



FLYWAY

DATABASE MIGRATION TOOL

Présenté par Hjaiej Mohamed



X PROBLEMATIQUE



Toutes les mises à jour des bases de données de test et de production sont effectuées manuellement.

X Risque de désynchronisation entre les schémas dans les divers environnements.



Pas d'automatisation

X Chaque changement et mise à jour de la base de données de production nécessite une intervention de l'administrateur.



Pas de Versionnement (Versionning)

X Pas de traçabilité sur les mises à jour de la base de données, ce qui rend le retour en arrière vers des versions précédentes difficile.

+ MOTIVATION : Pourquoi Flyway?



**Versionnement de la
base de données**



**Automatisation des
migrations**



**Réversibilité des
migrations**



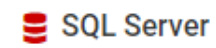
**Intégration avec : Spring
Boot**



**Intégration avec Jenkins
pipelines**

+ MOTIVATION : Pourquoi Flyway?

Supporte:



SQL Server



Azure SQL Database



Azure SQL Database
Managed Instance



SQL Server - Amazon RDS



SQL Server - Google Cloud



Azure Synapse



Oracle Database



Oracle Autonomous
Database



Oracle Base Database
Service



Oracle Exadata Database
Service



Oracle - Amazon RDS



PostgreSQL



PostgreSQL - Amazon RDS



PostgreSQL - Google Cloud



Azure PostgreSQL - Flexible
Server



Azure PostgreSQL - Single
Server



Aurora PostgreSQL



EnterpriseDB - EDB Postgres
Advanced Server (preview)



YugabyteDB



CockroachDB



MySQL



MySQL - Amazon RDS



MySQL - Azure



MySQL - Google Cloud



Aurora MySQL



Percona Server for MySQL



Percona XtraDB Cluster



MariaDB



MariaDB - Amazon RDS



MariaDB SkySQL



IBM DB2



Derby



Firebird



Google BigQuery



Google Cloud Spanner



H2



HSQLDB



Ignite



Informix



MongoDB (preview)



MongoDB (preview) -
Amazon DocumentDB



Redshift



SAP HANA



SAP HANA Cloud



SingleStoreDB



Snowflake



SQLite



Sybase ASE



Testcontainers



TiDB

+ MOTIVATION : Open Source ? OUI

Community includes:

- ✓ Flyway API/CLI core functionality
- ✓ Flyway Desktop GUI
- ✓ 6 basic commands: Migrate, Clean, Info, Validate, Baseline and Repair
- ✓ Support for current DB versions
- ✓ Community support

Community

Perfect for individual developers, or non-commercial projects looking for a basic and reliable framework for versioning and automating the deployment of database changes

Free

Open Source
Support from the community

[Download](#)

[Learn more >](#)

Community includes:

- ✓ Flyway API/CLI core functionality
- ✓ Flyway Desktop GUI
- ✓ 6 basic commands: Migrate, Clean, Info, Validate, Baseline and Repair
- ✓ Support for current DB versions
- ✓ Community support

Enterprise

Ideal for enterprises wanting to harness the power of script auto-generation and advanced controls to deliver database changes with increased speed, reliability and quality

Contact us

Get a quote
Redgate support included

[Get in touch](#)

[Free trial >](#) [Learn more >](#)

Everything in Teams plus:

- ✓ Change reports**
- ✓ Drift detection**
- ✓ SQL code standard checks**
- ✓ Migration script auto-generation**
- ✓ Schema comparison**
- ✓ Static data versioning***
- ✓ Data comparison***
- ✓ Support for extended set of DB versions

**SQL Server, Oracle, PostgreSQL and MySQL only.

***SQL Server and Oracle only.

Teams

Ideal for organizations looking to improve collaboration and fine tune their processes during development and the deployment of database changes

\$597 USD*

user/year
Redgate support included

[Buy now](#)

[Free trial >](#) [Learn more >](#)

*100 schema limit.

Contact us if you have a requirement for more

Everything in Community plus:

- ✓ Dry run
- ✓ Undo migration scripts
- ✓ Cherry pick
- ✓ Skip migrations (mark as deployed)
- ✓ Object-level versioning**
- ✓ Built-in Git client
- ✓ Support for older DB versions
- ✓ Access to technical support

**SQL Server, Oracle, PostgreSQL and MySQL only.

+ MOTIVATION : Base de données existante ? OUI



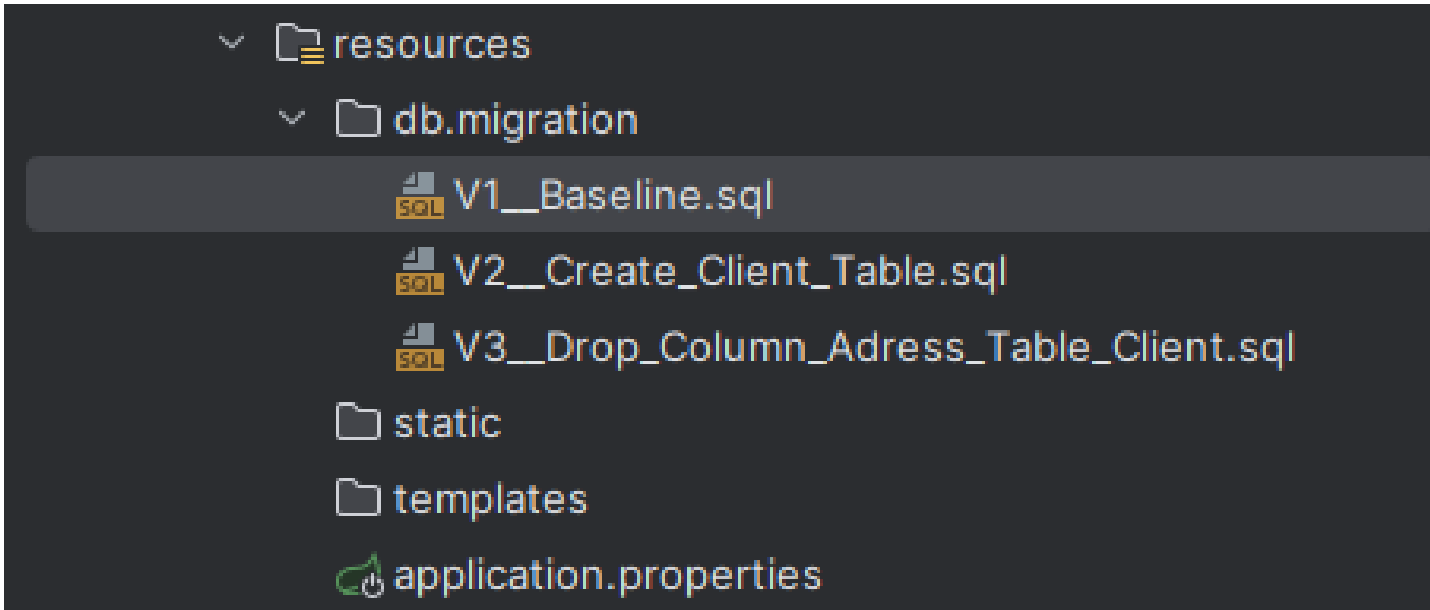
Baseline:

La fonction de baseline dans Flyway offre la possibilité de commencer à l'utiliser avec une base de données existante.

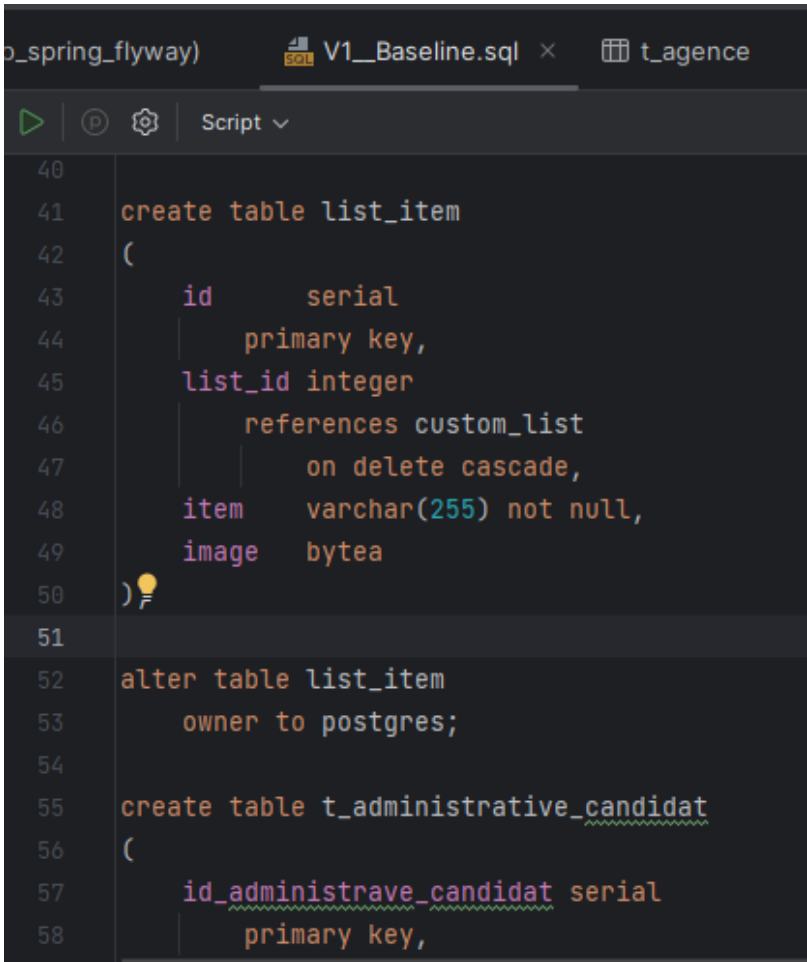
- **Baseline** dans flyway est un script **SQL** qui représente le **schéma** de la base de données **existante**.
- **Si la base de données est vide:** Flyway va créer / initialiser les tables, facilitant ainsi la migration d'un environnement vers un autre (de Prod vers Dev, par exemple).
- **Si la base de données existe déjà:** Flyway commencera à exécuter les scripts SQL de migration créés par le développeur, sans altérer les tables et les données existantes.

+ Spring Boot 3 + Flyway : Example

1



2



3

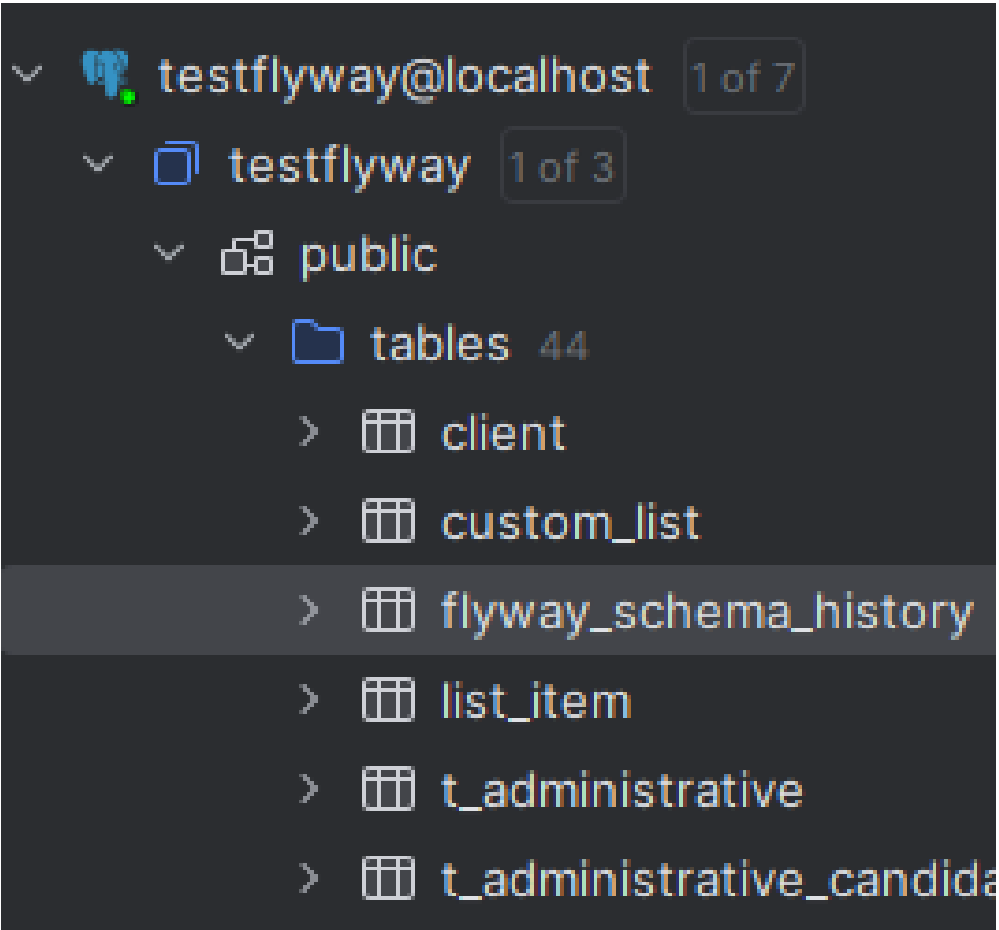


Table: flyway_schema_history

4

	installed_rank	version	description	type	script	checksum	installed_by	installed_on
1	1	1	<< Flyway Baseline >>	BASELINE	<< Flyway Baseline >>	<null>	postgres	2024-01-15 01:47:25.06
2	2	2	Create Client Table	SQL	V2__Create_Client_Table.sql	-604912420	postgres	2024-01-15 01:47:25.28
3	3	3	Drop Column Adress Table Client	SQL	V3__Drop_Column_Adress_Table_Client.sql	1680662096	postgres	2024-01-15 01:47:25.55

+ Spring Boot 3 + Flyway: Configuration avec Maven

```
35 # -----Flyway Config-----
36 spring.flyway.baseline-on-migrate=true
37 spring.flyway.enabled=true
38 spring.flyway.locations=classpath:db/migration
```

Configuration de Flyway dans le fichier `application.properties`
[baseline = true]

```
3 #-----DB Connection-----
4 spring.datasource.driver-class-name=org.postgresql.Driver
5 spring.datasource.url=jdbc:postgresql://localhost:5432/testflyway
6 spring.datasource.username=postgres
7 spring.datasource.password=postgres
8
```

Configuration de la base de données dans le fichier `application.properties`

```
<dependency>
  <groupId>org.flywaydb</groupId>
  <artifactId>flyway-core</artifactId>
</dependency>
```

Configuration de dépendances de flyway dans le fichier `pom.xml`

+ Spring Boot 3 + Flyway

Demo