGE



USER_ID (PK)	FIRST_NAME	LAST_NAME
3	Paul	Dupont
2	Jane	Doe
1	Foo	Bar
0	Axel	Thevenot



# Limite 2 : Consistance

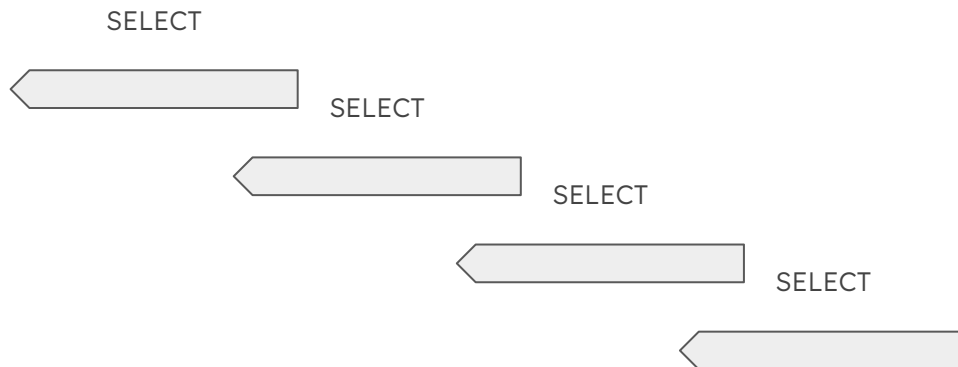
USER_ID (PK)	FIRST_NAME	LAST_NAME
3	Paul	Dupont
2	Jane	Dupont
1	Foo	Bar
5	Albus	Dumbledore
0	Axel	Thevenot



USER_ID (PK)	FIRST_NAME	LAST_NAME
3	Paul	Dupont
2	Jane	Doe
1	Foo	Bar
0	Axel	Thevenot

# Limite 3 : Impact Source

USER_ID (PK)	FIRST_NAME	LAST_NAME
3	Paul	Dupont
2	Jane	Doe
1	Foo	Bar
0	Axel	Thevenot
+ beaucoup de lignes		



TABLE

USER_ID (PK)	FIRST_NAME	LAST_NAME
3	Paul	Dupont
2	Jane	Doe
1	Foo	Bar
0	Axel	Thevenot

JOURNAL

OPERATION	USER_ID	FIRST_NAME	LAST_NAME
INSERT	3	Paul	Dupont
DELETE	4		
UPDATE	2	Jane	Doe

## Le Journal Transactionnel

Enregistre les modifications effectuées sur les données, de manière séquentielle, avant qu'elles ne soient appliquées sur les fichiers de données réels.

- Durabilité
- Audit
- Rollback
- Réplication

Des noms différents :

WAL (PostgreSQL)  
Binlog (MySQL / MariaDB)  
Redo Log (Oracle)  
Transaction Log (SQL Server)  
Oplog (MongoDB)  
Journal/Receiver (DB2)

...

# CDC

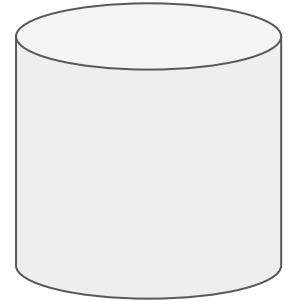
SOURCE

OPERATION	USER_ID	FIRST_NAME	LAST_NAME
INSERT	3	Paul	Dupont
DELETE	4		
UPDATE	2	Jane	Doe

INSERT INTO  
et/ou MERGE



DESTINATION



## EXTRACT / LOAD (EL)



### **Impact Source**

Scanner de grandes quantités de lignes sur les tables sources impacte la performance.



### **Inconsistance**

Impossible de savoir où en est la synchronisation et de reproduire les états intermédiaires.



### **Pas de DELETE**

Les mécanismes de SELECT ne capturent que la donnée qui existe dans les tables.

## CAPTURE DE CHANGEMENT (CDC)



### **Impact Minimisé**

en isolant les ressources des pipelines Data de celles de l'usage opérationnel.



### **Consistance Garantie**

de la donnée par un mécanisme capable de tracker la synchronisation.



### **Capture des DELETES**

grâce au journal transactionnel capable de communiquer les suppressions.

Solution miracle ?

# Bénéfices du CDC



## Opérations Hyper Simplifiées

Réplication automatique et sans perte avec récupération automatique grâce au séquençage des journaux.



## Replication Continue

Pas d'orchestration grâce au push automatique des transactions du journal en temps réel.



## Scalabilité

Répliquer vers plusieurs destinations n'augmente pas la charge sur les systèmes source.



## Time-travel

Voyager dans le temps pour rejouer un état passé de la donnée à un instant T.

Solution miracle ?

# Limites du CDC



## **Pas fait pour du call API**

Le CDC n'est simplement pas fait pour orchestrer des calls API.



## **Capacités de la Source**

Le CDC dépend de la capacité de la source à fournir un journal consistant et séquencé.



## **Configuration Source**

Mettre en place le CDC demande de configurer le journal et les droits associés.



## **Dépendance à la Rétention**

La rétention des logs impacte directement la résistance opérationnelle - particulièrement le replay.

TABLE

USER_ID (PK)	FIRST_NAME	LAST_NAME
3	Paul	Dupont
2	Jane	Doe
1	Foo	Bar
0	Axel	Thevenot

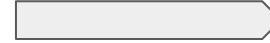
JOURNAL

OPERATION	USER_ID	FIRST_NAME	LAST_NAME
INSERT	3	Paul	Dupont
DELETE	4		
UPDATE	2	Jane	Doe

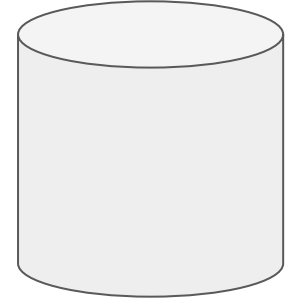
SELECT \*  
FROM <table>  
WHERE <clause>



INSERT INTO  
UPSERT  
MERGE

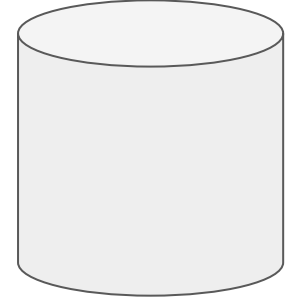


DESTINATION



DESTINATION

INSERT INTO  
et/ou MERGE

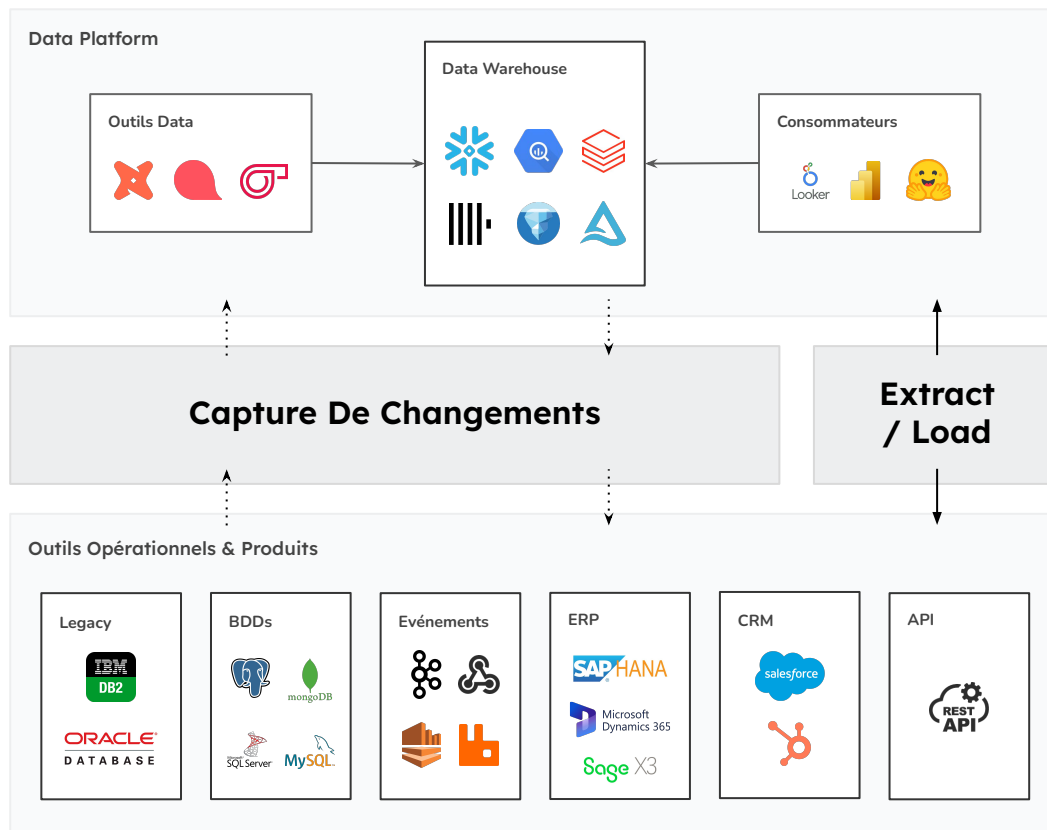




Architecture

# Pas de Data, pas d'IA

- ✓ Move-to-Cloud
- ✓ Centraliser la donnée référentielle
- ✓ Péreniser les ressources on-premise
- ✓ Synchroniser les outils



# Les chapitres

1. Apports du temps réel côté Business
2. Popsink : exemple Synergy
3. Apport de Popsink pour l'IT





# Les gains Business du temps réel

# Les gains Business du temps réel

## Quelques exemples :

- Détection de **fraudes** : usage de l'IA en temps réel pour détecter des fraudes potentielles et les bloquer le plus rapidement possible
- **Stock** à jour sur le site Web
- **Reporting opérationnel** : on visualise les changements immédiatement. C'est, par exemple, très pratique en finance lors de la clôture mensuelle
- Mise en place d'un **DataHub** pour les échanges inter-applicatifs.
- Gestion des **appros** au plus juste, plusieurs fois par jour sur des données les plus fraîches possibles



# Les gains Business du temps réel

## Quelques exemples :

- Suivi des ventes en magasins et web en **temps réel** :
  - Comparaison des ventes du magasin et au niveau national.
  - Permet de connaître les éventuels produits qui cartonnent ailleurs ! Cela permet de se comparer, de réagir rapidement et d'**augmenter les ventes**.
  - Meilleure **gestion des promotions** en analysant les ventes en temps réel
- Réagir à des **avis clients** le plus rapidement possible
- **Ajustement des prix** en fonction de la demande : non plus une fois par jour sur les données de la veille mais plusieurs fois par jour sur des données fraîches.





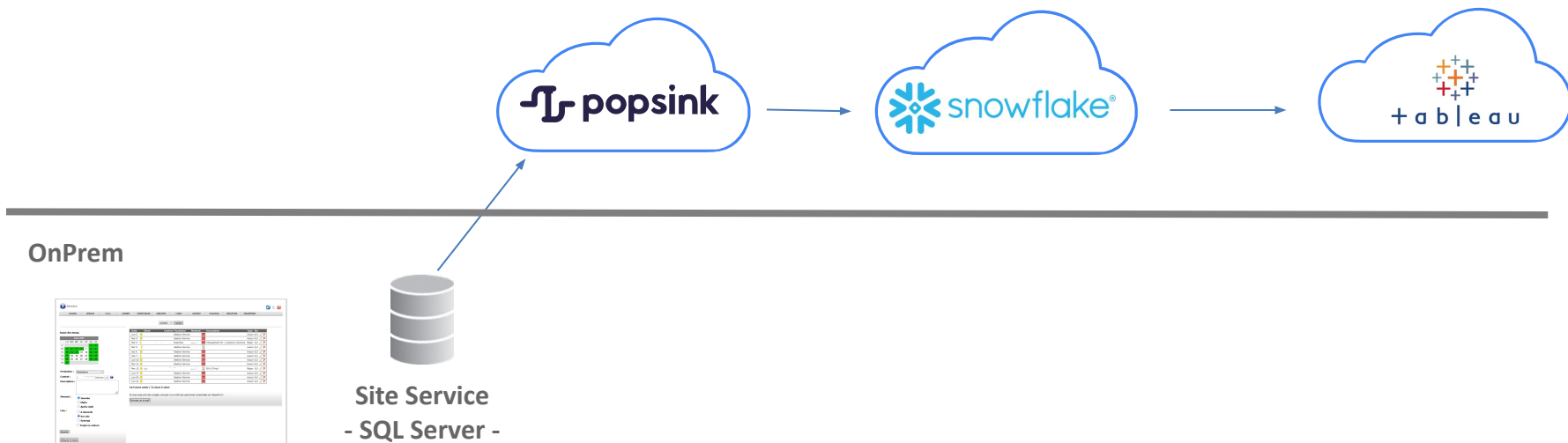
# **POPSINK – Usage chez SYNERGY**



# Usage de Popsink chez Synergy

## L'architecture cible :

- Utilisation en version **SAAS** : pas de serveur, d'installation, pas de mise à jour, pas de sauvegarde...
- Rapide à mettre en place et facturé en fonction des usages
- Avec Popsink : vrai **temps réel** sans impact sur l'application source.



# Usage de Popsink chez Synergy

## Prérequis :

- Activation CDC sur SQL-Server
- Ouverture IP Popsink
- Base + compte sur Snowflake

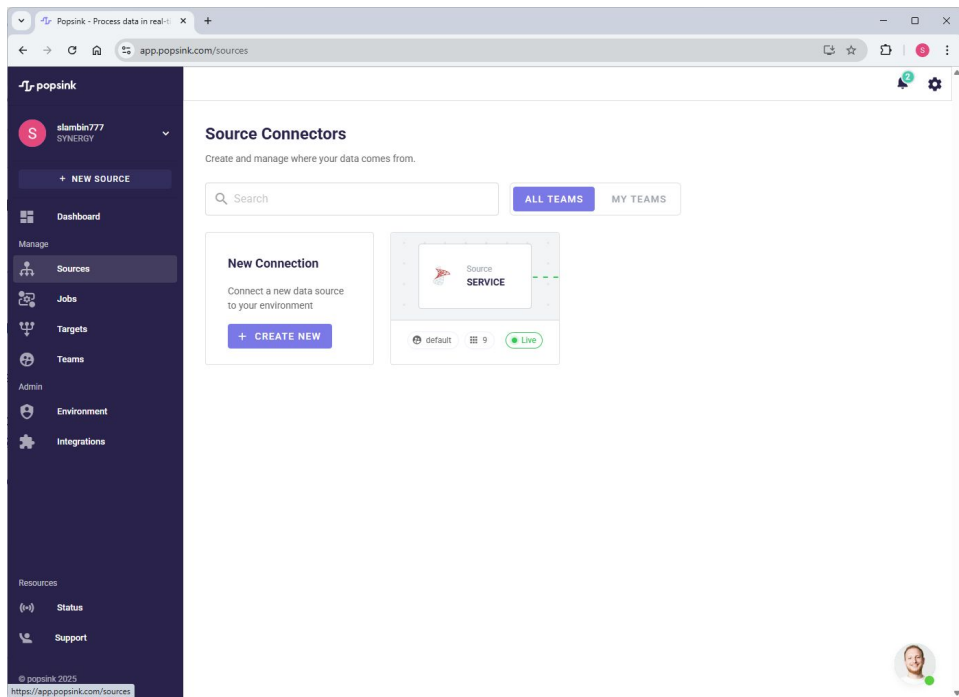
## Dans le portail Popsink :

1. Création de la source SQL-Server
2. Choix des tables à surveiller
3. Création de la destination Snowflake
4. Souscription de la source aux tables surveillées en CDC

Et c'est fini !

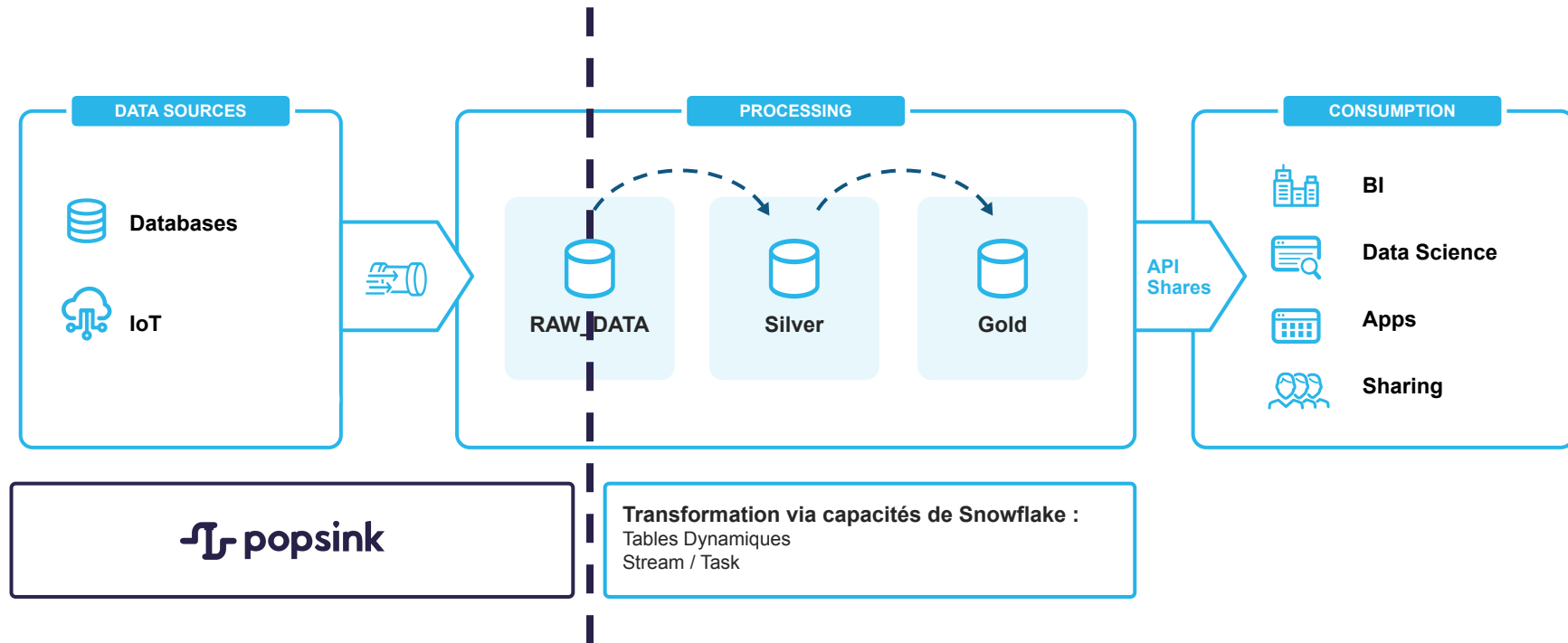
Les tables sont créées et gérées par Popsink.

Latence constatée : environ 6 à 7 secondes





# Usage de Popsink chez Synergy





# Les apports de Popsink pour l'IT

## Les apports techniques de Popsink :

- Popsink **facilite** et **accélère** la mise en place d'un **Stack Data moderne**
- Permet de créer un **DataLake sans effort**.
- L'ingestion des données brutes est entièrement gérée par Popsink :
  - Pas besoin de faire un job par table
  - Pas besoin de choisir entre full ou delta et de gérer la détection des modifications.
  - Popsink crée les tables et gère les modifications (ex : ajout de colonnes).
  - Aucune perte d'information : vous suivez toutes les modifications de données. Au choix, vous pouvez gérer le SCD (Slow Changing Dimension) ou avoir une vision iso à la prod. Avec la possibilité de choisir plus tard car tout est conservé.



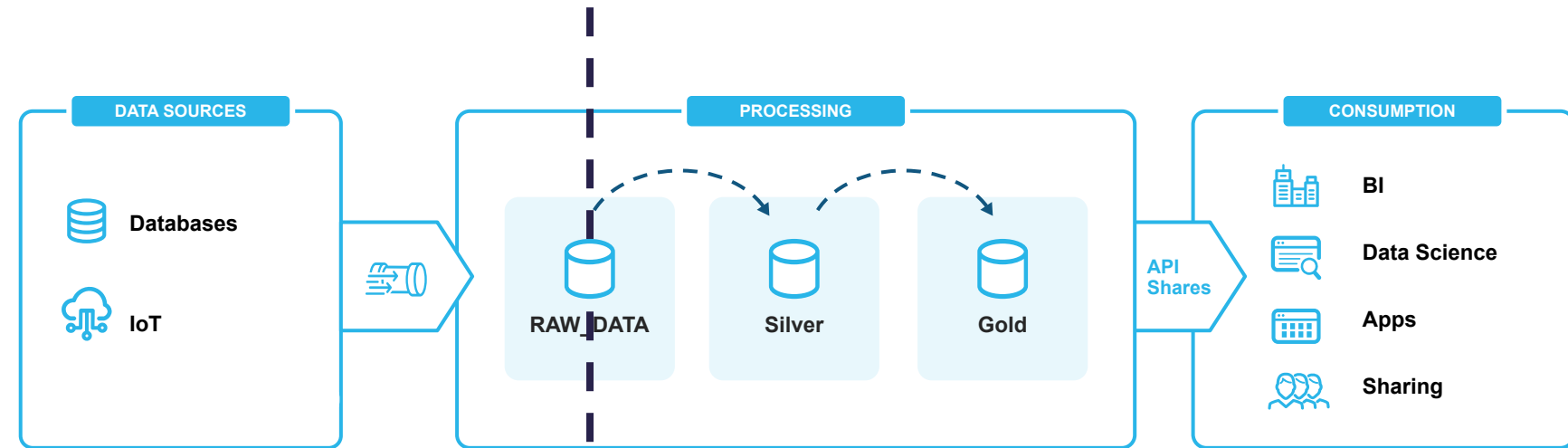
# Les apports de Popsink pour l'IT

## Les apports techniques de Popsink :

- Fini les traitements batch qui tournent au petit matin !  
En cas de souci, il est possible de **réagir durant la journée**
- S'il y a problème, le **retard est minime** et la reprise rapide et automatique.



# Les apports de Popsink pour l'IT



 popsink

**Vous vous concentrez sur l'essentiel :**

- La qualité et l'enrichissement des données
- Répondre aux cas d'usage métiers

Les transformations, en temps réels ou non, peuvent être traitées avec les capacités de la plateforme (SQL, Python) ou avec des outils comme DBT.



Next

Des questions ???

 popsink

