

在 SQL Server Data Tools (SSDT) 的 SQL Server Integration Services (SSIS) 中，**Union All Transformation** 和 **Merge Transformation** 是資料流任務中用於合併多個資料來源的兩個常用元件，但它們的用途、功能和使用場景有顯著差異。以下是詳細比較與說明，並結合您之前的問題（例如匯入錯誤、新增 `job_id`、SQL 語句修正和 Derived Column）提供具體應用。

1. Union All Transformation

定義與用途

- **定義：** Union All 是一個 SSIS 資料流轉換元件，用於將多個輸入資料流合併成單一輸出資料流，類似於 SQL 中的 `UNION ALL` 語句。
- **主要用途：**
 - 合併來自不同來源的資料（例如多個表、檔案或資料庫），不要求輸入資料排序。
 - 快速堆疊記錄，保留所有記錄（包括重複記錄）。
 - 適用於簡單的資料整合場景，例如合併多個相似結構的資料集。

特性

- **輸入要求：**
 - 接受多個輸入（2 個或更多）。
 - 輸入資料的欄位名稱和資料類型必須相容（例如 `[dbo].[104]` 的 `info_description` 必須與另一來源的對應欄位類型一致）。
 - 不要求輸入資料排序。
- **輸出：**
 - 產生單一輸出資料流，包含所有輸入的記錄，按輸入順序堆疊。
 - 不執行去重（保留所有記錄，包括重複）。
- **效能：**
 - 高效能，因為不涉及排序或比較操作。
 - 適合處理大量資料。
- **限制：**
 - 不支援複雜的合併邏輯（例如基於條件合併或去重）。
 - 欄位映射較為手動，需確保輸入欄位對齊。

使用場景

- 合併多個來源的相似資料，例如將多個 CSV 檔案的記錄合併到 `[dbo].[104]`。
- 將不同資料庫的表（例如 `[dbo].[JOBS]` 和另一個臨時表）堆疊成單一資料集。
- 在資料倉儲場景中，快速整合多個來源的歷史資料。

範例

假設您有兩個來源表 (`Source1` 和 `Source2`)，都包含 `job_id` 和 `info_description`，需合併到 `[dbo].[104]`：

1. 在 SSIS 資料流中：
 - 新增兩個 OLE DB 來源，分別連接到 `Source1` 和 `Source2`。
 - 拖曳 **Union All Transformation**，將兩個來源連接到其輸入。
 - 設定欄位映射 (例如 `Source1.job_id` 和 `Source2.job_id` 映射到輸出 `job_id`)。
 - 將 Union All 的輸出連接到 OLE DB 目標 (`[dbo].[104]`)。
2. 執行套件，所有記錄 (包括重複) 將合併到目標表。

與您案例的關聯

- **匯入錯誤 (`info_description` 截斷)**：若您從多個來源 (例如 CSV 和資料庫) 匯入 `[dbo].[104]`，可用 Union All 合併資料流，再使用 Derived Column 截斷 `info_description` (如 `LEFT(info_description, 50)`) 以避免截斷錯誤。
- **Unicode/非 Unicode 不匹配**：在 Union All 前，使用 Derived Column 統一 `J_person` 的資料類型 (例如 `(DT_STR, 100, 1252)J_person`)，確保輸入相容。

2. Merge Transformation

定義與用途

- **定義**：Merge 是一個 SSIS 資料流轉換元件，用於合併兩個已排序的輸入資料流，基於指定的合併鍵 (類似於 SQL 的 `JOIN`)。
- **主要用途**：
 - 合併兩個結構相似的資料來源，基於鍵值 (如 `job_id`) 進行有序整合。
 - 適用於需要排序或結構化合併的場景，例如資料倉儲的增量載入。
 - 類似 SQL 的 `UNION ALL` 或 `JOIN`，但要求輸入資料預先排序。

特性

- **輸入要求**：
 - 僅接受兩個輸入 (不同於 Union All 的多輸入)。
 - 輸入資料必須預先排序 (透過 Sort Transformation 或來源查詢的 `ORDER BY`)。
 - 合併鍵 (例如 `job_id`) 的資料類型和排序規則必須一致。
- **輸出**：
 - 產生單一輸出資料流，根據合併鍵排序記錄。
 - 不執行去重 (類似 Union All)。
- **效能**：
 - 效能低於 Union All，因為需要排序操作 (若未在來源排序)。
 - 適合中小型資料集或已排序的資料。

- 限制：
 - 僅限兩個輸入，限制靈活性。
 - 必須預先排序，否則會失敗或產生不正確結果。
 - 配置較複雜，需明確指定合併鍵。

使用場景

- 合併兩個已排序的資料來源，例如將 `[dbo].[JOBS]` 的歷史資料與新資料按 `job_id` 合併。
- 在資料倉儲中，合併主表與增量更新表，確保按鍵值有序。
- 當需要基於特定欄位（例如 `job_id`）進行結構化合併，而不是簡單堆疊。

範例

假設您有兩個已排序的來源表（`Source1` 和 `Source2`），都包含 `job_id` 和 `info_description`，需按 `job_id` 合併到 `[dbo].[104]`：

1. 在 SSIS 資料流中：

- 新增兩個 OLE DB 來源，查詢分別為：

```
SELECT job_id, info_description FROM Source1 ORDER BY job_id
SELECT job_id, info_description FROM Source2 ORDER BY job_id
```

- （若來源未排序）新增 Sort Transformation，按 `job_id` 排序。
- 拖曳 **Merge Transformation**，將兩個排序後的來源連接到其輸入。
- 設定合併鍵（`job_id`），映射其他欄位（`info_description`）。
- 將 Merge 的輸出連接到 OLE DB 目標（`[dbo].[104]`）。

2. 執行套件，記錄將按 `job_id` 有序合併。

與您案例的關聯

- 刪除操作（`DELETE FROM [dbo].[JOBS] WHERE job_id > 1000`）：若您需要合併 `[dbo].[JOBS]` 的保留記錄（`job_id <= 1000`）與另一來源的新記錄，可用 Merge 按 `job_id` 有序合併。
- 新增 `job_id`：若多個來源需要合併並插入 `[dbo].[104]`，可用 Merge 確保 `job_id` 的有序性（若 `IDENTITY` 由資料庫生成，則無需在 SSIS 中處理）。

Union All 與 Merge 的主要差異

特性	Union All	Merge
輸入數量	多個（2 個或更多）	僅限 2 個
排序要求	不需要排序	必須預先排序（按合併鍵）
合併方式	簡單堆疊記錄（類似 SQL <code>UNION ALL</code> ）	基於鍵值合併（類似 SQL <code>JOIN</code> 或 <code>UNION</code> ）
去重	不去重，保留所有記錄	不去重，保留所有記錄

特性	Union All	Merge
效能	高（無排序或比較）	較低（因排序要求）
欄位映射	需手動對齊欄位，類型必須相容	需指定合併鍵，其他欄位自動映射
使用場景	快速合併多來源，無排序需求	有序合併兩個來源，需基於鍵值整合
配置複雜度	簡單，僅需映射欄位	較複雜，需排序和設定合併鍵

與您案例的具體應用

以下是如何在您的問題中使用 Union All 或 Merge：

1. 匯入錯誤（`info_description` 截斷）：

- **Union All**：若您從多個來源（例如多個 CSV 或表）匯入 `[dbo].[104]`，可用 Union All 合併資料流，再使用 Derived Column 處理 `info_description`（如 `LEFT(info_description, 50)`）。
- **Merge**：若只有兩個來源且需按 `job_id` 有序合併（例如歷史資料與新資料），使用 Merge 並確保輸入已排序。

2. Unicode/非 Unicode 不匹配：

- 無論使用 Union All 或 Merge，都需在合併前使用 Derived Column 統一 `J_person` 的資料類型：

```
(DT_STR, 100, 1252)J_person
```

- Union All 更適合多來源場景，Merge 適合兩個有序來源。

3. 刪除操作（`[dbo].[JOBS]`）：

- **Union All**：若需合併保留的記錄（`job_id <= 1000`）與其他來源的記錄，Union All 是簡單選擇。
- **Merge**：若需確保合併後的記錄按 `job_id` 有序（例如資料倉儲需求），使用 Merge。

4. 新增 `job_id`：

- 若 `job_id` 由資料庫的 `IDENTITY` 生成，Union All 和 Merge 僅需處理其他欄位的合併。
- 若 SSIS 需生成臨時 `job_id`，可用 Derived Column 配合 Union All 或 Merge。

選擇 Union All 或 Merge 的原則

- 選擇 **Union All**：
 - 有 2 個以上輸入來源。
 - 不需要排序或基於鍵值合併。
 - 追求高效能（例如處理大量資料）。
 - 範例：合併多個 CSV 檔案到 `[dbo].[104]`。
- 選擇 **Merge**：
 - 僅有 2 個輸入來源。
 - 需要按特定鍵（例如 `job_id`）有序合併。

- 資料已排序或可輕鬆排序。
 - 範例：合併 `[dbo].[JOBS]` 的歷史與增量資料。
-

最佳實務

- **Union All：**
 - 確保輸入欄位類型一致（使用 Derived Column 轉換，如 `(DT_STR, 100, 1252)J_person`）。
 - 使用資料檢視器檢查合併結果，確認無資料遺失。
- **Merge：**
 - 驗證輸入是否已排序（檢查 Sort Transformation 或來源查詢）。
 - 明確指定合併鍵（例如 `job_id`），避免不正確的合併。
- **稽核：**
 - 在 SSIS 套件中啟用記錄，追蹤 Union All 或 Merge 的執行（例如處理的行數）。
 - 使用 Schema Compare 稽核目標表結構（例如 `[dbo].[104]` 的 `info_description` 長度）。
- **效能：**
 - 優先選擇 Union All，除非需要 Merge 的有序合併。
 - 對大型資料集，測試兩者的執行時間。