
Integration Service

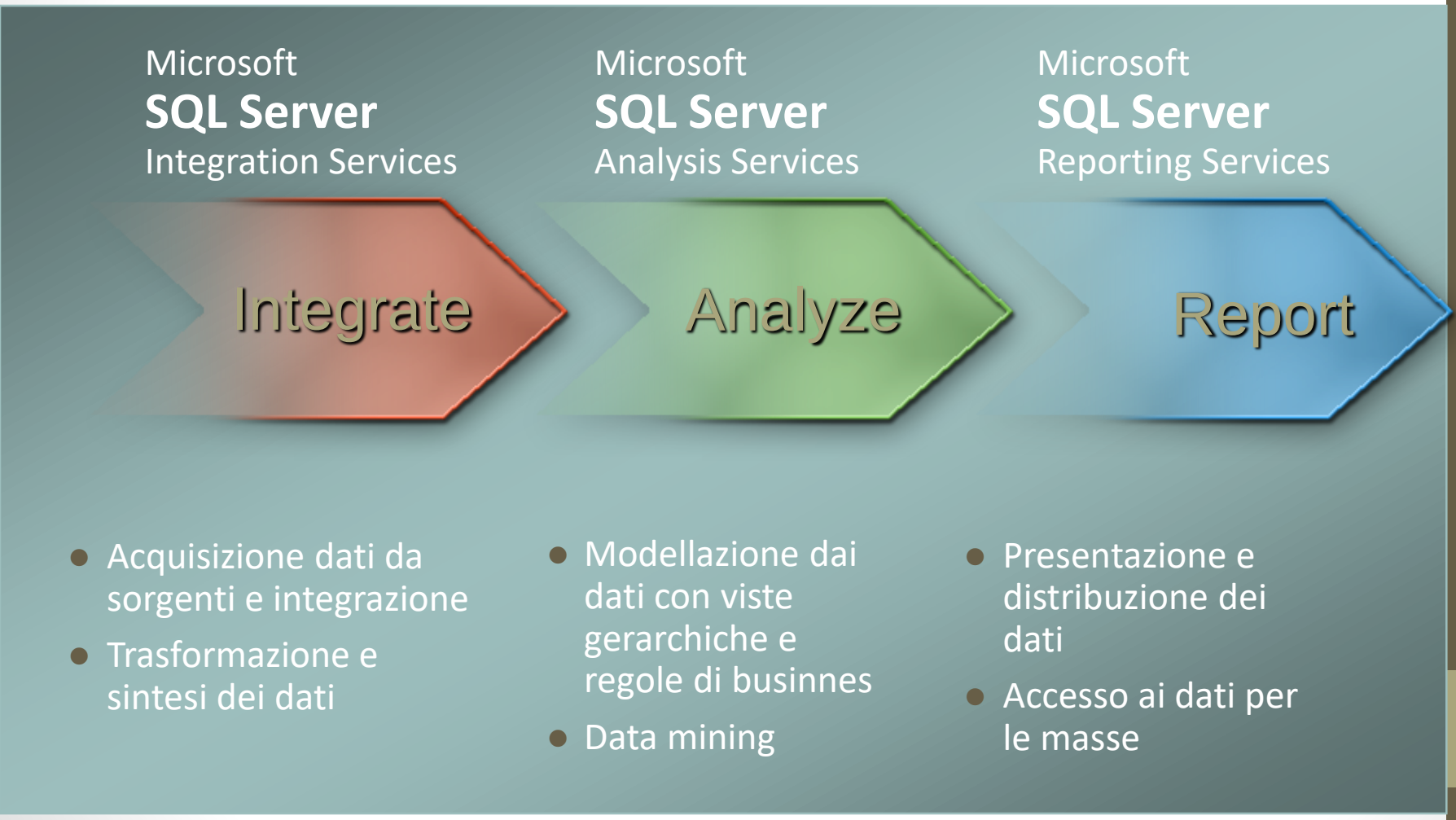
# Esercitazione SSIS

ALESSANDRA LUMINI

Alessandra.lumini@unibo.it

http://tinyurl.com/EsBI2019

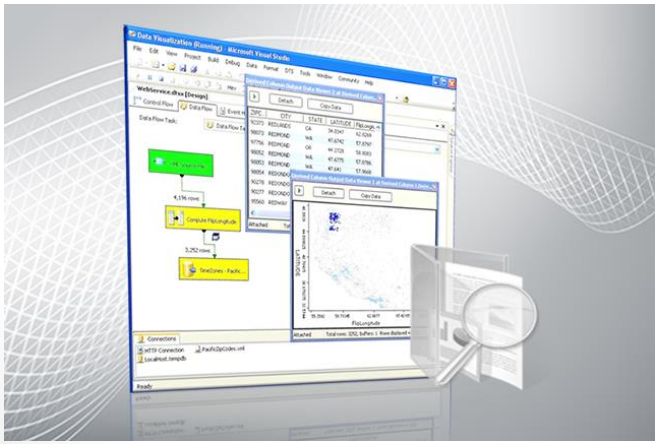
# SQL Server Business Intelligence

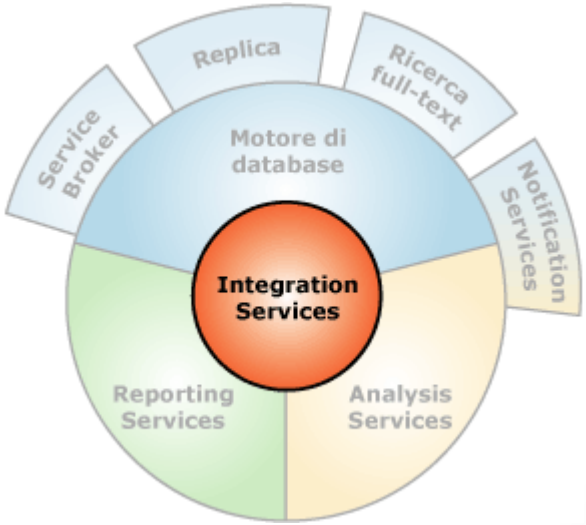


# Cos'è SSIS?

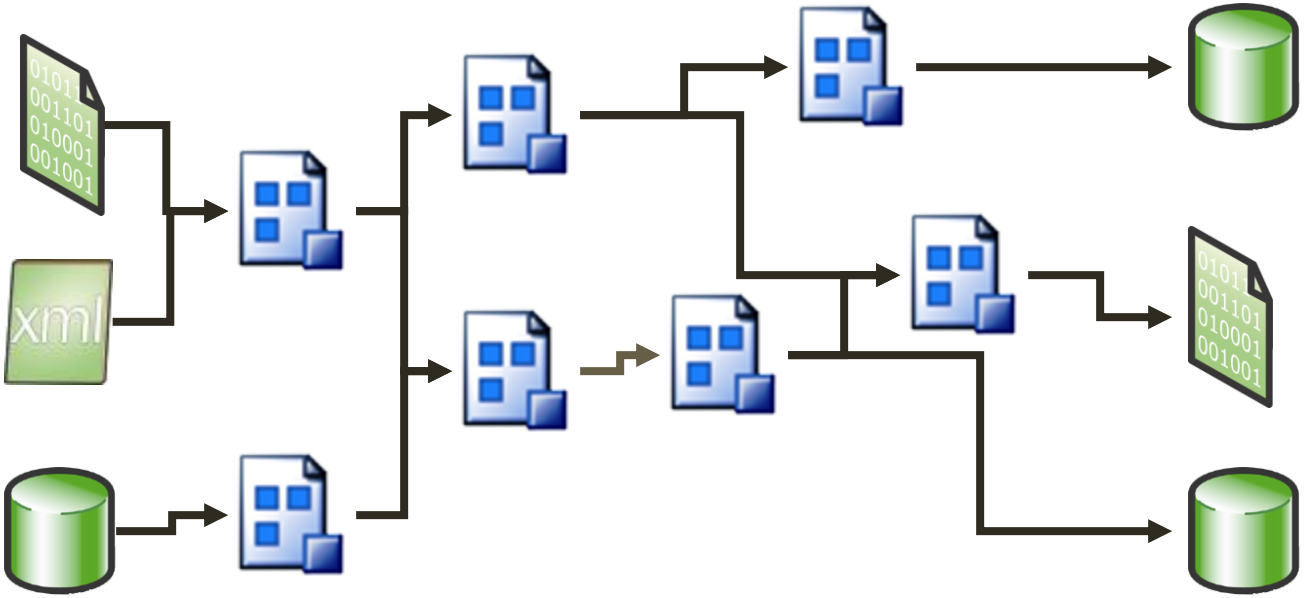
SQL Server Integration Service è  piattaforma per la creazione di soluzioni di **integrazione** di dati ad alte prestazioni che consente **l'estrazione**, la **trasformazione** e il **caricamento** di pacchetti (ETL) per il data warehousing

- Offre funzionalità per la gestione di progetti di **Master Data Management**.





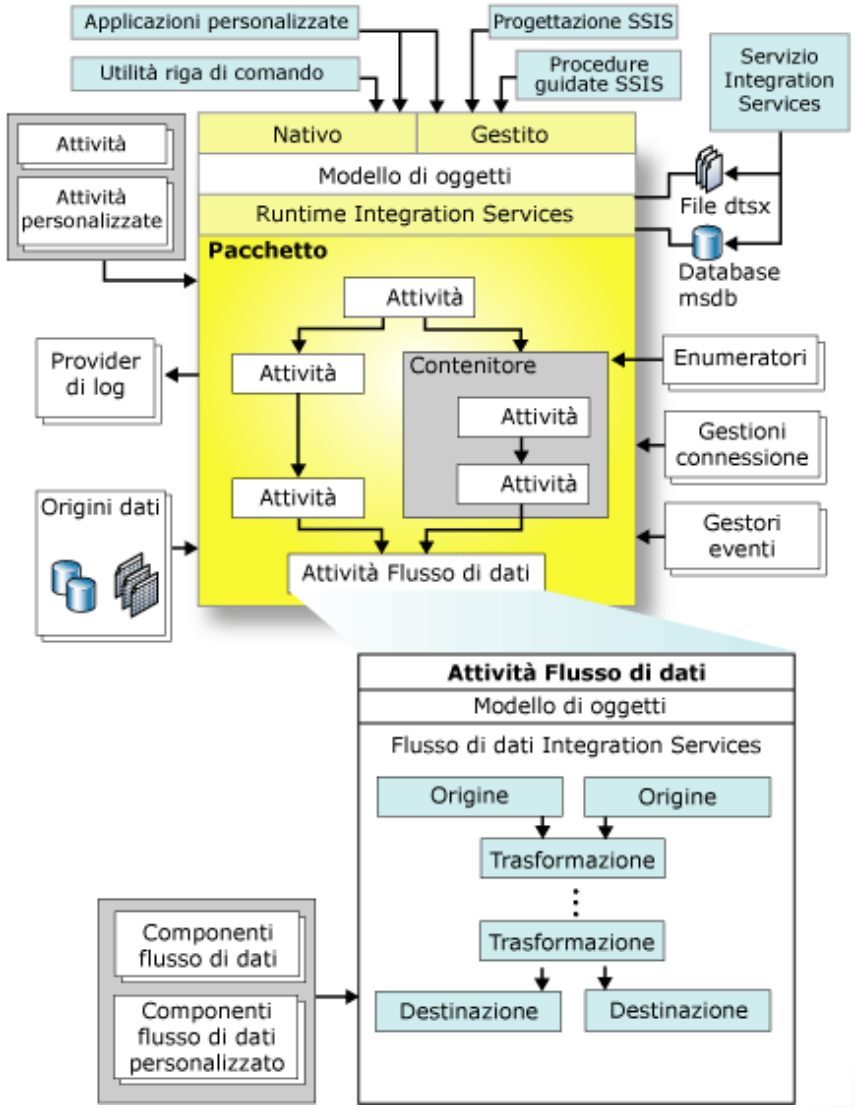
# Come funziona SSIS




- Le sorgenti possono essere eterogenee
- Le componenti di trasformazione modificano ed elaborano i dati in molti modi
- La pulizia dei dati si base su regole e condizioni di errore.
- I flussi sono complessi, ma altamente concorrenti
- I dati possono essere caricati in parallelo su diverse destinazioni.

# Architettura SSIS

- SSIS è composto da 2 motori:
  - Workflow engine
  - Data Flow engine



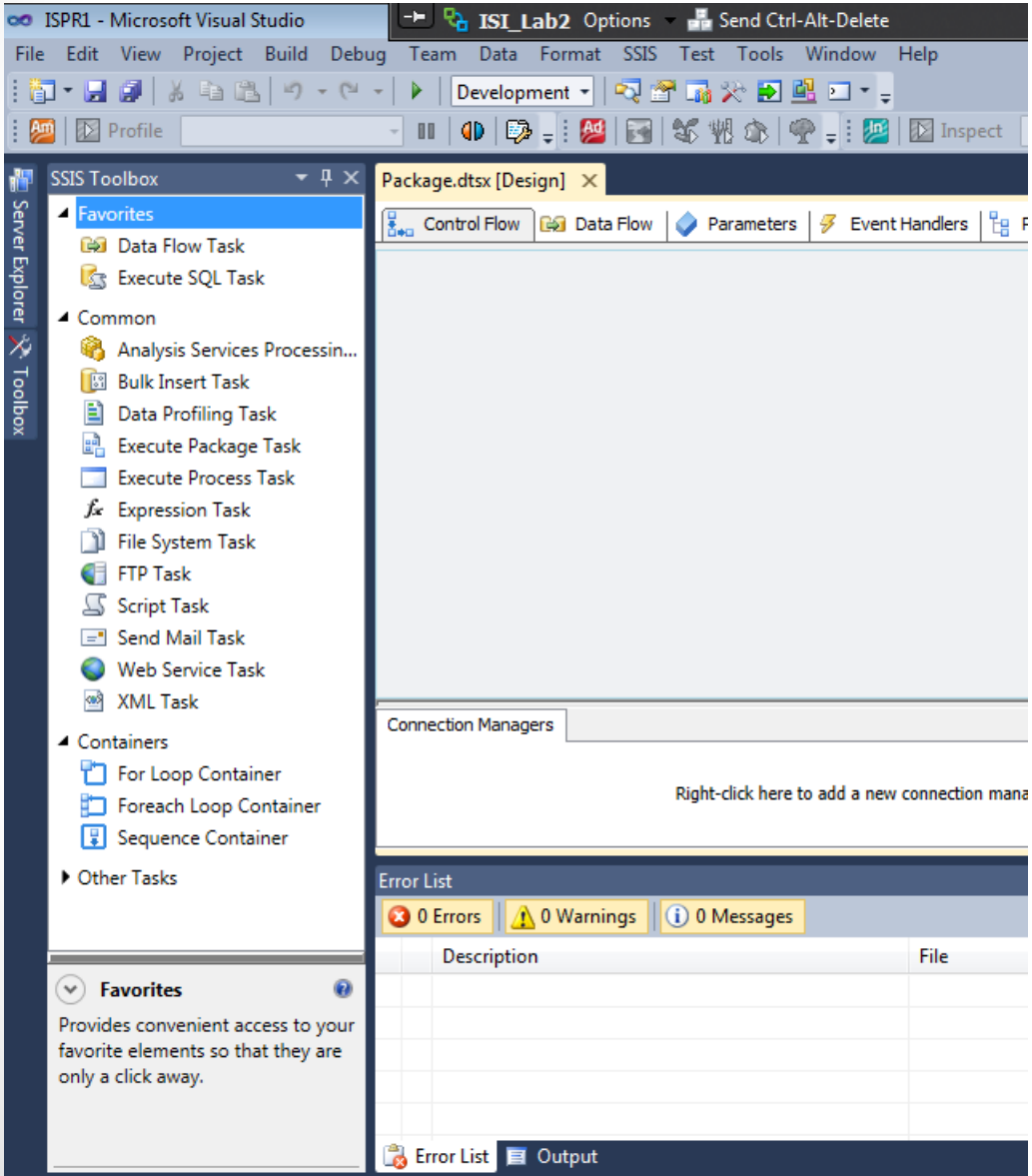
# Componenti di un pacchetto SSIS

- Flusso di controllo: insieme di componenti per il controllo del flusso dati all'interno di un pacchetto.
  - Inserimento di un controllo "Ciclo" che consenta di ripetere l'operazione di estrazione dati da molteplici sorgenti, evitando la definizione della procedura di esportazione per ogni singola sorgente.
  - Specifica di condizioni per l'esecuzione di particolari attività all'interno del pacchetto.
  - Definizione dell'ordine di esecuzione delle diverse attività che caratterizzano il pacchetto.
- Flusso dati: insieme delle origini, delle trasformazioni e delle destinazioni dati.
  - Esempi di controlli per la trasformazione dati: unione dati, raggruppamento fuzzy, suddivisione condizionale.

# Componenti di un pacchetto SSIS

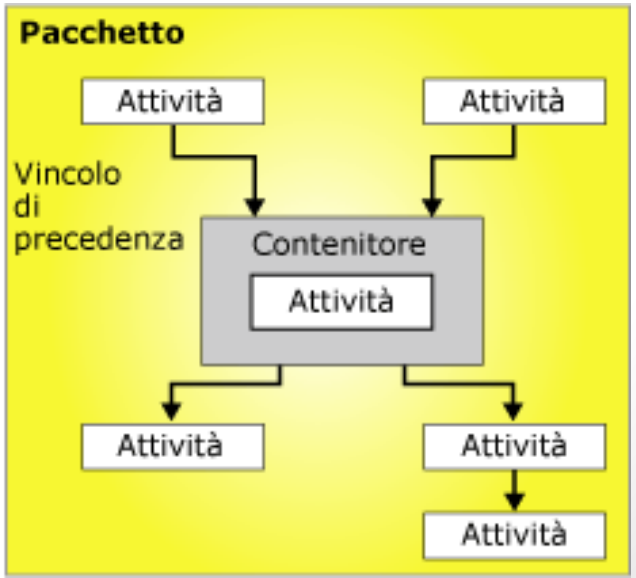
- Gestione connessioni: componente per la definizione della connessione alle sorgenti e destinazioni dati (es. Flat file, origine OLE DB).
- Variabili: utilizzate per aggiornare dinamicamente i valori di proprietà all'interno di un pacchetto o per gestire funzioni di controllo (es. variabile di ciclo).
- Gestore eventi: componente per la gestione degli eventi generati durante l'esecuzione di pacchetti SSIS.
- Provider log: gestisce informazioni di supporto (log) relative all'esecuzione di pacchetti SSIS (es. data/ora di esecuzione, elenco attività).

# Flusso di controllo



# Flusso di controllo

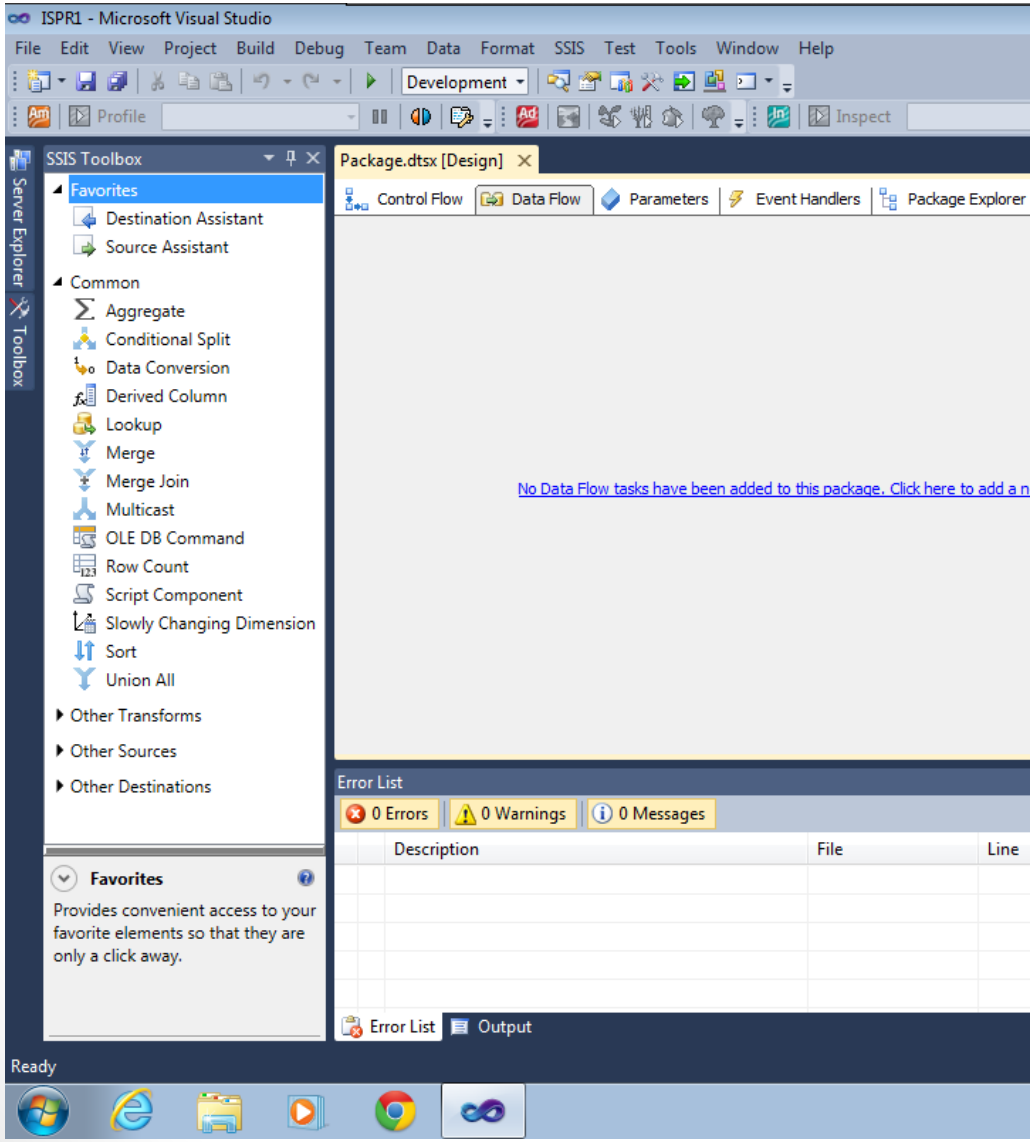
- Contenitori: definiscono la struttura del flusso di controllo, raggruppando attività e definendo operazioni cicliche:
  - Ciclo For: ripete il flusso di controllo finché un'espressione specificata non risulta falsa.
  - Ciclo Foreach: enumera un insieme di entità e ripete il flusso di controllo per ogni elemento dell'insieme.
  - Sequenza: consente di definire dei sottoinsiemi di attività e contenitori e considerarli come unità atomiche.



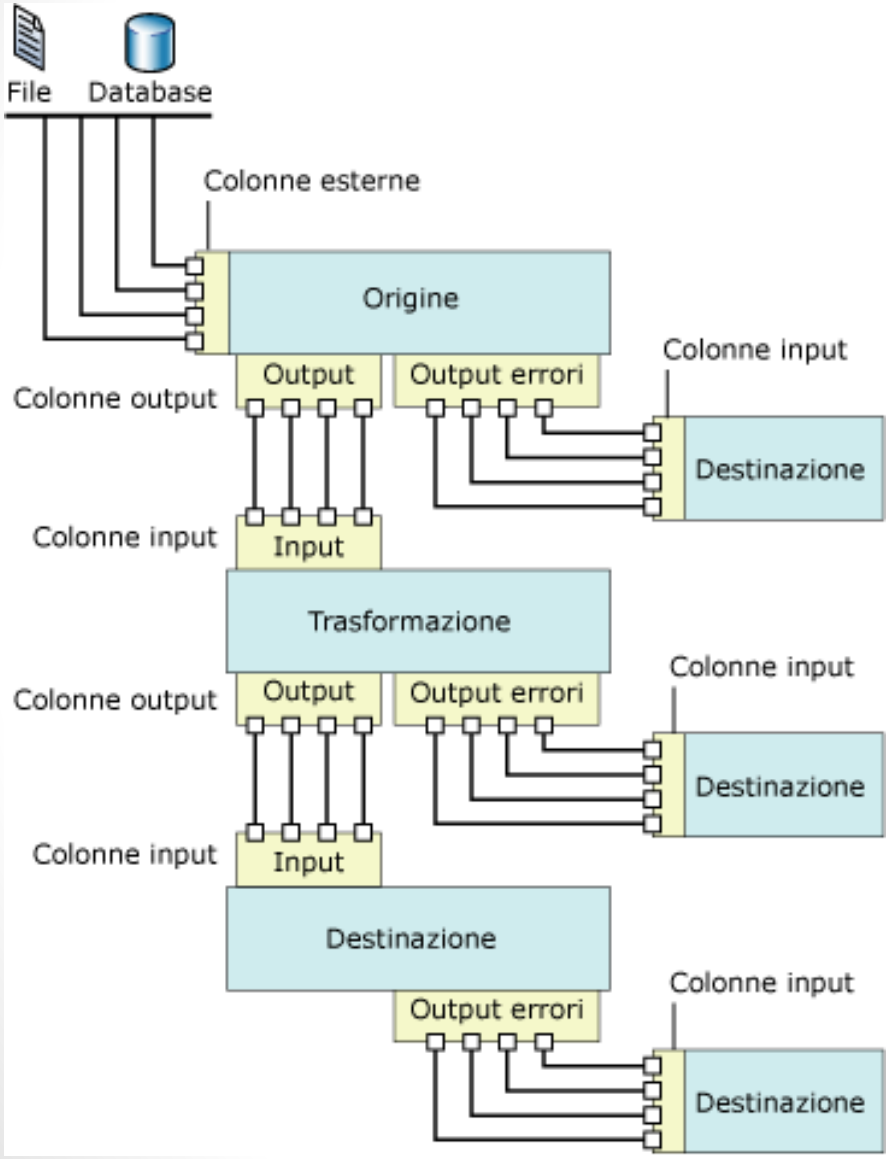
# Flusso di controllo

- Attività: elementi che eseguono i controlli definiti nel flusso dati; di seguito alcune possibili attività:
  - Attività Flusso Dati: specifica di origini, trasformazioni e destinazioni dati.
  - Attività di Preparazione Dati: copia di file, esecuzione di operazioni su file XML.
  - Attività di Scripting: definizione di procedure personalizzate per estendere le funzionalità dei pacchetti SSIS.
  - Attività di manutenzione: esecuzione di funzioni di amministrazione (es. procedure di backup su database SQL).
- Vincoli di precedenza: rappresentano dei connettori fra le attività e i contenitori di un pacchetto (flusso di controllo ordinato).

# Flusso dei Dati



# Flusso dati



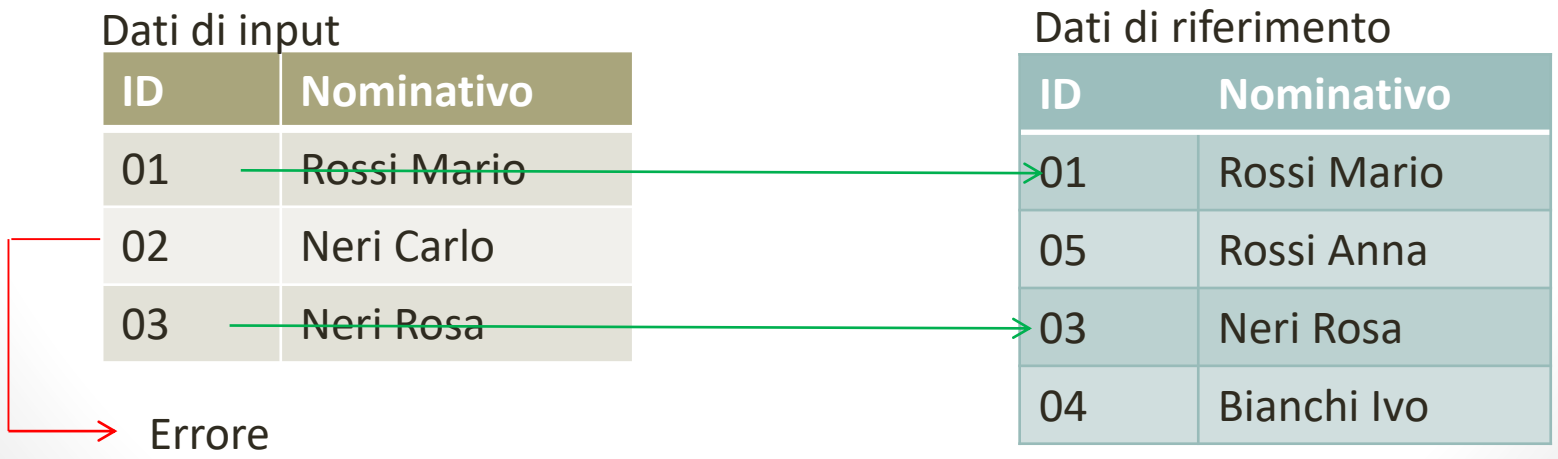

- Origine dati: controlli che permettono l'estrazione dei dati dalle sorgenti (es. Flat file, OLE DB, SQL server database).
- Trasformazioni: controlli per la definizione di trasformazioni sui dati.
- Destinazione dati: controlli per la memorizzazione dei dati trasformati sulle opportune destinazioni.

# Trasformazioni

- Ricerca (Lookup)
- Ricerca Fuzzy (Fuzzy Lookup )
- Raggruppamento Fuzzy (Fuzzy Grouping)
- Unione input multipli
- Suddivisione condizionale
- Merge join
- Colonna derivata
- Ordinamento: dei record di input
- Aggregazione: aggregazione dei dati di input
- Estrazione termini: estrazione di termini da campi testuali
- Conversione dati: trasformazione di tipo dei dati in input

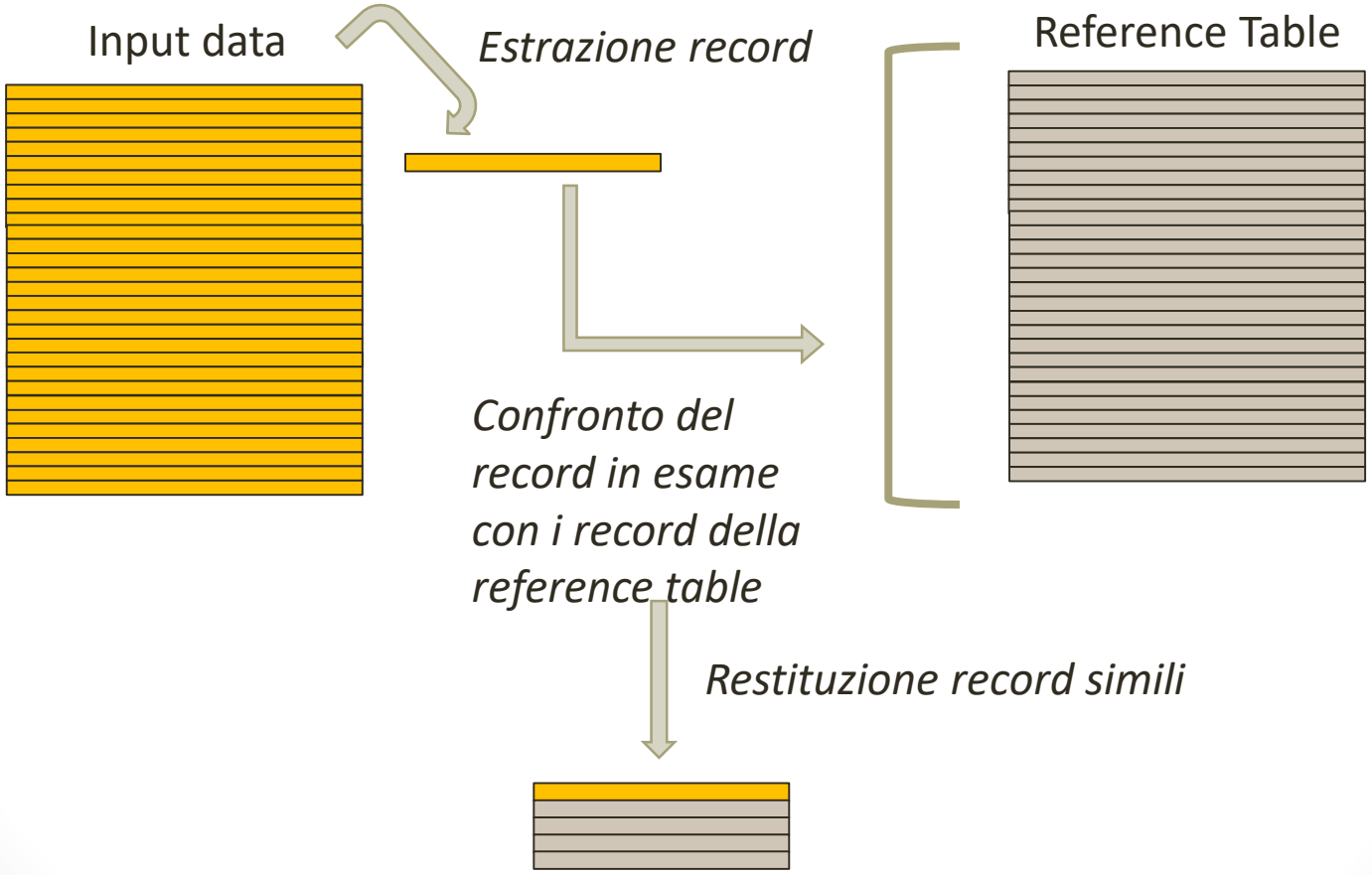
# Ricerca (Lookup)

- Esegue ricerche unendo in join (equi-join) i dati contenuti nelle colonne di input e le colonne in un set di dati di riferimento.
- Individua corrispondenze **esatte**.
- I record di input per cui non viene trovata alcuna corrispondenza, vengono gestiti come errori.
- In caso di corrispondenze multiple, viene mantenuta la prima corrispondenza individuata.
- I dati di riferimento devono essere memorizzati all'interno di un'origine dati OLE DB.



# Ricerca Fuzzy (Fuzzy Lookup)

La Ricerca Fuzzy esegue attività di pulitura dei dati (es. standardizzazione, correzione e inserimento di valori mancanti)



# Ricerca Fuzzy (Fuzzy Lookup)

- Reference table:
  - Deve essere una tabella memorizzata all'interno di un database di SQL Server (versione 2000 o successive).
  - I campi su cui viene applicata una corrispondenza fuzzy devono essere di tipo DT_WSTR o DT_STR.
- Parametri di input:
  - Numero massimo di corrispondenze: numero massimo di record simili da restituire per ciascun record di input.
  - Soglia: valore minimo di similarità affinché il record di input sia valutato come simile ad un record della reference table.
  - Delimitatori: delimitatori utilizzati per la suddivisione dei record in token (es. , ; .)

# Ricerca Fuzzy (Fuzzy Lookup)

- Algoritmo di confronto
  - Error-Tolerant Index (ETI): suddivide ciascun record in token o parole (la suddivisione avviene in base alla scelta di opportuni delimitatori).

| Stringa | Tokens |
| --- | --- |
| 13831 N.E. 8th St | 13831, N, E, 8th, and St |

(delimitatori: spazio, punto)

  - L'algoritmo confronta i token del record di input con i token presenti nella reference table.
  - La similarità fra token viene calcolata tenendo conto di:
    - Edit Distance a livello di token
    - Ordinamento dei token
    - Numero token simili
  - La similarità è calcolata nell'intervallo [0,1].
  - È possibile specificare un insieme limitato di campi del record su cui eseguire il confronto.

# Ricerca Fuzzy (Fuzzy Lookup)

- Output: è costituito da un insieme di campi che comprendono:
  - Il sottoinsieme di campi del record di input, i campi della reference table, valori di similarità e confidenza.
    - Similarità: grado di somiglianza tra i valori dei campi di input (record di input) e di riferimento (record reference table).
    - Confidenza: probabilità con cui un valore specifico risulta essere la corrispondenza migliore tra le corrispondenze individuate nella tabella di riferimento.

# Raggruppamento Fuzzy (Fuzzy grouping)

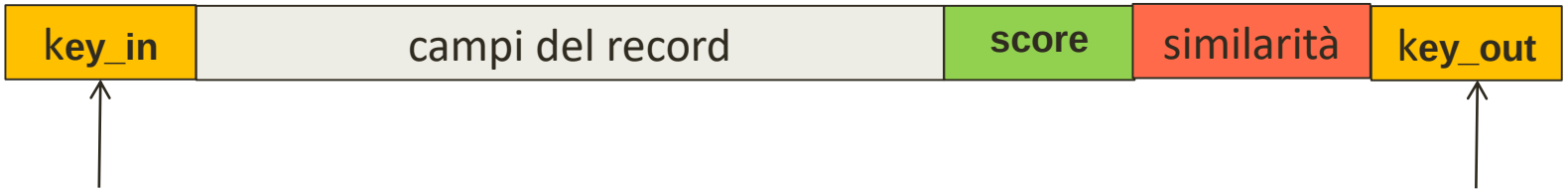
- Consente di eseguire operazioni di pulizia sui dati.
- Permette l'individuazione di eventuali duplicati e la creazione di un insieme di record standardizzati e ripuliti.
- Per la ricerca dei duplicati e dei record "rappresentanti" si basa sull'algoritmo Fuzzy Lookup.
- Non viene utilizzata una ReferenceTable, ma i record "rappresentanti" vengono scelti fra le righe dell'input.

# Fuzzy grouping

- Parametri di input:
  - Soglia: valore minimo di similarità affinché il record di input sia valutato come simile ad un record riferimento.
  - Delimitatori: utilizzati per la suddivisione dei record in token (es. , ; .).
- Algoritmo di confronto:
  - A ciascun record di input viene associato un identificatore key_in.
  - Il sistema deriva (tramite algoritmo interno) un insieme di record "rappresentanti".
  - Utilizza l'algoritmo Fuzzy Lookup per l'individuazione dei duplicati.
  - Al termine dell'algoritmo, il sistema ha individuato un insieme di record di riferimento e un insieme di potenziali record duplicati per ciascun record di rappresentante.
  - A ciascun record viene aggiunto un campo key_out che identifica il gruppo di appartenenza, ovvero memorizza l'identificatore univoco (key_in) del record rappresentante a cui il record in esame risulta simile.

# Fuzzy grouping

- Output:
  - I campi del record di input.
  - Similarità: grado di somiglianza tra i valori dei campi di input (record di input) e i campi del record di riferimento (record reference table).
  - Score: grado di somiglianza complessivo fra il record di input e il record di riferimento.
  - Valori di key_in e key_out.




Identificatore univoco

Identificatore del gruppo di appartenenza (key_in del record di riferimento)

I record riferimento hanno key_in=key_out

# Unione input multipli (Union all)

Consente di combinare più input in un unico output (es. è possibile utilizzare gli output di cinque diverse origini dati come input e combinarli in un singolo output).

- Gli input della trasformazione vengono aggiunti all'output della trasformazione uno dopo l'altro, senza riordinare le righe.
- Vengono inseriti valori nulli per i campi mancanti.

# Suddivisione condizionale (Conditional split)

- Distribuisce i record di input in diverse destinazioni a seconda dei criteri di suddivisione impostati (analogo ad un'istruzione di programmazione *switch*), per utilizzarlo è necessario:
  - Specificare uno o più condizioni da verificare durante la trasformazione.
  - Specificare l'ordine di valutazione delle condizioni.
  - Specificare l'output predefinito per i record che non soddisfano alcuna condizione.
- Applicazione:
  - Suddividere i record risultanti da una trasformazione di raggruppamento fuzzy per categorie di similarità.
  - Categorizzare i dati di input (es. classificazione dei clienti in differenti categorie di fedeltà al negozio, basandosi sui dati degli acquisti del cliente, risultanti da una procedura di integrazione).

# Merge Join

- Consente di definire il join fra record di due insieme di dati **ordinati sulle chiavi di join**.
- Tipi di join supportati:
  - Full outer join
  - Left outer join
  - Inner join

# Colonna derivata (Derived column)

- Consente di creare nuovi valori di colonna tramite l'applicazione di espressioni alle colonne di input della trasformazione.
- Applicazione:
  - In caso di record duplicati, si possono definire delle colonne derivate per stabilire quale valore associare ai campi per cui non esiste un valore univoco.
  - Sostituzione di valori mancanti.
  - Definire una priorità fra i valori dei campi di record diversi.

# Preparazione alle esercitazioni

- Aprire Visual Studio Creare nuovo Progetto: Integration Services Project
- Per le connessioni di tipo OLE DB utilizzare il seguenti parametri:
  - Server: ISI-SQLNEW
  - Database: SISSLab

# Esempio 1- Ricerca

- Specifiche:
  - Estrarre i record da due sorgenti distinte. I dati riguardano le vendite eseguite in stati diversi (USA e Canada). Ciascun record di vendita contiene anche il riferimento al prodotto oggetto della transazione. Il sistema dispone di un'anagrafica centralizzata dei prodotti in vendita. Si vuole verificare che i codici dei prodotti venduti abbiano una corrispondenza in anagrafica e fondere i dati in un'unica tabella.

VENDITE USA

```
IDPRODUCT;IDSALES
1;0011
2;0031
3;0050
4;0014
5;0037
6;0037
7;0037
```

VENDITE CANADA

```
IDPRODUCT;IDSALES
148;0001
882;0088
891;0150
1302;0001
1428;0043
103;0100
503;0101
503;0100
3;0100
```

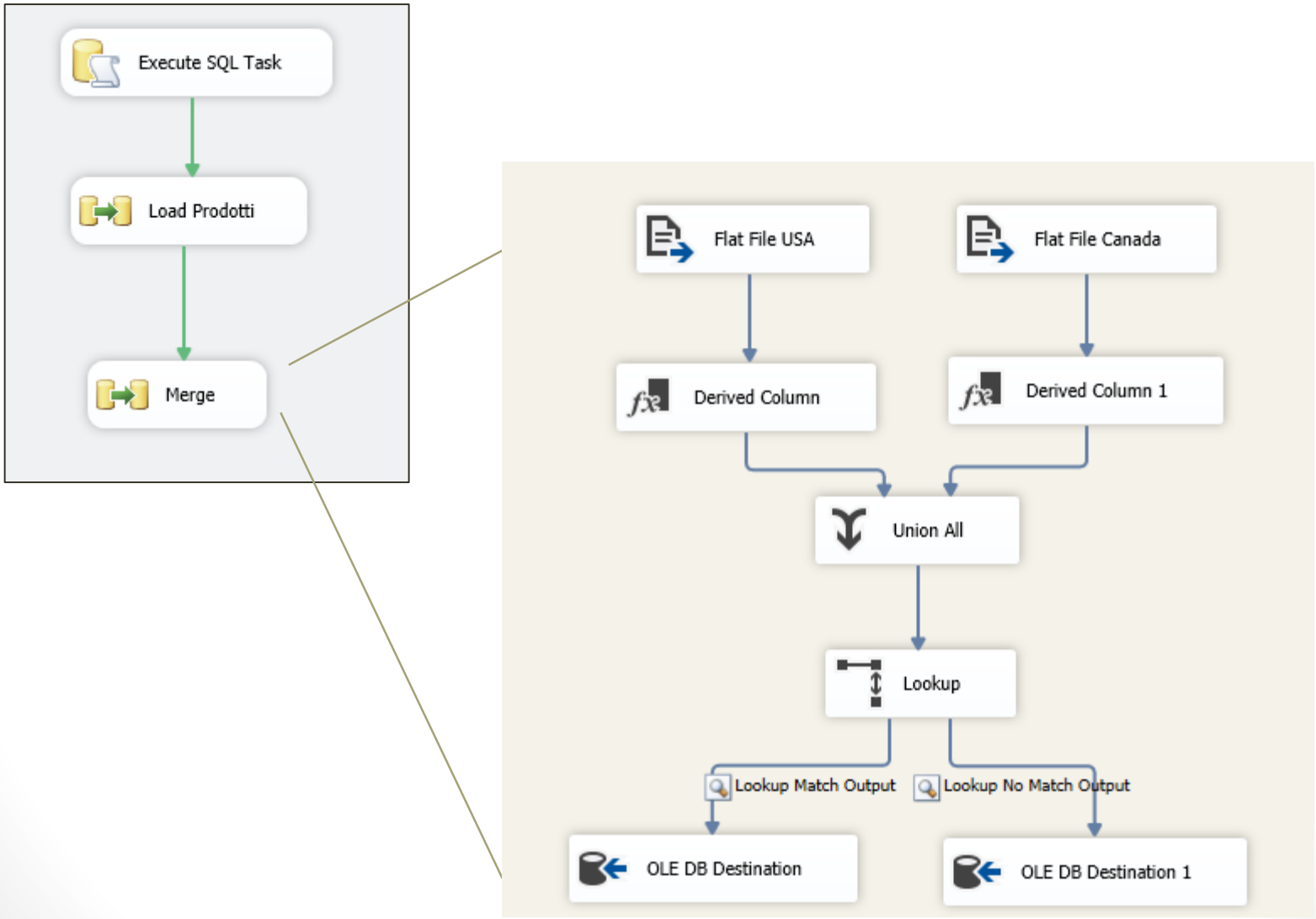
ANAGRAFICA PRODOTTI

```
IDPRODUCT
1
2
148
891
1302
1428
503
103
```

# Esempio 1

- Competenze:
  - Estrazione dati da più sorgenti.
  - Trasformazione Unione Multipla e Ricerca.
  - Gestione del flusso di errore.
  - Impostazione Sorgenti e Destinazioni.
- Traccia di soluzione:
  - Trasferire il file dei prodotti su DB
  - Unire i dati di input.
  - Confrontare i dati di input (vendite) con quelli presenti in una tabella di riferimento (anagrafica prodotti). Si ricercano corrispondenze esatte fra i record basandosi sul codice prodotto.
  - Gestire il flusso di errore per i record per cui non è stato trovata alcuna corrispondenza.
  - Creare una nuova tabella di database contenente i record per cui è stata individuata una corrispondenza.

# Esempio 1- Soluzione



# Esercizio 2 - Integrazione anagrafiche clienti

- Specifiche:
  - Integrare le anagrafiche clienti memorizzate da diverse reparti aziendali (vendite, call center, marketing) e individuare possibili record duplicati.

CUSTOMERS CALL CENTER

| ID | FirstN | MiddleI | LastName |
|---|---|---|---|
| 4 | Patty | T | Arun |
| 1 | Mary | A | Jane |
| 10 | Carl | R | Shor |
| 11 | Bridget | | Bhat |
| 105 | Caleb | | Bryan |
| 11 | Mitchell | D | Raji |
| 128 | Matthew | | Thompson |
| 1302 | Logan | | Simmons |

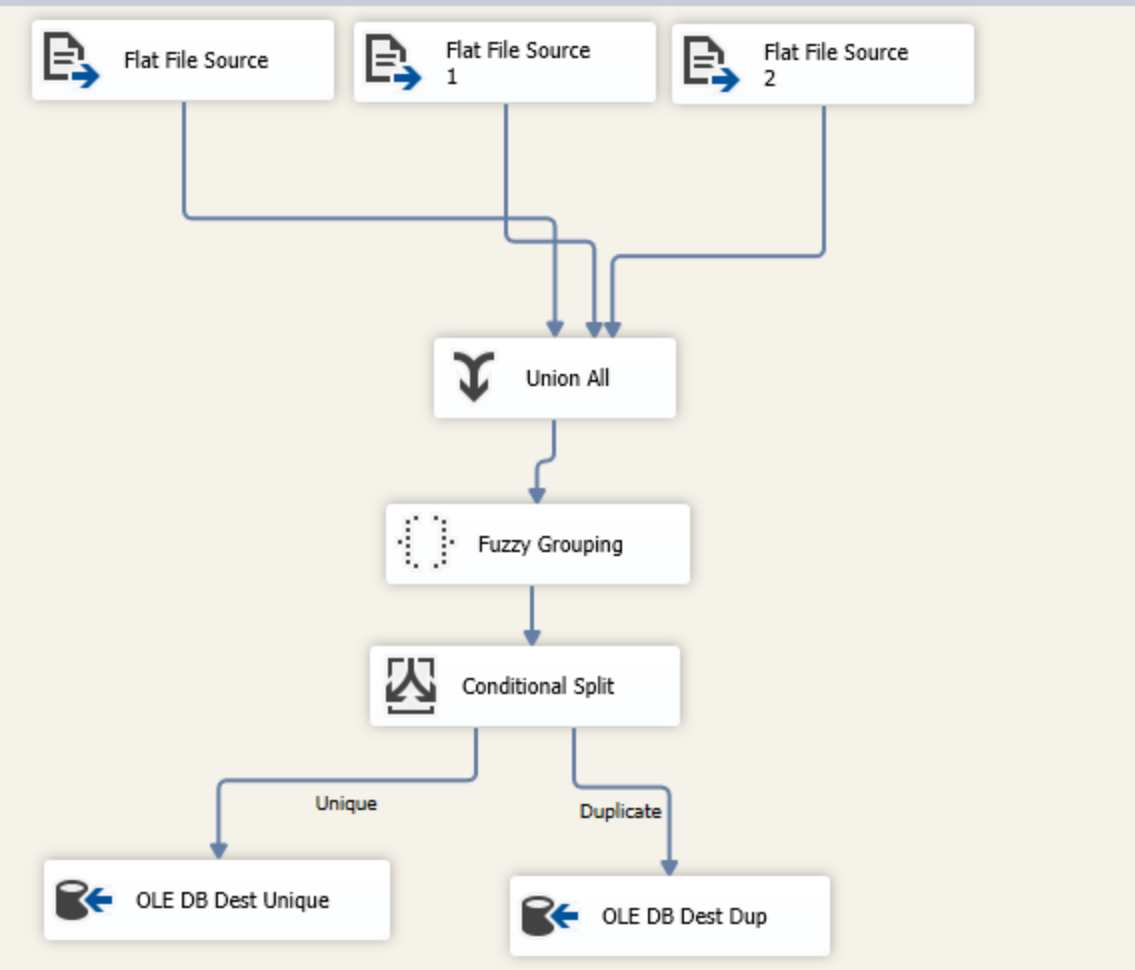
CUSTOMERS MKT

CUSTOMERS SALES

# Esercizio 2- Integrazione anagrafiche clienti

- Competenze:
  - Estrazione dati da più sorgenti.
  - Trasformazione Unione Multipla, Raggruppamento Fuzzy, Suddivisione Condizionale.
  - Impostazione Sorgenti e Destinazioni.
- Traccia soluzione:
  - Unire i dati di input.
  - Applicare un raggruppamento Fuzzy per l'individuazione dei duplicati (soglia 0.8)
  - Applicare una suddivisione condizionale per distinguere i dati standardizzati (puliti) dai potenziali duplicati. (_key_in == _key_out → puliti)
  - Memorizzare dell'output della suddivisione condizionale in corrispondenti tabelle di database SQL Server (generazione automatica).

# Esercizio 2- Soluzione



# Esercizio 3- Integrazione dati e aggiornamento campi

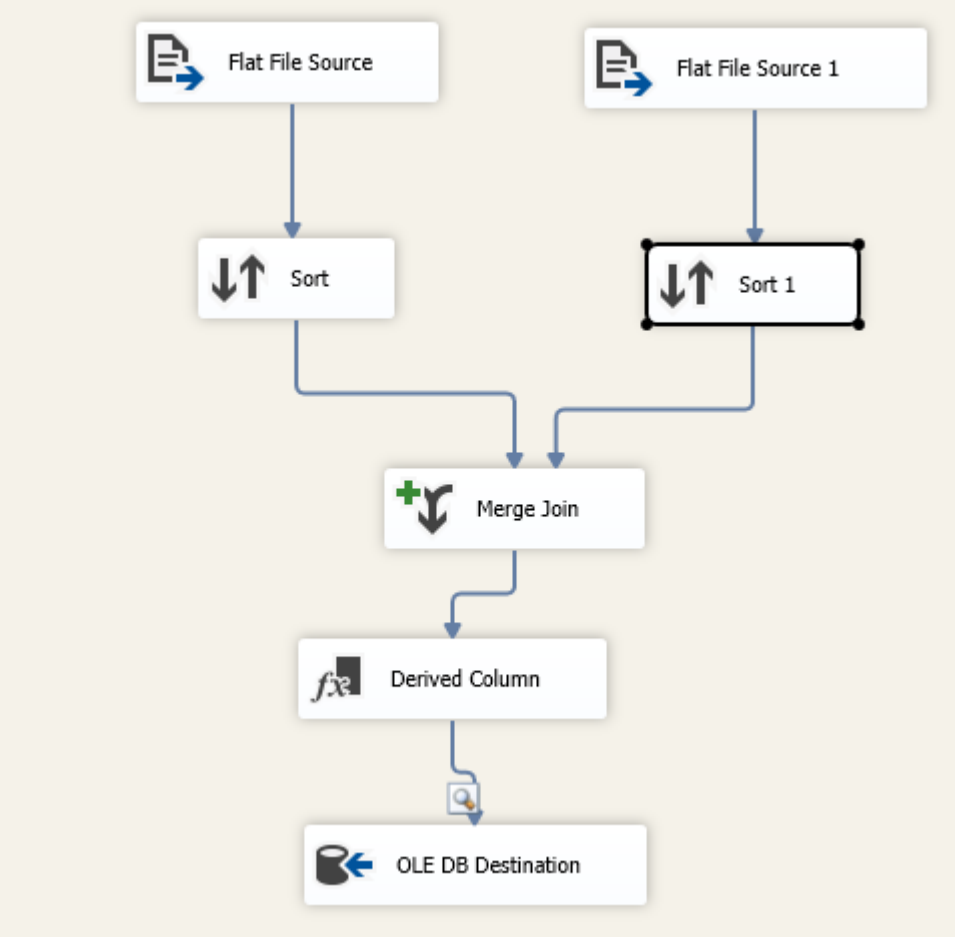
- Specifiche
  - Si vogliono integrare le anagrafiche clienti dei reparti vendite e mkt.
  - Le due anagrafiche memorizzano gli stessi clienti (congruenza fra ID), ma con diversi livelli di aggiornamento dai dati contenuti. Si vuole creare un'unica anagrafica contenente tutti i clienti e i dati aggiornati. In particolare:
    - MaritalStatus: è prevista una marca temporale per questo attributo. Si mantiene il valore più recente;
    - Phone: se presente si mantiene il valore specificato nell'anagrafica clienti del reparto MKT, altrimenti quello dell'anagrafica vendite;
    - Per tutti gli altri dati si mantengono i valori delle vendite;

| ID | FirstName | MiddleInitial | LastName | ValidityDate | MaritalStatus | EmailAddress | Address | City | State | Phone |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Abby | C | Malhotra | 12/11/2000 | M | amalhotra@thepho... | 1019 Carletto Drive | Sedro Woolley | WA | 645-555-... |
| 2 | Abby | | Prasad | 08/06/1952 | S | aprasad@blueyond... | 3261 Vista Bonita | Concord | CA | |
| 3 | Abby | | Srini | 20/4/1961 | M | asrini@adatum.com | 9191 Camelback Ct. | Berkeley | CA | 827-555-... |
| 4 | Abby | E | Rodriguez | 05/05/1951 | M | arodriguez@fabrika... | 6753 Howard Hugh... | Las Vegas | NV | 1 (11) 50... |
| 5 | Abigail | | Brown | 03/05/1946 | M | abrown@treyresear... | 4710 Northridge Drive | Port Orchard | WA | 155-555-... |
| 6 | Abigail | A | Watson | 21/7/1972 | S | awatson@thephone... | 8757 Keith Court | Seattle | WA | 1 (11) 50... |
| 7 | Abigail | C | Bryant | 06/09/1977 | M | abryant@northwind... | 2639 Anchor Court | Edmonds | WA | |
| 8 | Abigail | C | Hall | 24/6/1972 | S | ahall@northwindtra... | 8036 Summit View Dr. | Gold Bar | WA | |
| 9 | Abigail | | Davis | 22/12/1955 | S | adavis@fabrikam.com | 70 N.w. Plaza | Saint Ann | MO | 1 (11) 50... |
| 10 | Abigail | E | Flores | 27/1/1953 | M | aflores@baldwinmu... | 867 Maria Vega Court | Colma | CA | 622-555-... |

# Esercizio 3- Integrazione dati e aggiornamento campi

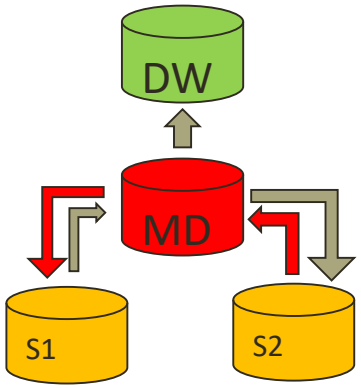
- Competenze:
  - Estrazione dati da due sorgenti.
  - Trasformazione Merge Join, Colonna derivata.
- Traccia di soluzione
  - Ordinare i dati di input sul campo ID.
  - Applicare un merge Join per fondere i due file (Inner join).
  - Utilizzare una colonna derivata per la selezione dei campi MaritalStatus e Phone in base alle specifiche (Usare nvarchar per le nuove colonne per evitare problemi di conversione)
    - ValidityDate > ValidityDate_MKT ? MaritalStatus : MaritalStatus_MKT
    - !ISNULL(Phone_MKT) ? Phone_MKT : Phone
  - Memorizzare dell'output in una nuova tabella di database SQL Server.

# Esercizio 3- Soluzione



# Esercizio 4: MDM coesistenza

- Si vuole progettare una architettura per MDM in cui un hub centrale (CLIENTI_GLOBAL) mantiene una versione aggiornata dei dati caricati dalle sorgenti (CLIENTI SORGENTE 1 e CLIENTI SORGENTE 2).
- Il progetto consiste nella definizione di 2 flussi dati sulla base di criteri dati:
  1. Caricamento quotidiano dei clienti modificati nella sorgente 1 sull'hub
  2. Aggiornamento asincrono dei dati della sorgente 2 a partire dall'hub




- NOTA: il join tra sorgenti e hub è fatto della base della similarità ($\sigma$>0.8) del campo NOMINATIVO. In caso di confidenza minore di 1 l'aggiornamento deve essere manuale
- Gestione degli identificatori univoci (campo ID)
  - Quando viene inserito un nuovo record sull'hub oppure sulle sorgenti è necessario generare un nuovo ID (autoincremento)

# Dati di input

**CLIENTI_GLOBAL**

| ID | NOMINATIVO | PROFESSIONE | TELFISSO | DATARECORD |
|---|---|---|---|---|
| 01 | Mario Rossi | MANAGER | 054214785 | 2011-11-13 |
| 02 | Mario Neri | MANAGER | 054214785 | 2011-06-24 |
| 03 | Mauro Nero | IMPIEGATO | 05463256 | 2011-11-04 |
| 04 | Anna Verdi | IMPIEGATO | 05442356 | 2011-11-05 |
| 05 | Anna Verde | SEGRETARIA | 054785236 | 2011-01-17 |
| 06 | Anna Vardi | MANAGER | 05489632 | 2011-02-07 |

**CLIENTI SORGENTE 1**

| ID | NOMINATIVO | PROFESSIONE | DATARECORD |
|---|---|---|---|
| 01 | Elisa Turricchia | STUDENTE | 2011-11-25 |
| 02 | Mario Rossi | IMPIEGATO | 2011-11-10 |
| 03 | Mario Neri | IMPIEGATO | 2011-11-25 |
| 04 | Anna Verdi | DIRETTORE | 2011-11-25 |

**CLIENTI SORGENTE 2**

| ID | NOMINATIVO | TELFISSO | PROFESSIONE | DATARECORD |
|---|---|---|---|---|
| 04 | Mario Rossi | 054214785 | MANAGER | 2011-11-13 |
| 01 | Anna Verde | 054785236 | SEGRETARIA | 2011-11-01 |

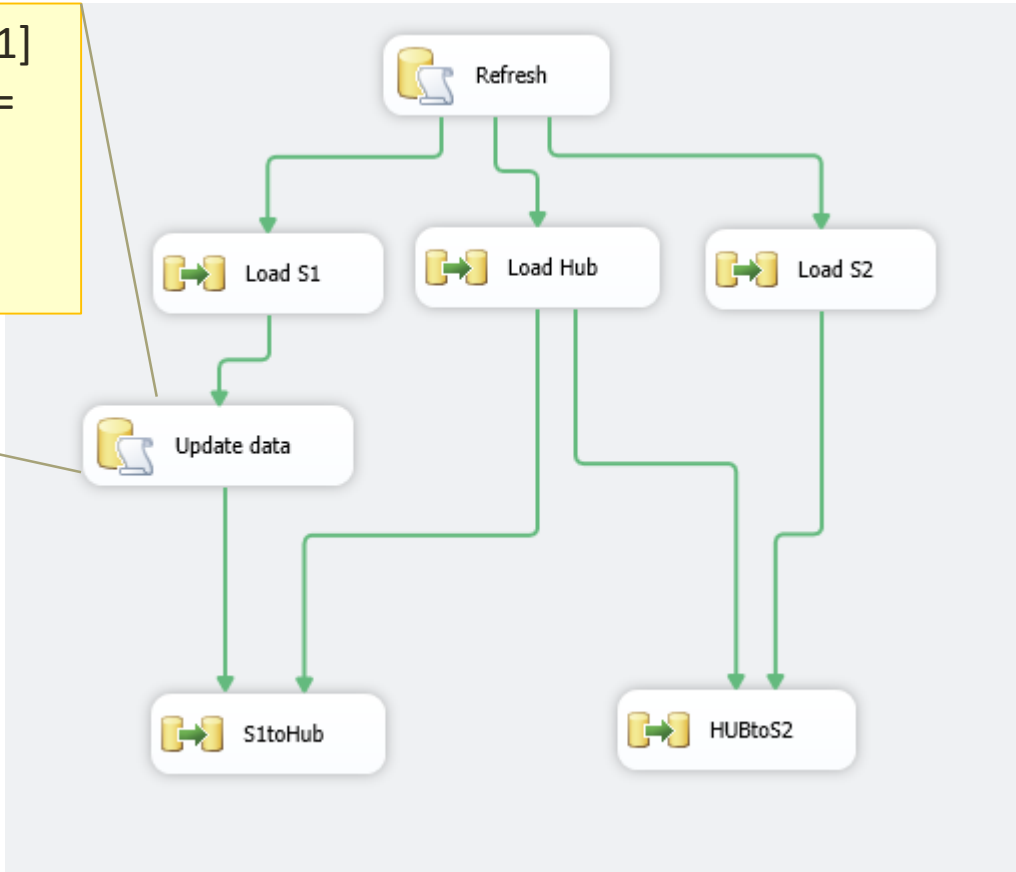
# Criteri di aggiornamento dei dati

- SORGENTE 1 → HUB
  - Dalla sorgente vengono estratti tutti i record con DATARECORD uguale alla data odierna
  - Nuovo cliente → inserimento di un nuovo record nell'hub
  - Cliente preesistente → l'aggiornamento avviene se è stata eseguita una modifica sul campo PROFESSIONE

- HUB → SORGENTE2
  - Vengono estratti tutti i record dell'hub
  - Nuovo cliente → inserimento di un nuovo record nella sorgente
  - Cliente preesistente → l'aggiornamento avviene se il record dell'hub è più recente di quello della sorgente

# Soluzione: flusso di controllo

```
UPDATE [Sorgente1]
SET DATARECORD =
CONVERT(date,
SYSDATETIME())
WHERE ID>2
```



# Creazione tabelle

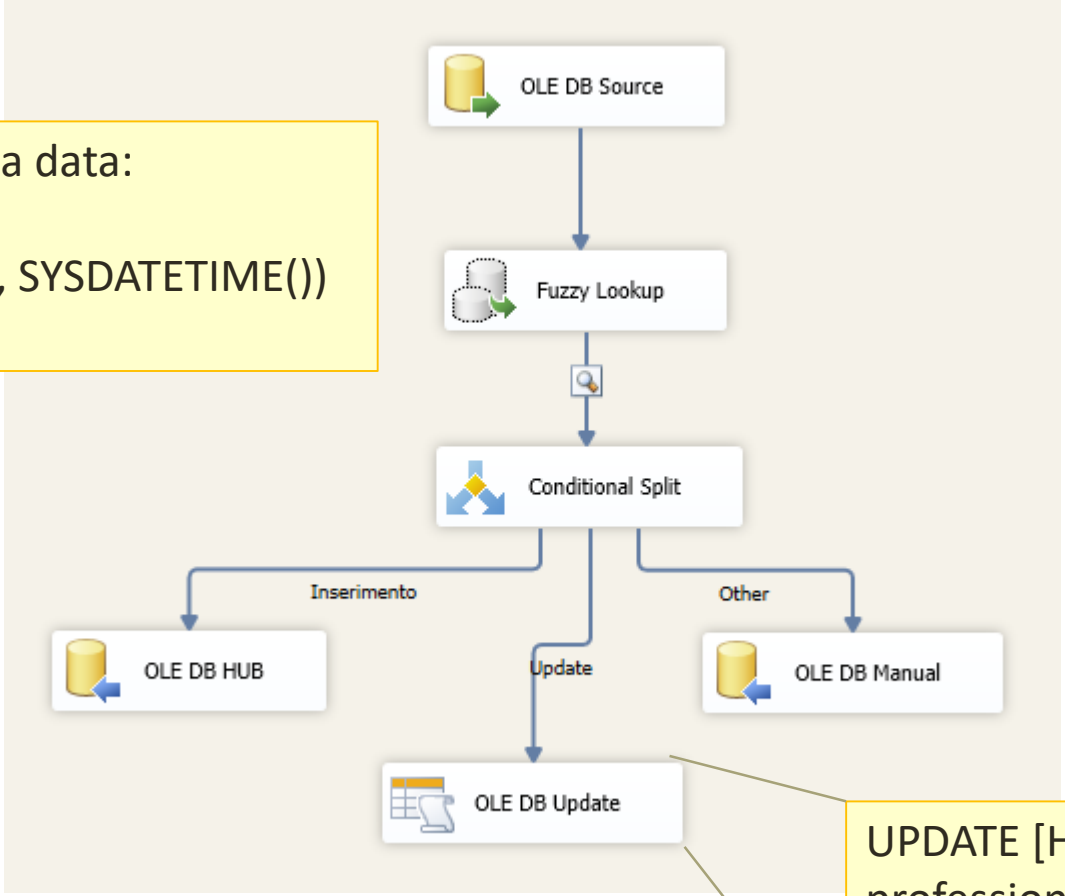
```
CREATE TABLE [Sorgente1](
        [ID] int IDENTITY(1,1),
        [NOMINATIVO] [varchar](50) NULL,
        [PROFESSIONE] [varchar](50) NULL,
        [DATARECORD] date NULL
)
```

```
CREATE TABLE [Hub](
        [ID] int IDENTITY(1,1),
        [NOMINATIVO] [varchar](50) NULL,
        [PROFESSIONE] [varchar](50) NULL,
        [DATARECORD] date NULL,
        [TELFISSO] [varchar](50) NULL
)
```

Settare a true il flag KEEP IDENTITY nel mapping

# Soluzione: Flusso dati S1→ Hub

Condizione sulla data:
DATARECORD=
CONVERT(date, SYSDATETIME())




UPDATE [HUB] set professione=?, datarecord=? WHERE ID= ?

# Soluzione: Flusso dati Hub→ S2