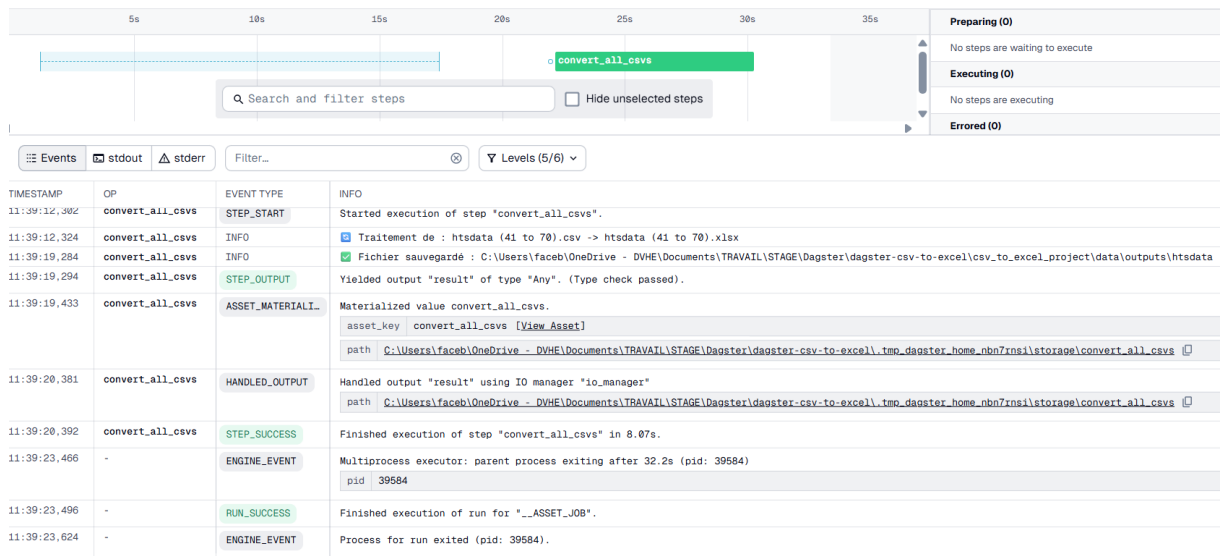


SYNTHÈSE

Lorsque le programme reconnaît **un ou plusieurs** fichiers .csv :



```
# On loggue un message pour dire quel fichier on est en train de traiter
context.log.info(f"📄 Traitement de : {filename} -> {new_filename}")

# Lecture du CSV
df = pd.read_csv(input_path)

# Écriture en Excel (garder le même nom de base)
df.to_excel(output_path, index=False)

# On ajoute le chemin à notre liste de succès
processed_files.append(output_path)

context.log.info(f"✅ Fichier sauvegardé : {output_path}")
```

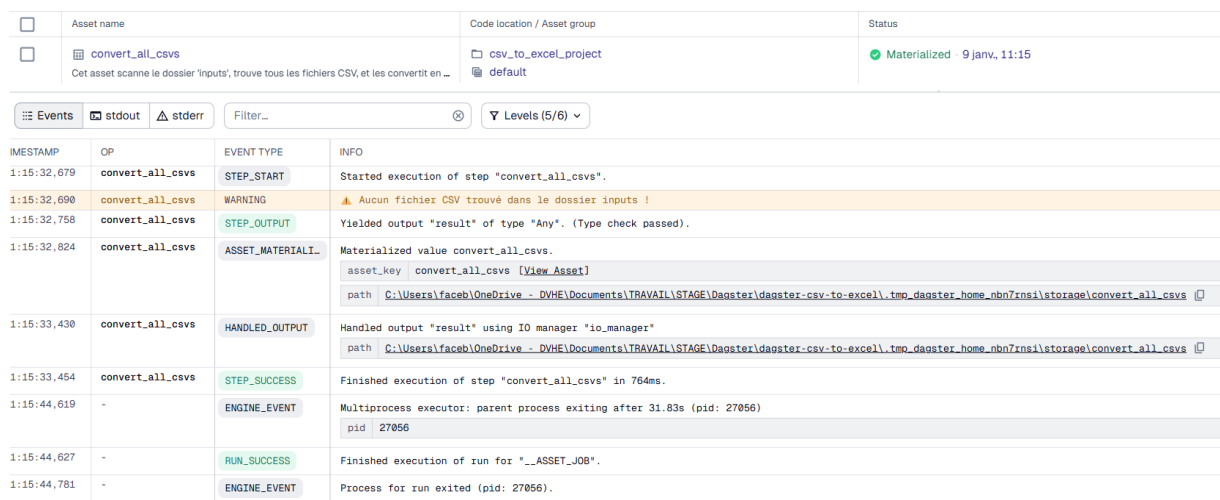
Un bout du code `assets.py`

Lorsque le programme ne reconnaît pas **un ou plusieurs** fichiers .csv :

```
# Si la liste est vide, on prévient qu'on n'a rien trouvé
if not processed_files:
    context.log.warning("⚠️ Aucun fichier CSV trouvé dans le dossier inputs !")
```

- Pourquoi **Materialized** ?

Cela montre tout simplement que le programme est arrivé à terme. Dans le code, le programme se finit qu'il ait trouvé un fichier .csv à convertir ou non. Le programme n'a pas été fait pour tourner en boucle jusqu'à trouver un fichier à convertir.



Lorsque le programme n'arrive pas à lire un fichier .csv mais qu'il en lit un autre en même temps :

- [Pourquoi ?](#)

Il n'y avait pas de try et except dans mon `assets.py` ce qui veut dire qu'un simple fichier corrompu pouvait faire crasher l'analyse. Dans `bad_data.csv`, Pandas a lu l'en-tête (ligne 1) et a vu 4 colonnes. Arrivé à la ligne 4, il a trouvé 9 morceaux séparés par des virgules. Il ne sait pas faire rentrer 9 valeurs dans 4 cases ce qui crash l'analyse.

Sans try/except :

14:15:17,346	Since start of run	16,286		Traitement de : bad_data.csv -> bad_data.xlsx
14:15:17,651	Since start of step	0,023	FAILURE	<code>dagster.core.errors.DagsterExecutionStepExecutionError: Error occurred while executing op "convert_all_csvs":</code> The above exception was caused by the following exception: <code>pandas.errors.ParserError: Error tokenizing data. C error: Expected 4 fields in line 4, saw 9</code> Stack Trace: File "C:\Users\faceb\AppData\Local\Programs\Python\Python313\Lib\site-packages\dagster_core\execution\plan\utils.py", line 57, in <code>op_execution_error_boundary</code> yield File "C:\Users\faceb\AppData\Local\Programs\Python\Python313\Lib\site-packages\dagster_core\utils_init_.py", line 394, in <code>iterate_with_context</code> next_output = next(iterator) View full message
14:15:24,722	-		ENGINE_EVENT	Multiprocess executor: parent process exiting after 23.36s (pid: 8292) pid 8292
14:15:24,755	-		RUN_FAILURE	Execution of run for "_ASSET_JOB" failed. Steps failed: ['convert_all_csvs'].
14:15:25,006	-		ENGINE_EVENT	Process for run exited (pid: 8292).

Avec :

```
# >>> DEBUT DE LA ZONE PROTÉGÉE <<<
try:
    # 1. On essaie de lire le CSV
    # C'est ici que 'bad_data.csv' va déclencher une alerte, mais pas un crash
    df = pd.read_csv(input_path)

    # Petite vérification supplémentaire si le fichier est vide
    if df.empty:
        raise ValueError("Le fichier est vide")

    # 2. On écrit le fichier Excel
    df.to_excel(output_path, index=False)

    # 3. Si on arrive ici, c'est que tout s'est bien passé
    context.log.info(f"✅ Succès : {new_filename}")
    processed_files.append(output_path)

except Exception as e:
    # >>> ZONE DE GESTION D'ERREUR <<<
    # Si n'importe quoi se passe mal au-dessus, on atterrit ici.
    error_message = f"❌ ÉCHEC sur {filename}. Raison : {str(e)}"

    # On note l'erreur en rouge dans les logs
    context.log.error(error_message)

    # On ajoute le fichier à la liste des échecs pour le bilan
    failed_files.append(filename)

# >>> FIN DE LA ZONE PROTÉGÉE <<<
```

EVENT TYPE	INFO
RESOURCE_AVAILABLE	Finalizing initialization of resources [io_manager].
LOGS_CAPTURED	Started capturing logs in process (pid: 34956). step_keys ["convert_all_csvs"] captured_logs View stdout / stderr
CSVS	STEP_START Started execution of step "convert_all_csvs".
CSVS	INFO Tentative de traitement : bad_data.csv
CSVS	ERROR ❌ ÉCHEC sur bad_data.csv. Raison : Error tokenizing data. C error: Expected 4 fields in line 4, saw 9
CSVS	INFO Tentative de traitement : htdata (41 to 70).csv
CSVS	INFO ✅ Succès : htdata (41 to 70).xlsx
CSVS	WARNING ⚠️ Terminé avec des erreurs. Fichiers échoués (1) : ['bad_data.csv']
CSVS	STEP_OUTPUT Yielded output "result" of type "Any". (Type check passed).
CSVS	ASSET_MATERIALIZED Materialized value convert_all_csvs. asset_key convert_all_csvs [View Asset] path C:\Users\faceb\OneDrive - DYHE\Documents\TRAVAIL\STAGE\Danster\danster.csv-to-excel\tmp_danster_ht
CSVS	HANDLED_OUTPUT Handled output "result" using IO manager "io_manager" path C:\Users\faceb\OneDrive - DYHE\Documents\TRAVAIL\STAGE\Danster\danster.csv-to-excel\tmp_danster_ht
CSVS	STEP_SUCCESS Finished execution of step "convert_all_csvs" in 8.44s.
ENGINE_EVENT	Multiprocess executor: parent process exiting after 24.57s (pid: 35656) pid 35656
RUN_SUCCESS	Finished execution of run for "_ASSET_JOB".
ENGINE_EVENT	Process for run exited (pid: 35656).