Projet Qualité des Données : ETL Consommation Énergétique

Khaled Bouabdallah Théo Joly Mohammed Nassim Fellah
Sarah Boundaoui

M2 DataScale 2025/2026
Zoubida Kedad

Dernière mise à jour : 11 novembre 2025

Table des matières

1 Conception de Mapping : Approche Union-First

1.1 Stratégie

Fusionner les sources tôt, transformer une fois, séparer les sorties tard.

1.2 Pipeline

```
S1 (Paris) + S2 (Evry) -> Union -> Transform -> Split -> Cibles
```

1.3 Pourquoi union-first

— Pas de logique dupliquée pour Paris/Évry
— Cohérence garantie entre les sources
— Comparaison facile des sources via la colonne `Source`
— Passage à l'échelle vers de nouvelles villes sans modification de l'ETL

1.4 Détails d'implémentation

— Ajout d'une colonne `Source` ('Paris' ou 'Evry') lors de l'union
— Utilisation de clés composites : `ID` → `ID_Source`, `ID_Adr` → `ID_Adr_Source`
— Filtrage par `Source` uniquement à l'étape finale pour les tables cibles séparées

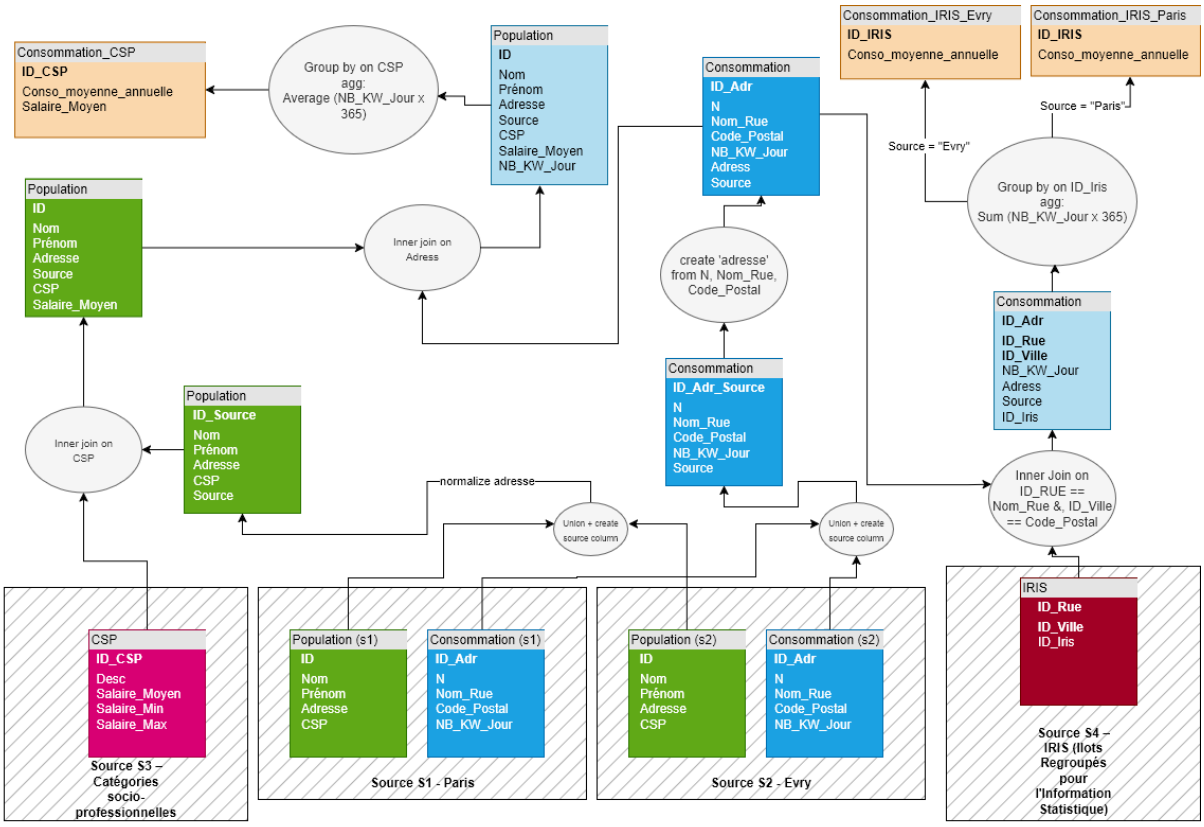

FIGURE 1 – Diagramme de Mapping

1.5 Résumé des Flux

1.5.1 Chemin 1 : Consommation_CSP

```
Population (S1+S2) -> Join CSP -> Enrichissement avec Salaire_Moyen
   |
   v
Consommation (S1+S2) -> Creation adresse complete
   |
   v
Join sur Adresse -> Group by CSP -> AVG(consommation), MAX(salaire)
```

1.5.2 Chemin 2 : Consommation_IRIS (Paris/Évry)

```
Consommation (S1+S2) -> Creation adresse complete
   |
   v
Join IRIS (ID_Rue==Nom_Rue, ID_Ville==Code_Postal)
   |
   v
Group by ID_IRIS -> SUM(consommation)
   |
   v
Split par Source -> Cible Paris, Cible Evry
```

2 Implémentation des mappings

2.1 Copie d'écran de l'implémentation du job

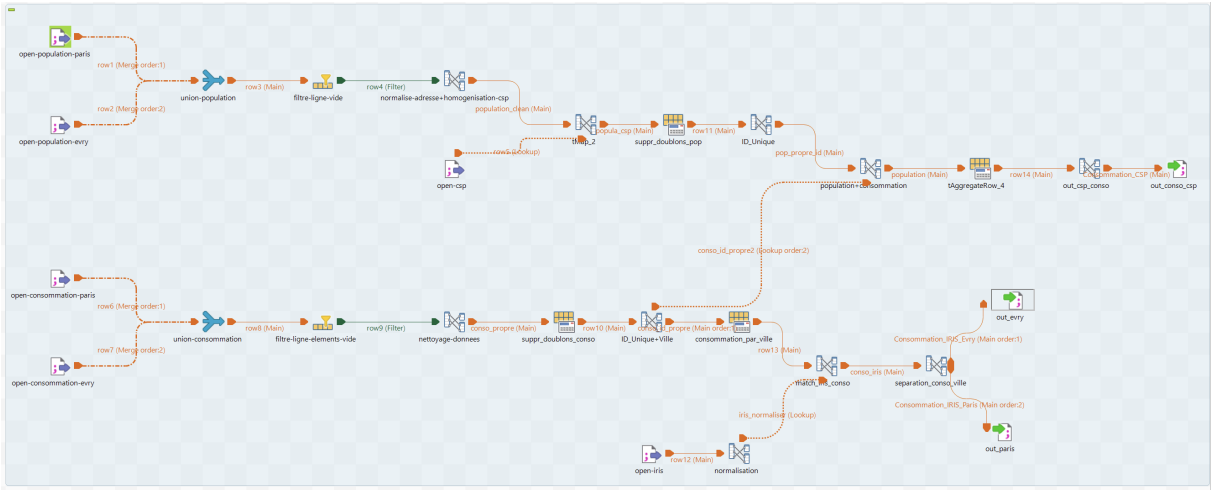

FIGURE 2 – Implémentation de Mapping

L'implémentation des mappings a été réalisée à l'aide de l'ETL Talend.

2.2 Problèmes rencontrés

Lors de l'implémentation, plusieurs difficultés ont été rencontrées :

- **Formatage des adresses :**
  Les adresses n'étaient pas uniformes entre les différentes sources. Nous avons donc choisi de les décomposer en plusieurs champs : numéro, nom de rue, ville et code postal. Nous avons également supprimé les accents, converti l'ensemble en minuscules et traité les cas particuliers où certaines adresses étaient manquantes. De plus, certains numéros de rue contenaient des lettres ou des suffixes tels que A, B, C, BIS ou TER, ce qui a nécessité un traitement spécifique.
- **Codes postaux :**
  Le format des codes postaux n'était pas toujours conforme : certains contenaient un caractère en trop ou n'étaient pas composés de cinq chiffres.
- **CSP (Catégorie Socio-Professionnelle) :**
  Nous avons dû gérer deux cas distincts selon les sources :
  - l'une faisait référence à l'identifiant du CSP,
  - l'autre à la description (le nom) du CSP.

  Il a donc fallu établir une correspondance entre ces deux formats à l'aide de la table de référence CSP.
- **Échelles de consommation :**
  Les valeurs de consommation n'étaient pas exprimées dans la même unité : certaines en Watt (W), d'autres en Kilowatt (kW). Une mise à l'échelle a donc été effectuée pour harmoniser les données.
- **Normalisation des chaînes de caractères :**
  Avant d'effectuer les jointures, toutes les chaînes ont été converties en minuscules et les accents supprimés afin d'éviter les divergences lors des comparaisons.

— **Problèmes de jointure sur les noms de rue :**
Les formats différaient selon les sources : certaines ne contenaient que le nom de la rue, tandis que d'autres incluaient le type de voie (rue, boulevard, allée, etc.). Ce décalage a nécessité un nettoyage et une harmonisation supplémentaires avant la jointure.

3 Règles de Transformation

3.1 Séparation des composants d'adresse

Pour harmoniser les données, les adresses ont été nettoyées puis décomposées en quatre éléments : numéro, nom de rue, ville et code postal.

3.1.1 Nettoyage

Avant l'extraction, les adresses ont été normalisées :
— suppression des accents, guillemets et espaces inutiles,
— uniformisation des majuscules/minuscules,
— correction des séparateurs (espaces, virgules).

3.1.2 Extraction

À l'aide d'expressions régulières :
— le code postal est identifié comme une suite de cinq chiffres ;
— le numéro de voie correspond à une suite de 1 à 4 chiffres, éventuellement suivie de bis, ter ou d'une lettre ;
— la ville est extraite selon sa position relative au code postal ou déduite de celui-ci (ex. 75 $\rightarrow$ Paris, 69 $\rightarrow$ Lyon) ;
— le nom de rue est obtenu après suppression des autres éléments, puis les abréviations de type de voie (av, bd, r, pl, etc.) sont remplacées par leur forme complète.

3.1.3 Gestion des cas particuliers

Des règles de substitution complètent les valeurs manquantes (ex. déduction du code postal à partir de la ville) et garantissent une structure d'adresse uniforme.

3.2 Stratégie de Jointure

Toutes les jointures sont **INNER** (abandon des enregistrements non correspondants) :
— Population $\bowtie$ CSP : Suppression des codes CSP invalides
— Population $\bowtie$ Consommation : Suppression des adresses non correspondantes
— Consommation $\bowtie$ IRIS : Suppression des adresses hors zones IRIS

4 Plan d'Évaluation de la Qualité des Données

4.1 Dimensions de Qualité Considérées

Les contrôles de qualité portent sur **4 dimensions principales** issues des besoins métiers et techniques du projet :

Dimension	Nombre de Métriques	Préfixe ID	Priorité
Complétude	12	C001–C012	Obligatoire
Cohérence Syntaxique	8	CS001–CS008	Obligatoire
Granularité	1	G001	Obligatoire
Doublons	1	D001	Souhaitable

Table 1 – Dimensions de Qualité

4.2 Complétude (C001-C012)

Exemple : C006 - conso_kwh_non_null

Cette métrique vérifie le taux de présence des valeurs de consommation électrique dans la table Consommation. Elle calcule le pourcentage de lignes où `NB_KW_Jour` n'est pas null.

Cas d'usage : Si cette métrique tombe à 45%, cela signifie que 55% des enregistrements n'ont pas de valeur de consommation. Le problème peut venir d'une défaillance du système de comptage ou d'un bug dans l'extraction des données depuis la source. Cette métrique permet d'alerter rapidement l'équipe data pour corriger le pipeline avant que les tables cibles ne soient impactées.

4.3 Cohérence Syntaxique (CS001-CS008)

Exemple : CS003 - cp_geo_valide_paris

Cette métrique vérifie que les codes postaux de la table Consommation appartiennent bien à la plage Paris (75001-75020). Elle retourne le pourcentage de codes postaux valides pour Paris.

Cas d'usage : Si cette métrique descend à 60%, cela indique qu'environ 40% des codes postaux sont hors plage parisienne. Cela peut révéler une contamination des données par d'autres sources géographiques ou une erreur de mapping lors de l'intégration. Cette détection permet d'éviter des jointures incorrectes avec la table IRIS de référence.

4.4 Granularité (G001)

Exemple : G001 - echelle_kwh_s1_s2

Cette métrique compare l'échelle des consommations moyennes entre deux tables sources `Consommation1` et `Consommation2`. Elle vérifie que le ratio des moyennes est entre 0.1 et 10, retournant TRUE ou FALSE.

Cas d'usage : Si le ratio sort de cette plage (par exemple 1000), cela signifie qu'une des sources utilise probablement des kWh alors que l'autre utilise des Wh ou MWh. Cette

détection précoce évite d'agréger des données à des échelles incompatibles dans les tables cibles, ce qui fausserait complètement les analyses de consommation par IRIS ou CSP.

4.5 Doublons (D001)

Exemple : D001 - conso_uni_adresse

Cette métrique détecte les adresses dupliquées dans la table Consommation en calculant le pourcentage de doublons sur la combinaison (N, Nom_Rue, Code_Postal).

Cas d'usage : Si cette métrique indique 15%, cela signifie que 15% des enregistrements ont la même adresse qu'un autre enregistrement. Cela peut être légitime (plusieurs compteurs à la même adresse) ou problématique (réingestion accidentelle des mêmes données). Cette métrique permet d'investiguer et de décider si un dédoublonnage est nécessaire avant l'agrégation par IRIS.

4.6 Liste des métriques

Cette section détaille l'ensemble des métriques de qualité définies pour le projet. Chaque métrique est présentée avec sa description, son contexte d'application et son implémentation technique.

4.6.1 Métriques de Complétude (C001-C014)

C001 - pop_adresse_non_null
- **Table :** Population | **Colonne :** Adresse | **Phase :** Source
- **Description :** Pourcentage de lignes ayant une valeur d'adresse non nulle
- **Implémentation :**

```
SELECT (COUNT(Adresse) * 100.0 / COUNT(*)) AS taux_completude
FROM Population
WHERE Adresse IS NOT NULL;
```

C002 - pop_csp_non_null
- **Table :** Population | **Colonne :** CSP | **Phase :** Source
- **Description :** Pourcentage de lignes ayant une catégorie socio-professionnelle renseignée
- **Implémentation :**

```
SELECT (COUNT(CSP) * 100.0 / COUNT(*)) AS taux_completude
FROM Population
WHERE CSP IS NOT NULL;
```

C003 - conso_num_rue_non_null
- **Table :** Consommation | **Colonne :** N | **Phase :** Source
- **Description :** Pourcentage de lignes ayant un numéro de rue renseigné
- **Implémentation :**

```
SELECT (COUNT(N) * 100.0 / COUNT(*)) AS taux_completude
FROM Consommation
WHERE N IS NOT NULL;
```

C004 - conso_nom_rue_non_null
- **Table :** Consommation | **Colonne :** Nom_Rue | **Phase :** Source
- **Description :** Pourcentage de lignes ayant un nom de rue renseigné
- **Implémentation :**

```
SELECT (COUNT(Nom_Rue) * 100.0 / COUNT(*)) AS taux_completude
FROM Consommation
WHERE Nom_Rue IS NOT NULL;
```

C005 - conso_cp_non_null
- **Table :** Consommation | **Colonne :** Code_Postal | **Phase :** Source
- **Description :** Pourcentage de lignes ayant un code postal renseigné
- **Implémentation :**

```
SELECT (COUNT(Code_Postal) * 100.0 / COUNT(*)) AS
   taux_completude
FROM Consommation
WHERE Code_Postal IS NOT NULL;
```

C006 - conso_kwh_non_null

- **Table :** Consommation | **Colonne :** NB_KW_Jour | **Phase :** Source
- **Description :** Pourcentage de lignes ayant une valeur de consommation électrique renseignée
- **Implémentation :**

```
SELECT (COUNT(NB_KW_Jour) * 100.0 / COUNT(*)) AS
   taux_completude
FROM Consommation
WHERE NB_KW_Jour IS NOT NULL;
```

C007 - csp_ref_id

- **Table :** CSP | **Colonne :** ID_CSP | **Phase :** Source
- **Description :** Pourcentage de lignes complètes dans la table de référence CSP
- **Implémentation :**

```
SELECT (COUNT(*) * 100.0 /
    (SELECT COUNT(*) FROM CSP)) AS taux_completude
FROM CSP
WHERE ID_CSP IS NOT NULL
  AND Desc IS NOT NULL
  AND Salaire_Moyen IS NOT NULL;
```

C008 - csp_ref_salaire_moyen

- **Table :** CSP | **Colonne :** Salaire_Moyen | **Phase :** Source
- **Description :** Pourcentage de lignes ayant un salaire moyen renseigné dans la table CSP
- **Implémentation :**

```
SELECT (COUNT(Salaire_Moyen) * 100.0 / COUNT(*)) AS
   taux_completude
FROM CSP
WHERE Salaire_Moyen IS NOT NULL;
```

C009 - csp_ref_desc

- **Table :** CSP | **Colonne :** Desc | **Phase :** Source
- **Description :** Pourcentage de lignes ayant une description CSP renseignée
- **Implémentation :**

```
SELECT (COUNT(Desc) * 100.0 / COUNT(*)) AS taux_completude
FROM CSP
WHERE Desc IS NOT NULL;
```

C010 - iris_ref_id_rue

— **Table :** IRIS | **Colonne :** ID_Rue | **Phase :** Source

— **Description :** Pourcentage de lignes ayant un identifiant de rue renseigné dans IRIS

— **Implémentation :**

```
SELECT (COUNT(ID_Rue) * 100.0 / COUNT(*)) AS taux_completude
FROM IRIS
WHERE ID_Rue IS NOT NULL;
```

C011 - iris_ref_id_ville

— **Table :** IRIS | **Colonne :** ID_Ville | **Phase :** Source

— **Description :** Pourcentage de lignes ayant un identifiant de ville renseigné dans IRIS

— **Implémentation :**

```
SELECT (COUNT(ID_Ville) * 100.0 / COUNT(*)) AS
   taux_completude
FROM IRIS
WHERE ID_Ville IS NOT NULL;
```

C012 - iris_ref_iris

— **Table :** IRIS | **Colonne :** ID_Iris | **Phase :** Source

— **Description :** Pourcentage de lignes ayant un identifiant IRIS renseigné

— **Implémentation :**

```
SELECT (COUNT(ID_Iris) * 100.0 / COUNT(*)) AS taux_completude
FROM IRIS
WHERE ID_Iris IS NOT NULL;
```

C013 - target_iris_complet

— **Table :** Consommation_IRIS | **Colonnes :** ID_IRIS, Conso_moyenne_annuelle | **Phase :** Cible

— **Description :** Pourcentage de lignes complètes dans la table cible IRIS

— **Implémentation :**

```
SELECT (COUNT(*) * 100.0 /
    (SELECT COUNT(*) FROM Consommation_IRIS)) AS
       taux_completude
FROM Consommation_IRIS
WHERE ID_IRIS IS NOT NULL
  AND Conso_moyenne_annuelle IS NOT NULL;
```

C014 - target_csp_complet

— **Table :** Consommation_CSP | **Colonnes :** ID_CSP, Conso_moyenne_annuelle, Salaire_Moyen | **Phase :** Cible

— **Description :** Pourcentage de lignes complètes dans la table cible CSP

— **Implémentation :**

```
SELECT (COUNT(*) * 100.0 /
    (SELECT COUNT(*) FROM Consommation_CSP)) AS
       taux_completude
FROM Consommation_CSP
WHERE ID_CSP IS NOT NULL
  AND Conso_moyenne_annuelle IS NOT NULL
  AND Salaire_Moyen IS NOT NULL;
```

4.6.2 Métriques de Cohérence Syntaxique (CS001-CS009)

CS001 - conso_num_rue_positif

- **Table :** Consommation | **Colonne :** N | **Phase :** Source
- **Description :** Pourcentage de numéros de rue valides (entiers positifs)
- **Implémentation :**

```
SELECT (COUNT(*) * 100.0 /
    (SELECT COUNT(*) FROM Consommation)) AS taux_validite
FROM Consommation
WHERE N IS NOT NULL
  AND CAST(N AS INTEGER) > 0;
```

CS002 - conso_kwh_positif

- **Table :** Consommation | **Colonne :** NB_KW_Jour | **Phase :** Source
- **Description :** Pourcentage de valeurs de consommation positives ou nulles
- **Implémentation :**

```
SELECT (COUNT(*) * 100.0 /
    (SELECT COUNT(*) FROM Consommation)) AS taux_validite
FROM Consommation
WHERE NB_KW_Jour IS NOT NULL
  AND NB_KW_Jour >= 0;
```

CS003 - cp_geo_valide_paris

- **Table :** Consommation | **Colonne :** Code_Postal | **Phase :** Source
- **Description :** Pourcentage de codes postaux dans la plage valide pour Paris (75001-75020)
- **Implémentation :**

```
SELECT (COUNT(*) * 100.0 /
    (SELECT COUNT(*) FROM Consommation)) AS taux_validite
FROM Consommation
WHERE Code_Postal BETWEEN '75001' AND '75020';
```

CS004 - cp_geo_valide_evry

- **Table :** Consommation | **Colonne :** Code_Postal | **Phase :** Source
- **Description :** Pourcentage de codes postaux dans la plage valide pour Évry (91000-91099)
- **Implémentation :**

```
SELECT (COUNT(*) * 100.0 /
    (SELECT COUNT(*) FROM Consommation)) AS taux_validite
FROM Consommation
WHERE Code_Postal BETWEEN '91000' AND '91099';
```

CS005 - iris_rue_normalisee

- **Table :** IRIS | **Colonne :** ID_Rue | **Phase :** Source
- **Description :** Pourcentage de noms de rue correctement normalisés (minuscules, sans espaces superflus)
- **Implémentation :**

```
SELECT (COUNT(*) * 100.0 /
    (SELECT COUNT(*) FROM IRIS)) AS taux_validite
FROM IRIS
WHERE ID_Rue = LOWER(TRIM(ID_Rue));
```

CS006 - pop_csp_domaine1

- **Table :** Population (Paris) | **Colonne :** CSP | **Phase :** Source
- **Description :** Pourcentage de valeurs CSP correspondant aux descriptions de la table de référence
- **Implémentation :**

```
SELECT (COUNT(*) * 100.0 /
    (SELECT COUNT(*) FROM Population1)) AS taux_validite
FROM Population1
WHERE CSP IN (SELECT Desc FROM CSP);
```

CS007 - pop_csp_domaine2

- **Table :** Population (Évry) | **Colonne :** CSP | **Phase :** Source
- **Description :** Pourcentage de valeurs CSP dans le domaine numérique valide (1-6)
- **Implémentation :**

```
SELECT (COUNT(*) * 100.0 /
    (SELECT COUNT(*) FROM Population2)) AS taux_validite
FROM Population2
WHERE CSP IN (1, 2, 3, 4, 5, 6);
```

CS008 - adresse_format_standard_paris

- **Table :** Population (Paris) | **Colonne :** Adresse | **Phase :** Source
- **Description :** Pourcentage d'adresses respectant le format standard Paris (Ville, N Nom_Rue)
- **Implémentation :**

```
SELECT (COUNT(*) * 100.0 /
    (SELECT COUNT(*) FROM Population1)) AS taux_validite
FROM Population1
WHERE Adresse LIKE 'Paris, % %';
```

CS009 - adresse_format_standard_evry

— **Table :** Population (Évry) | **Colonne :** Adresse | **Phase :** Source

— **Description :** Pourcentage d'adresses respectant le format standard Évry (Ville, Code_postal, N Nom_Rue)

— **Implémentation :**

```
SELECT (COUNT(*) * 100.0 /
    (SELECT COUNT(*) FROM Population2)) AS taux_validite
FROM Population2
WHERE Adresse LIKE 'Evry, 91%, %%';
```

4.6.3 Métrique de Granularité (G001)

G001 - echelle_kwh_s1_s2

— **Tables :** Consommation1, Consommation2 | **Colonne :** NB_KW_Jour | **Phase :** Source

— **Description :** Vérifie que le ratio des consommations moyennes entre les deux sources est dans une plage acceptable (0.1 à 10), indiquant une cohérence d'échelle

— **Implémentation :**

```
SELECT
    CASE
        WHEN (AVG(s1.NB_KW_Jour) / AVG(s2.NB_KW_Jour))
           BETWEEN 0.1 AND 10
        THEN 'TRUE'
        ELSE 'FALSE'
    END AS echelle_coherente
FROM Consommation1 s1, Consommation2 s2;
```

4.6.4 Métrique de Doublons (D001)

D001 - conso_uni_adresse

— **Table :** Consommation | **Colonnes :** N, Nom_Rue, Code_Postal | **Phase :** Intermédiaire

— **Description :** Pourcentage d'adresses dupliquées dans la table Consommation

— **Implémentation :**

```
SELECT
    ((COUNT(*) - COUNT(DISTINCT N, Nom_Rue, Code_Postal)) *
       100.0
    / COUNT(*)) AS taux_doublons
FROM Consommation;
```

4.7 Justification de la Granularité des Métriques

Les métriques de qualité sont volontairement granulaires, c'est-à-dire définies au niveau colonne plutôt qu'au niveau table global. Cette approche permet d'identifier précisément la source des problèmes de qualité et d'accélérer leur résolution.

Exemple concret avec la complétude

Au lieu d'une seule métrique globale pour la table Consommation, on a défini 4 métriques séparées :

— C003 : `conso_num_rue_non_null` (colonne N)
— C004 : `conso_nom_rue_non_null` (colonne Nom_Rue)
— C005 : `conso_cp_non_null` (colonne Code_Postal)
— C006 : `conso_kwh_non_null` (colonne NB_KW_Jour)

Si on avait une métrique globale indiquant "Consommation : 75% de complétude", on saurait qu'il y a un problème mais pas où. Avec les métriques granulaires, on obtient par exemple :

— C003 (N) : 98%
— C004 (Nom_Rue) : 95%
— C005 (Code_Postal) : 92%
— C006 (NB_KW_Jour) : 45%

Le problème est immédiatement localisé sur `NB_KW_Jour`. On peut alors investiguer directement la source de cette colonne sans perdre de temps à analyser les autres.

Bénéfices opérationnels

1. **Diagnostic rapide :** Identification immédiate de la colonne problématique
2. **Priorisation :** Les colonnes critiques (comme Code_Postal pour les jointures avec IRIS) peuvent avoir des seuils d'alerte plus stricts que les colonnes optionnelles
3. **Traçabilité :** Quand une ingestion échoue partiellement, on voit exactement quelle partie du processus ETL est impactée
4. **Monitoring ciblé :** Suivi de l'évolution de chaque colonne dans le temps pour détecter les dégradations progressives

Cette logique s'applique aussi aux autres dimensions. Pour la cohérence syntaxique, on a par exemple CS003 et CS004 qui vérifient les codes postaux Paris et Évry séparément au lieu d'une validation générique, permettant d'identifier si le problème vient d'une source géographique spécifique.

5 Résultats des Métriques de Qualité

5.1 Vue d'ensemble

Dimension	Total Métriques	Résultat Global
Complétude	12	Acceptable (83-100%)
Cohérence Syntaxique	8	Problématique (25-100%)
Granularité	1	Échec (FALSE)
Doublons	1	OK (0%)

TABLE 2 – Vue d'ensemble des résultats

5.2 Complétude - Qualité Acceptable

Les données sources sont globalement complètes avec 10 métriques au-dessus de 90%. Deux points d'attention :

- **C007 et C008 (83,33%) :** Tables CSP de référence incomplètes sur `ID_CSP` et `Salaire_Moyen`. Environ 17% des lignes manquent ces informations, ce qui peut limiter les jointures et analyses par catégorie socioprofessionnelle.
- **C006 (80%) :** 20% des enregistrements Consommation n'ont pas de valeur `NB_KW_Jour`. Problème modéré mais acceptable si ces lignes correspondent à des compteurs inactifs.

Les colonnes critiques pour les jointures (Code_Postal, adresses, identifiants IRIS) sont toutes à 100%.

5.3 Cohérence Syntaxique - Problèmes Majeurs Détectés

Codes postaux hors normes

- **CS003 (48%) :** Seulement 48% des codes postaux sont dans la plage Paris (75001-75020)
- **CS004 (36%) :** Seulement 36% sont dans la plage Évry (91000-91099)

Ces résultats indiquent une contamination importante des données par d'autres zones géographiques ou des erreurs de saisie. Plus de la moitié des données ne correspondent pas aux périmètres géographiques attendus. Impact direct sur la fiabilité des agrégations par IRIS.

CSP non conformes

- **CS005 (25%) :** Seulement 25% des CSP de Population_Paris correspondent aux valeurs de la table de référence

75% des catégories socioprofessionnelles ne matchent pas avec le référentiel. Cela rend impossible une jointure fiable entre Population et CSP de référence pour récupérer les salaires moyens.

Formats d'adresse

- **CS008 (58,33%) :** Format Évry non standard sur 42% des adresses

Validations positives

— CS002 (80%) : Consommations positives correctes
— CS006 (100%) : CSP Population_Évry conformes au domaine numérique
— CS007 (100%) : Format Paris respecté

5.4 Granularité - Échec Critique

G001 (FALSE) : Le ratio des consommations moyennes entre Consommation_Paris et Consommation_Évry est hors de la plage [0.1, 10].

Les deux sources utilisent des unités ou échelles différentes (probablement Wh vs kWh ou kWh vs MWh). Les données ne peuvent pas être agrégées directement sans transformation. Il faut identifier quelle source utilise quelle unité et appliquer un facteur de conversion avant toute consolidation.

5.5 Doublons - Aucun Problème

D001 (0%) : Aucun doublon détecté sur les adresses dans Consommation.

Chaque combinaison (N, Nom_Rue, Code_Postal) est unique. Les données sont propres de ce point de vue.

5.6 Recommandations Prioritaires

1. **Urgent :** Corriger le problème d'échelle des consommations (G001) avant toute agrégation
2. **Important :** Investiguer la source des codes postaux hors périmètre (CS003, CS004) - possible mélange de datasets
3. **Important :** Vérifier le mapping CSP entre Population1 et la table de référence (CS005)
4. **Souhaitable :** Compléter les tables de référence CSP (C007, C008)

6 Amélioration

6.1 Amélioration de la complétude

La complétude des données concerne les valeurs manquantes (null). Malheureusement, il n'est pas possible d'améliorer cette dimension sans disposer de sources externes ou de règles de déduction fiables. En d'autres termes, on ne peut pas créer ou deviner des valeurs à partir de données inexistantes.

6.2 Amélioration de la cohérence syntaxique

Concernant la cohérence syntaxique, certaines anomalies ont été observées dans les tables de consommation, notamment au niveau des codes postaux (erreurs de saisie, caractères parasites, zéros en trop, etc.).

Pour corriger ces problèmes, on peut appliquer un nettoyage automatique à l'aide d'expressions régulières : elles permettraient de repérer les schémas de type 75— ou 91— et de supprimer les caractères indésirables ou les zéros surnuméraires, qu'ils se trouvent en début, au milieu ou à la fin de la chaîne.

Dans la table Population_Paris, les valeurs de la variable CSP ne correspondent pas toujours aux intitulés de la table de référence CSP. Ces incohérences peuvent provenir de variations de genre (masculin/féminin), de nombre (singulier/pluriel) ou de fautes de frappe.

Pour corriger cela, nous avons mis en place une détection par racine de mots : par exemple, si une valeur contient les chaînes `commerç` ou `arstisan`, elle sera associée à la catégorie « Artisans, commerçants et chefs d'entreprise » de la table CSP.

6.3 Amélioration de la granularité

Après analyse, nous avons constaté un facteur d'échelle de 1 000 entre deux sources : l'une exprimait les valeurs en W/h, l'autre en kW/h.

Pour harmoniser la granularité, une simple division par 1 000 des valeurs exprimées en W/h permet d'obtenir une cohérence entre les deux tables Consommation.