

Candidato:

Andreoli Alberto

Relatore:

**Prof. Sonia
Bergamaschi**

**Progettazione e sviluppo di tecniche di full outer
join distribuito per il sistema di data integration
MOMIS tramite il framework Apache Flink**

Università degli Studi di Modena e Reggio Emilia
Dipartimento di Ingegneria "Enzo Ferrari"
Laurea Magistrale in Ingegneria Informatica (270/04)
Anno Accademico 2014/2015

Introduzione

- **Tirocinio** universitario presso DataRiver S.r.l.
- **Momis:** Data Integration framework virtuale
- **Flink:** open-source framework sviluppato dalle università di Berlino ed acquisito dalla Community della Apache Software Foundation per l'analisi distribuita di grandi quantità di dati
 - **Docker:** framework per la facile creazione e deployment di macchine virtuali

Obiettivo:

- Sviluppo di una classe per l'esecuzione di un Full Outer Join ad N vie tramite il framework Flink per l'utilizzo in Momis

Data Integration (motivazioni)

- Grandi quantità di dati organizzati secondo diversi schemi
- Aziende diverse hanno schemi diversi
- Problemi di reperimento ed analisi di dati provenienti da più fonti

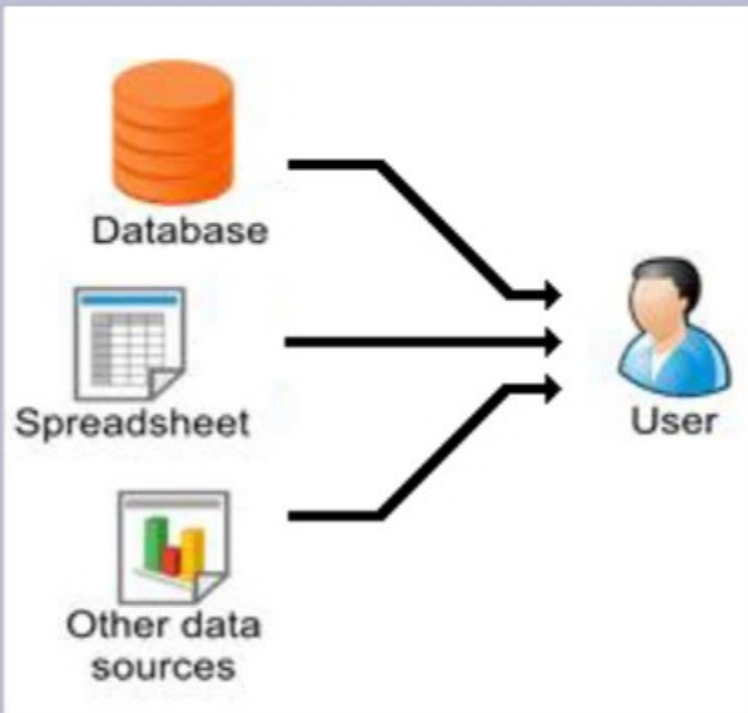
Retrocompatibilità

**Esempio
di due
schemi:**

Persone				
ID	Nome	Cognome	DataDiNascita	mail
0	Priscilla	Padovano	04/09/1939	PriscillaPadovano@libero.it
1	Marcella	Ferri	03/12/1944	MarcF@gmail.com
2	Addolorata	Manna	27/03/1996	AddolorataManna1@gmail.com
3	Indro	Milano	05/01/1972	pino274@libero.it
4	Pietro	Cavalli	06/06/1966	PietroCavalli4@libero.it

Utenti			
Codice	Nome	Data_Nascita	Telefono
1A001	Marcella Ferri	1944-12-03	0376-4549364
1A002	Priscilla Padovana	1939-09-04	0326-4548353
1A003	Gherardo Letti	1996-03-27	0442-4117940
1A004	Giovanni Castagna	1972-01-08	0535-9186020

Data Integration (motivazioni)



**Metodi di
salvataggio diversi**



Excel

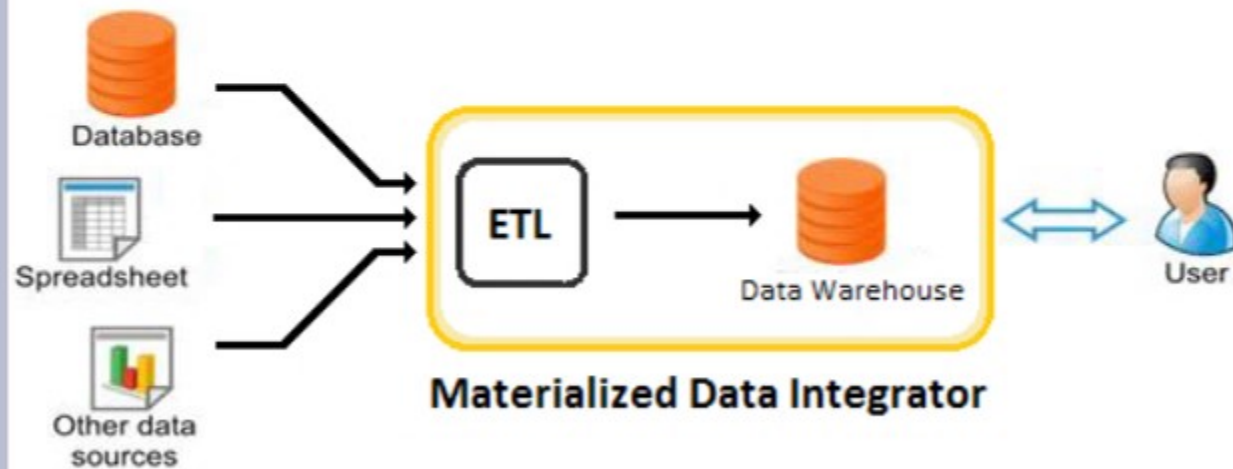


File XML



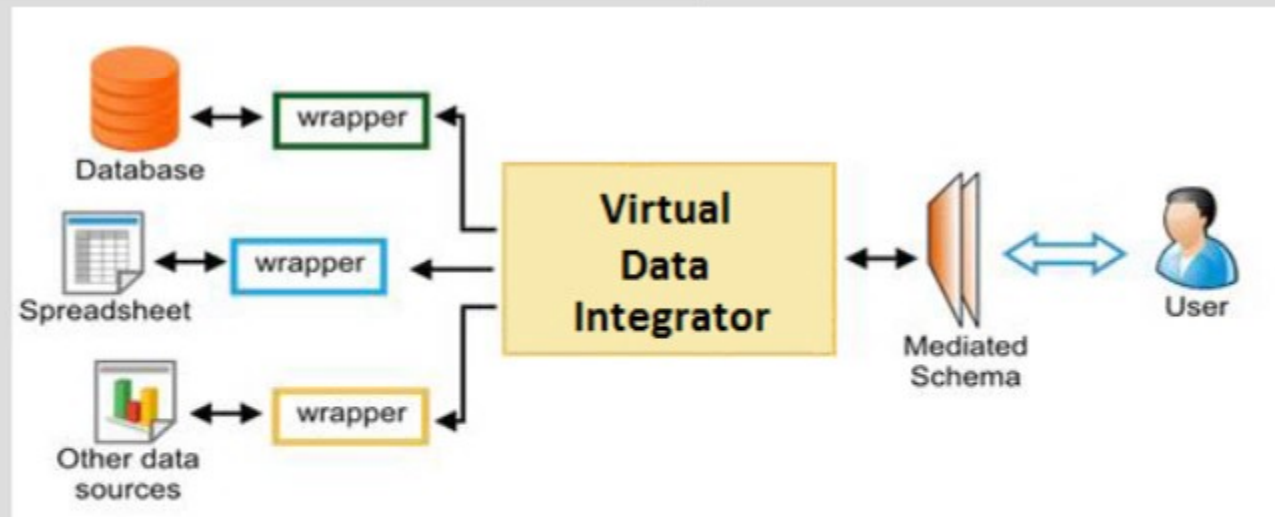
**L'utente affronta
molti problemi**

Data Integration (motivazioni)



**Soluzione:
Data Integration
System**

**Esegue
l'integrazione
automaticamente
per l'utente**



Data Integration (SG & SL)

- **Schemi Locali:** strutture dati delle fonti
- **Schema Globale:** visione integrata e di insieme degli Schemi Locali
- L'utente Finale interagisce solamente con lo Schema Globale.

Persone(ID, Nome, Cognome, DataDiNascita, mail)
Libri(ID, Titolo, Pagine, Autore, DataDiPubblicazione)
LibriLetti(IDPersona, IDLibro, Data)
FK: IDPersona → Persone(ID)
FK: IDLibro → Libri(ID)

**Necessita di una
visione unificata**

Utenti(Codice, Nome, Data_Nascita, Telefono)
Libri(Codice, Titolo, N_Pagine, Data_Pubblicazione)
Autori(Codice, Nome, Data_Nascita)
Autorato(Codice_Libro, Codice_Autore)
FK: Codice_Libro → Libri(Codice)
FK: Codice_Autore → Autori(Codice)

Persone(ID, Nome, Cognome, DataDiNascita, mail)
Libri(ID, Titolo, Pagine, Autore, DataDiPubblicazione)
LibriLetti(IDPersona, IDLibro, Data)
FK: IDPersona → Persone(ID)
FK: IDLibro → Libri(ID)

Mappaggio tra Schemi Locali e Schema Globale

Utenti(Codice, Nome, Data_Nascita, Telefono)
Libri(Codice, Titolo, N_Pagine, Data_Pubblicazione)
Autori(Codice, Nome, Data_Nascita)
Autorato(Codice_Libro, Codice_Autore)
FK: Codice_Libro → Libri(Codice)
FK: Codice_Autore → Autori(Codice)

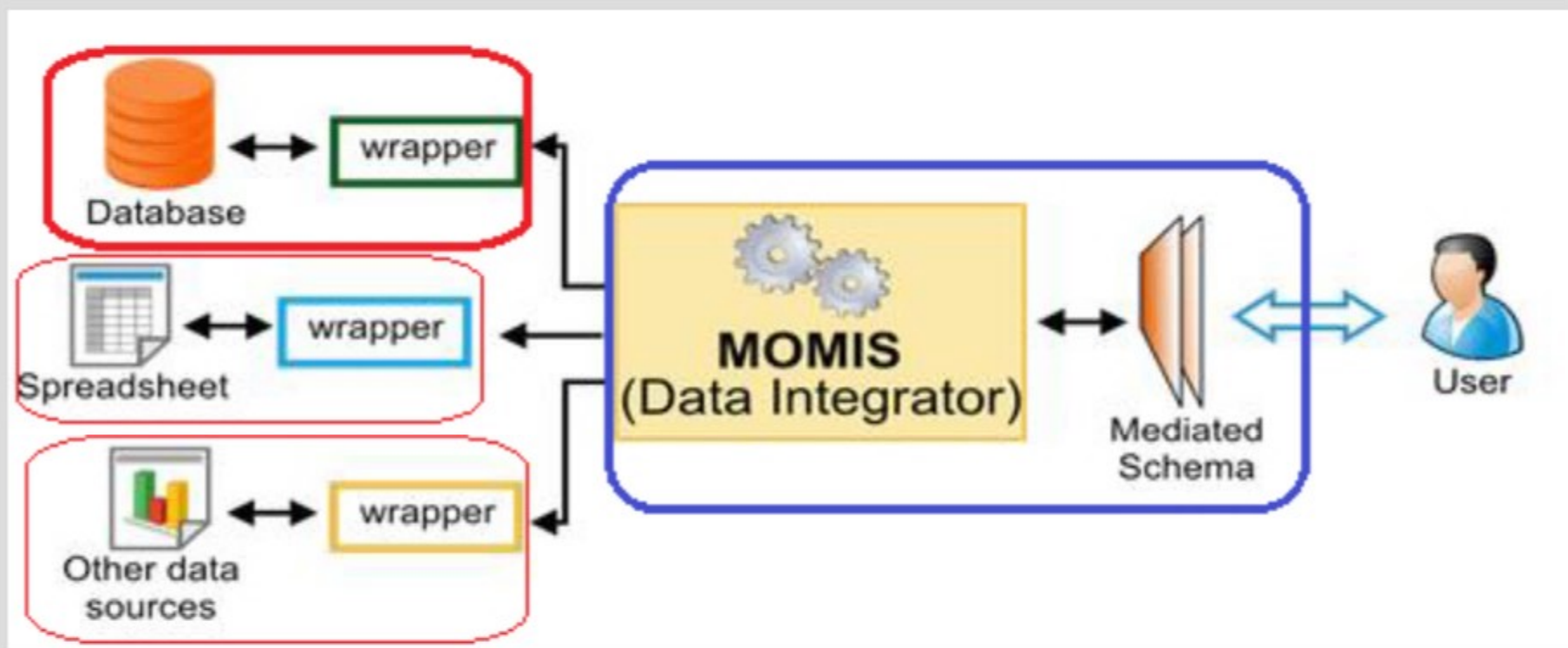
Utenti(ID, Nome, Data_Nascita, Telefono, mail)
Libri(ID, Titolo, N_Pagine, Data_Pubblicazione)
Autori(ID, Nome, Data_Nascita)
Autorato(ID_Libro, ID_Autore)
FK: ID_Libro → Libri(ID)
FK: ID_Autore → Autori(ID)
LibriLetti(ID_Utente, ID_Libro, Data)
FK: ID_Utente → Utenti(ID)
FK: ID_Libro → Libri(ID)

Momis

- MOMIS: **M**ediator **e**nvironment for **M**ultiple **I**nformation **S**ources
- Data-Integration virtuale di fonti strutturate e semi-strutturate
- Sviluppato dal DBGroup di Modena e distribuito open-source dall'azienda DataRiver



Momis



Momis: sistema
distribuito

Momis (funzionamento)

Persone(ID, Nome, Cognome, DataDiNascita, mail)
 Libri(ID, Titolo, Pagine, Autore, DataDiPubblicazione)
 LibriLetti(IDPersona, IDLibro, Data)
 FK: IDPersona → Persone(ID)
 FK: IDLibro → Libri(ID)

Utenti(Codice, Nome, Data_Nascita, Telefono)
 Libri(Codice, Titolo, N_Pagine, Data_Pubblicazione)
 Autori(Codice, Nome, Data_Nascita)
 Autorato(Codice_Libro, Codice_Autore)
 FK: Codice_Libro → Libri(Codice)
 FK: Codice_Autore → Autori(Codice)

Utenti(ID, Nome, Data_Nascita, Telefono, mail)
 Libri(ID, Titolo, N_Pagine, Data_Pubblicazione)
 Autori(ID, Nome, Data_Nascita)
 Autorato(ID_Libro, ID_Autore)
 FK: ID_Libro → Libri(ID)
 FK: ID_Autore → Autori(ID)
 LibriLetti(ID_Utente, ID_Libro, Data)
 FK: ID_Utente → Utenti(ID)
 FK: ID_Libro → Libri(ID)



SELECT ...
FROM ...
WHERE ...

L'utente interroga
 lo Schema Globale
 tramite una query
SELECT

Momis (funzionamento)

Persone(ID, Nome, Cognome, DataDiNascita, mail)
 Libri(ID, Titolo, Pagine, Autore, DataDiPubblicazione)
 LibriLetti(IDPersona, IDLibro, Data)
 FK: IDPersona → Persone(ID)
 FK: IDLibro → Libri(ID)

SELECT ...
 FROM ...
 WHERE ...

Utenti(Codice, Nome, Data_Nascita, Telefono)
 Libri(Codice, Titolo, N_Pagine, Data_Pubblicazione)
 Autori(Codice, Nome, Data_Nascita)
 Autorato(Codice_Libro, Codice_Autore)
 FK: Codice_Libro → Libri(Codice)
 FK: Codice_Autore → Autori(Codice)

SELECT ...
 FROM ...
 WHERE ...

Utenti(ID, Nome, Data_Nascita, Telefono, mail)
 Libri(ID, Titolo, N_Pagine, Data_Pubblicazione)
 Autori(ID, Nome, Data_Nascita)
 Autorato(ID_Libro, ID_Autore)
 FK: ID_Libro → Libri(ID)
 FK: ID_Autore → Autori(ID)
 LibriLetti(ID_Utente, ID_Libro, Data)
 FK: ID_Utente → Utenti(ID)
 FK: ID_Libro → Libri(ID)

La query Globale
 viene tradotta in
 query Locali per il
 reperimento dei
 dati sulle sorgenti

Momis (funzionamento)

Persone(ID, Nome, Cognome, DataDiNascita, mail)
 Libri(ID, Titolo, Pagine, Autore, DataDiPubblicazione)
 LibriLetti(IDPersona, IDLibro, Data)
 FK: IDPersona → Persone(ID)
 FK: IDLibro → Libri(ID)

Utenti(Codice, Nome, Data_Nascita, Telefono)
 Libri(Codice, Titolo, N_Pagine, Data_Pubblicazione)
 Autori(Codice, Nome, Data_Nascita)
 Autorato(Codice_Libro, Codice_Autore)
 FK: Codice_Libro → Libri(Codice)
 FK: Codice_Autore → Autori(Codice)

Utenti(ID, Nome, Data_Nascita, Telefono, mail)
 Libri(ID, Titolo, N_Pagine, Data_Pubblicazione)
 Autori(ID, Nome, Data_Nascita)
 Autorato(ID_Libro, ID_Autore)
 FK: ID_Libro → Libri(ID)
 FK: ID_Autore → Autori(ID)
 LibriLetti(ID_Utente, ID_Libro, Data)
 FK: ID_Utente → Utenti(ID)
 FK: ID_Libro → Libri(ID)

<record>
 <record>
 <record>

<record>
 <record>
 <record>

Le sorgenti
 ritornano i record
 richiesti

Momis (funzionamento)

Persone(ID, Nome, Cognome, DataDiNascita, mail)
Libri(ID, Titolo, Pagine, Autore, DataDiPubblicazione)
LibriLetti(IDPersona, IDLibro, Data)
FK: IDPersona → Persone(ID)
FK: IDLibro → Libri(ID)

Utenti(Codice, Nome, Data_Nascita, Telefono)
Libri(Codice, Titolo, N_Pagine, Data_Pubblicazione)
Autori(Codice, Nome, Data_Nascita)
Autorato(Codice_Libro, Codice_Autore)
FK: Codice_Libro → Libri(Codice)
FK: Codice_Autore → Autori(Codice)

Utenti(ID, Nome, Data_Nascita, Telefono, mail)
Libri(ID, Titolo, N_Pagine, Data_Pubblicazione)
Autori(ID, Nome, Data_Nascita)
Autorato(ID_Libro, ID_Autore)
FK: ID_Libro → Libri(ID)
FK: ID_Autore → Autori(ID)
LibriLetti(ID_Utente, ID_Libro, Data)
FK: ID_Utente → Utenti(ID)
FK: ID_Libro → Libri(ID)

<record>
<record>
<record>

<record>
<record>
<record>



Full Outer
Join molto
pesante
trattandosi di
Big Data

<record>
<record>
<record>

Esecuzione del
Full Outer Join con
integrazione dei
dati delle fonti

Ritorno dei dati
all'utente

Momis (Full Outer Join)

Full Outer
Join

Persone			
ID	Nome	DataDiNascita	mail
0	Priscilla Padovana	04/09/1939	PriscillaPadovano@libero.it
1	Marcella Ferri	03/12/1944	MarcF@gmail.com
2	Addolorata Manna	27/03/1996	AddolorataManna1@gmail.com
3	Indro Milano	05/01/1972	pino274@libero.it
4	Pietro Cavalli	06/06/1966	PietroCavalli4@libero.it

Utenti			
Codice	Nome	Data_Nascita	Telefono
1A001	Marcella Ferri	1944-12-04	0376-4549364
1A002	Priscilla Padovana	1939-09-04	0326-4548353
1A003	Gherardo Letti	1996-03-27	0442-4117940
1A004	Giovanni Castagna	1972-01-08	0535-9186020

Persone			
Nome	DataDiNascita	mail	Telefono
Priscilla Padovana	04/09/1939	PriscillaPadovano@libero.it	0326-4548353
Marcella Ferri	03/12/1944	MarcF@gmail.com	0376-4549364
Addolorata Manna	27/03/1996	AddolorataManna1@gmail.com	<i>null</i>
Indro Milano	05/01/1972	pino274@libero.it	<i>null</i>
Pietro Cavalli	06/06/1966	PietroCavalli4@libero.it	<i>null</i>
Gherardo Letti	04/12/1944	<i>null</i>	0442-4117940
Giovanni Castagna	08/01/1972	<i>null</i>	0535-9186020

Momis (Full Outer Join)

Risoluzione
dei conflitti
&
Integrazione

Persone			
ID	Nome	DataDiNascita	mail
0	Priscilla Padovana	04/09/1939	PriscillaPadovano@libero.it
1	Marcella Ferri	03/12/1944	MarcF@gmail.com
2	Addolorata Manna	27/03/1996	AddolorataManna1@gmail.com
3	Indro Milano	05/01/1972	pino274@libero.it
4	Pietro Cavalli	06/06/1966	PietroCavalli4@libero.it

Utenti			
Codice	Nome	Data_Nascita	Telefono
1A001	Marcella Ferri	1944-12-04	0376-4549364
1A002	Priscilla Padovana	1939-09-04	0326-4548353
1A003	Gherardo Letti	1996-03-27	0442-4117940
1A004	Giovanni Castagna	1972-01-08	0535-9186020

Big data
necessitano
un Full Outer
Join
distribuito ed
ottimizzato

Persone			
Nome	DataDiNascita	mail	Telefono
Priscilla Padovana	04/09/1939	PriscillaPadovano@libero.it	0326-4548353
Marcella Ferri	03/12/1944	MarcF@gmail.com	0376-4549364
Addolorata Manna	27/03/1996	AddolorataManna1@gmail.com	<i>null</i>
Indro Milano	05/01/1972	pino274@libero.it	<i>null</i>
Pietro Cavalli	06/06/1966	PietroCavalli4@libero.it	<i>null</i>
Gherardo Letti	04/12/1944	<i>null</i>	0442-4117940
Giovanni Castagna	08/01/1972	<i>null</i>	0535-9186020

Momis (Flink FOJ)

Flink è una
piattaforma
per la
gestione di
Big Data.

```

Persone(ID, Nome, Cognome, DataDiNascita, mail)
Libri(ID, Titolo, Pagine, Autore, DataDiPubblicazione)
LibriLetti(IDPersona, IDLibro, Data)
FK: IDPersona → Persone(ID)
FK: IDLibro → Libri(ID)
    
```

```

SELECT ...
FROM ...
WHERE ...
    
```

Flink

```

Utenti(ID, Nome, Data_Nascita, Telefono, mail)
Libri(ID, Titolo, N_Pagine, Data_Pubblicazione)
Autori(ID, Nome, Data_Nascita)
Autorato(ID_Libro, ID_Autore)
FK: ID_Libro → Libri(ID)
FK: ID_Autore → Autori(ID)
LibriLetti(ID_Utente, ID_Libro, Data)
FK: ID_Utente → Utenti(ID)
FK: ID_Libro → Libri(ID)
    
```

Flink

```

Utenti(Codice, Nome, Data_Nascita, Telefono)
Libri(Codice, Titolo, N_Pagine, Data_Pubblicazione)
Autori(Codice, Nome, Data_Nascita)
Autorato(Codice_Libro, Codice_Autore)
FK: Codice_Libro → Libri(Codice)
FK: Codice_Autore → Autori(Codice)
    
```

```

SELECT ...
FROM ...
WHERE ...
    
```

Flink

**FULL
OUTER
JOIN
MERGE**

Si vuole creare un
Full Outer Join
tramite Flink

Flink

- Flink: open-source framework per l'analisi distribuita di Big Data.
- Core scritto in java e scala
- Stream e Batch
- Sorgenti locali o cloud

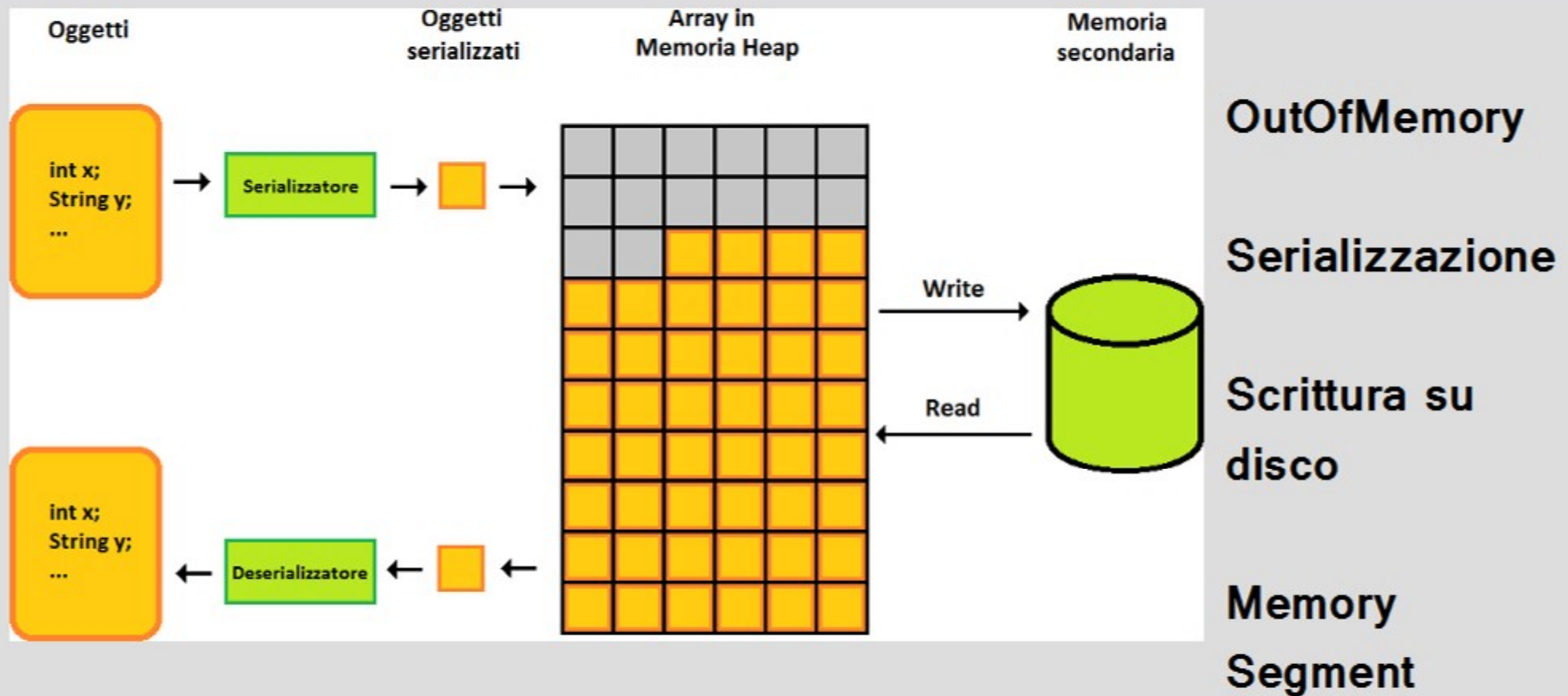


Il motore di **Flink** verrà trattato da
Guido Mazza



Flink

(Data management)

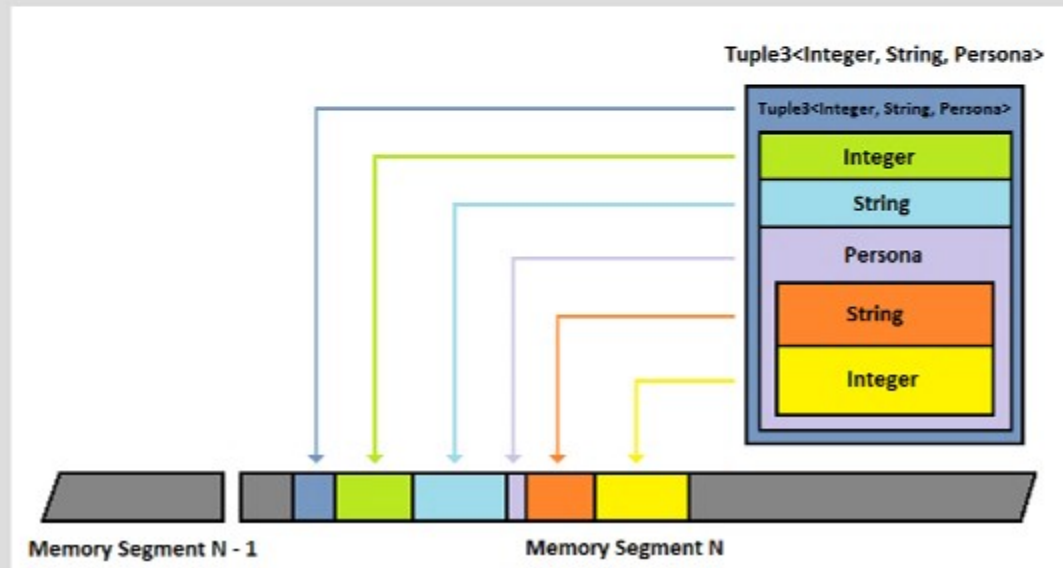


Come Flink
gestisce Big Data



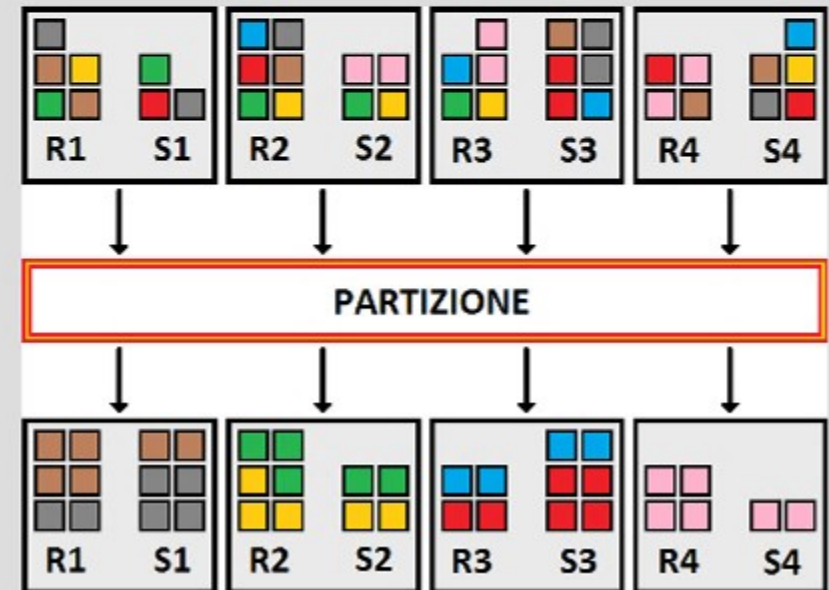
Flink

(Data management)



Serializzazione personalizzata

Gestione trasparente all'utente



Controllo automatico e trasparente dei dati

Flink sa come partizionare al meglio i dati



Flink (Stack)

Livelli di astrazione:

API di alto livello:

- Flink ML
- Gelly
- Table API

API di basso livello:

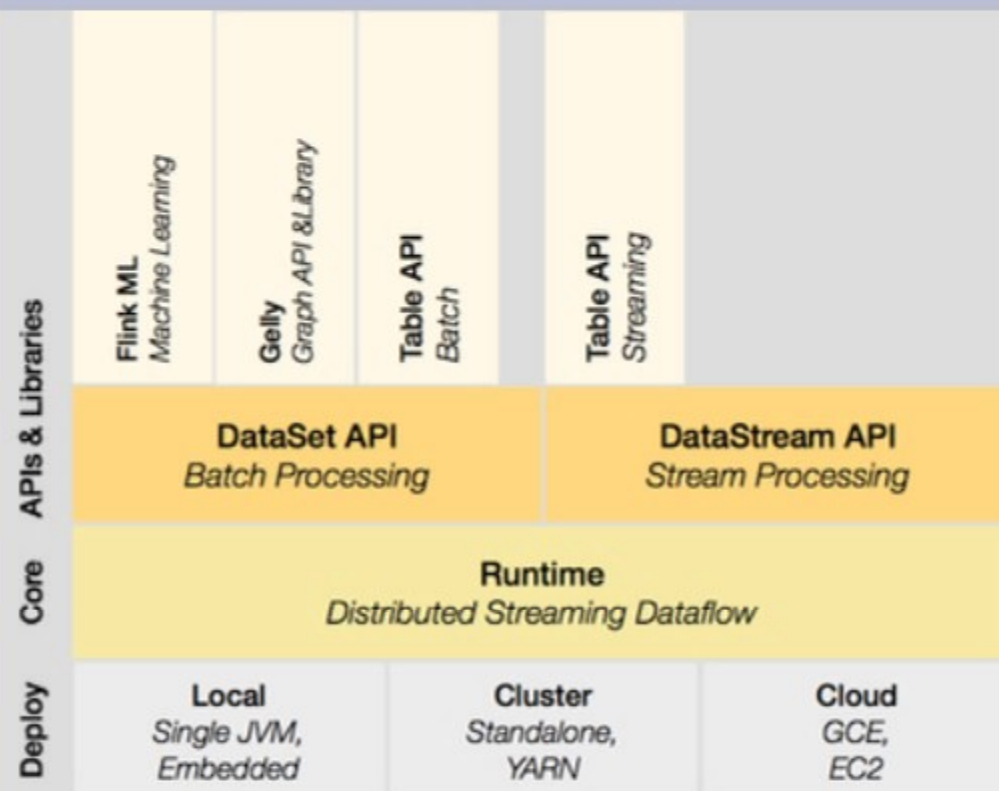
- DataSet API
- DataStream API

Librerie Core:

- Runtime

Esecuzione:

- Locale
- Cluster
- Cloud



Stack

Flink (Stack)



Livelli di astrazione:

API di alto livello:

- Flink ML
- Gelly
- Table API

API di basso livello:

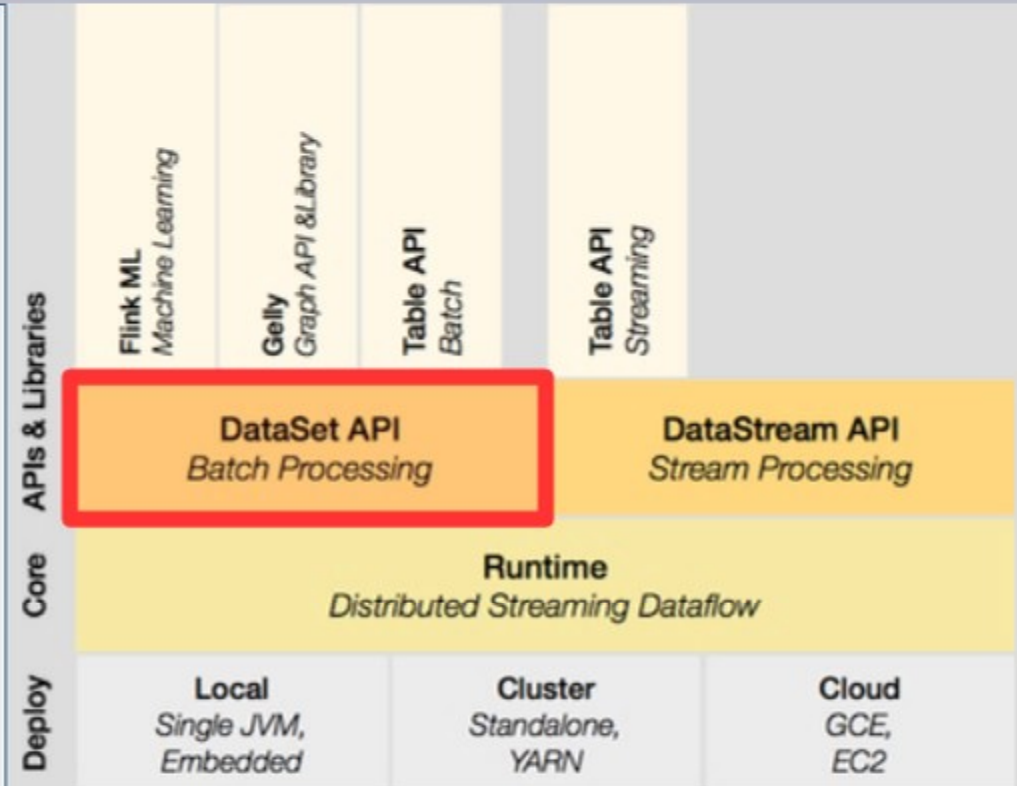
- **DataSet API**
- DataStream API

Librerie Core:

- Runtime

Esecuzione:

- Locale
- Cluster
- Cloud

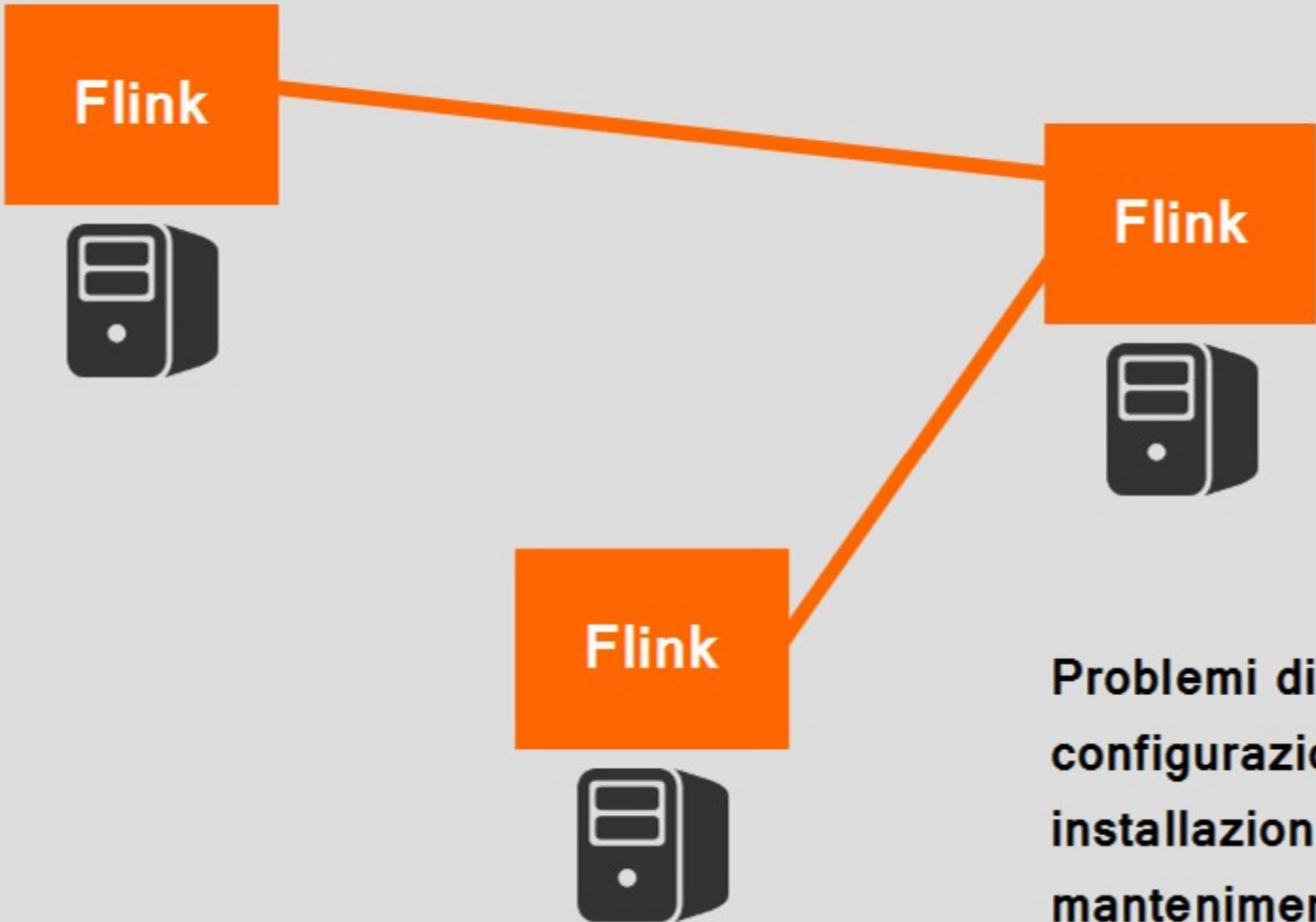


Stack

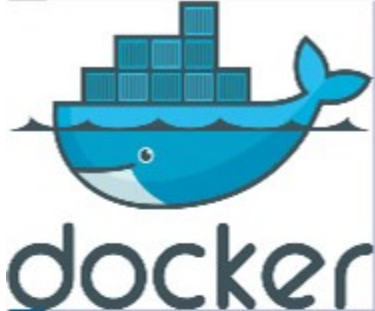
Full Outer Join ad N vie



Flink (Cluster)



**Problemi di
configurazione,
installazione e
mantenimento**



Flink con Docker

Flink

Docker



Flink

Docker



Flink

Docker

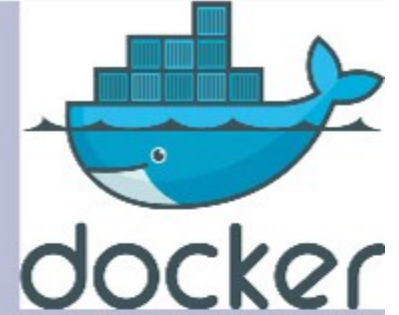


Funzioni di
virtualizzazione di
Linux

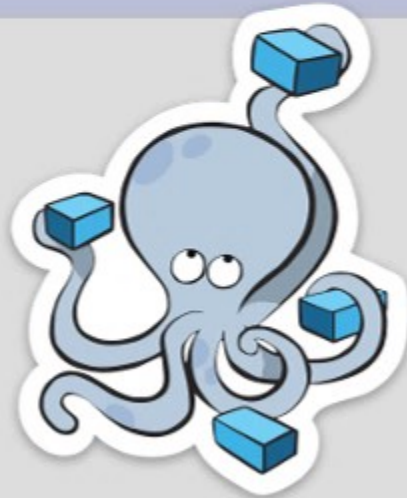
Docker: piattaforma per la
facile creazione e gestione
di macchine virtuali

Servizi aggiuntivi
messi a disposizione
da Docker

Flink con Docker



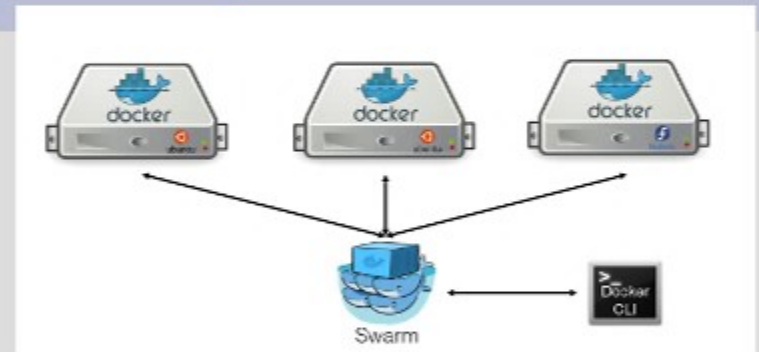
Toolbox



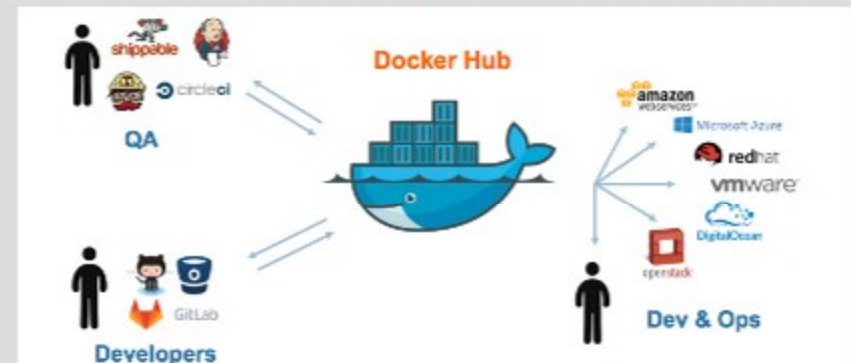
Compose



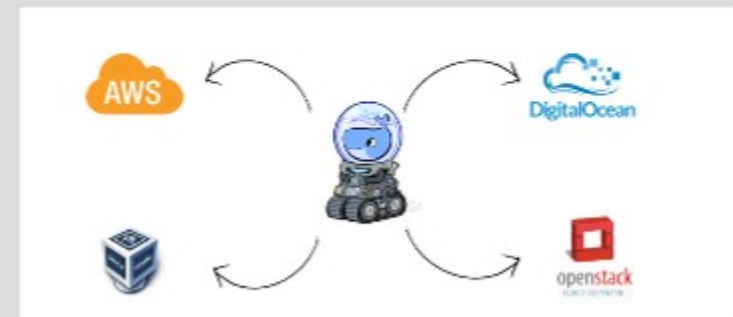
Trusted
Registry



Swarm



Hub



Machine



Flink Full Outer Join

- Prima della versione 0.10.2 Flink non implementava nessun Outer Join
- Dalla 0.10.2 in poi per le operazioni batch gli Outer Join esistono ma solo a due vie

**Prima dell'inizio della mia
tesi non c'era nessun
Outer Join**



**FULL
OUTER
JOIN**



Flink Full Outer Join

Se si vuole utilizzare l'Outer Join di Flink in Momis:

- Per ogni diversa coppia di input diversi bisogna scrivere una diversa classe.
- Per eseguire un FullOuterJoin ad N vie servono $N - 1$ classi.
- Ogni classe Globale richiede la costruzione di un proprio FullOuterJoin.

Flink deve conoscere il tipo di dato in ingresso ed in uscita

L'utente non può costruire $N - 1$ classi temporanee per ogni classe Globale e Locale

il metodo implementato da Flink non può essere utilizzato così com'è da Momis.



Flink Full Outer Join

BASI:

- Flink richiede di conoscere il tipo dei dati in ingresso, ma accetta il tipo Object purchè non debba essere utilizzato per eseguire dei confronti durante le operazioni sui DataSet
- Flink può avere tuple con al massimo 25 campi, ma questi campi possono essere altre tuple
- La maggioranza dei campi sui quali si esegue il join sono trasformabili in stringhe senza perdita di univocità



Flink Full Outer Join

L'utente deve
innanzitutto
normalizzare il
formato dei
DataSet in
ingresso

AAAA
AAAA
AAAA

BBB
BBB

CC
CC
CC

map



AAAA	
AAAA	
AAAA	

BBB	
BBB	

CC	
CC	
CC	

key = 1, 1, 1

Quadrati dello stesso colore
indicano lo stesso
valore di chiave



Flink Full Outer Join

Red	AAAA	Purple
Blue	AAAA	Purple
Green	AAAA	Purple

Red	BBB	Purple
Blue	BBB	Purple

Red	CC	Purple
Green	CC	Purple
Yellow	CC	Purple

key = 1, 1, 1

**cross
union**



Red	1	Red	AAAA	Purple
Blue	1	Blue	AAAA	Purple
Green	1	Green	AAAA	Purple
Red	2	Red	BBB	Purple
Blue	2	Blue	BBB	Purple
Red	3	Red	CC	Purple
Green	3	Green	CC	Purple
Yellow	3	Yellow	CC	Purple

La classe estrae le meta-informazioni e riunisce i dati un unico DataSet

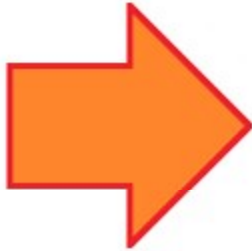
Meta:
Chiave e Sorgente



Flink Full Outer Join

1	AAAA	
1	AAAA	
1	AAAA	
2	BBB	
2	BBB	
3	CC	
3	CC	
3	CC	

groupBy

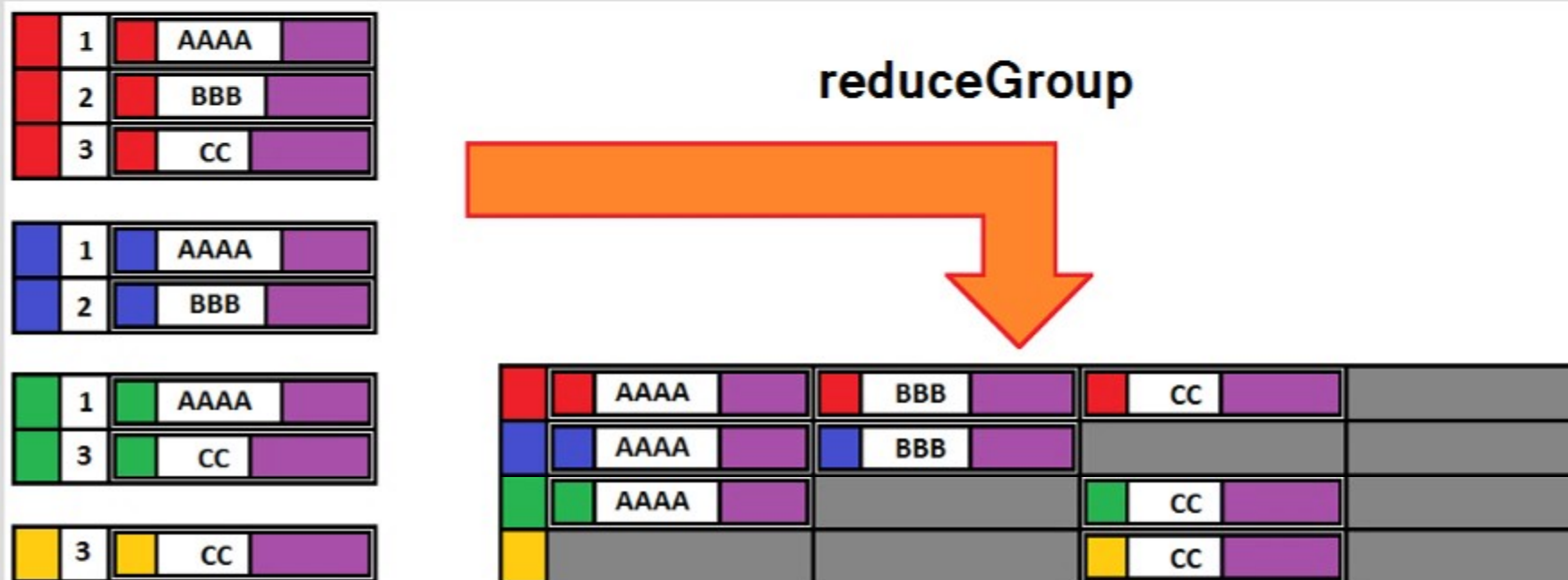


1	AAAA	
2	BBB	
3	CC	
1	AAAA	
2	BBB	
1	AAAA	
3	CC	
3	CC	

Si raggruppano i dati in base alla chiave: tutte le tuple riferite ad un singolo oggetto reale sono raggruppate



Flink Full Outer Join



Date tutte le tuple di un oggetto si crea una singola tupla contenente tutte le tuple provenienti dalle sorgenti: possono esserci al massimo 25 sorgenti quindi la tupla finale deve poter contenere 25 sotto-tuple



Flink Full Outer Join

Red	Red	AAAA	Purple	Red	BBB	Purple	Red	CC	Purple	Grey
Blue	Blue	AAAA	Purple	Blue	BBB	Purple	Grey	Grey	Grey	Grey
Green	Green	AAAA	Purple	Grey	Grey	Grey	Green	CC	Purple	Grey
Yellow	Grey	Grey	Grey	Grey	Grey	Grey	Yellow	CC	Purple	Grey

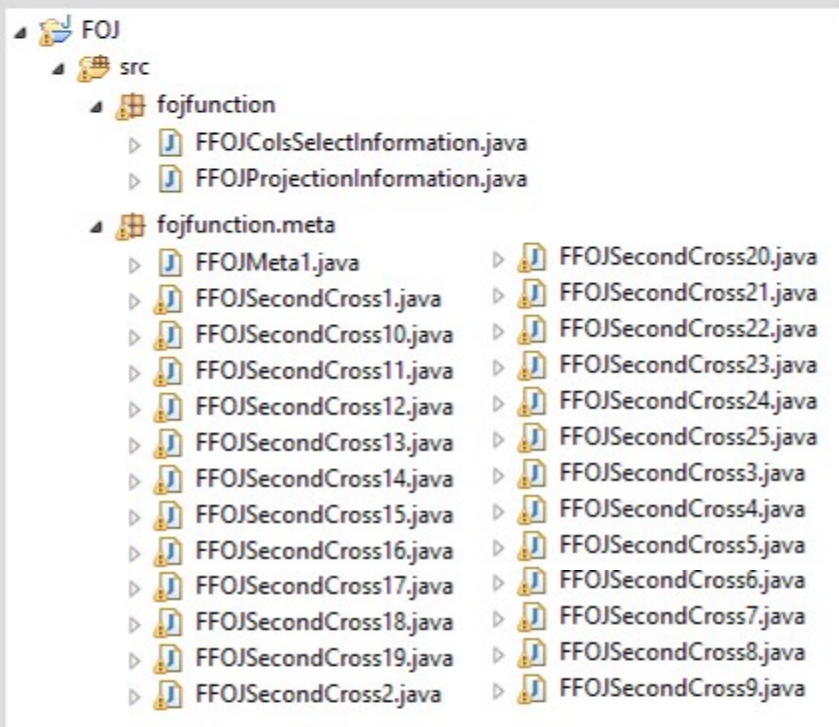
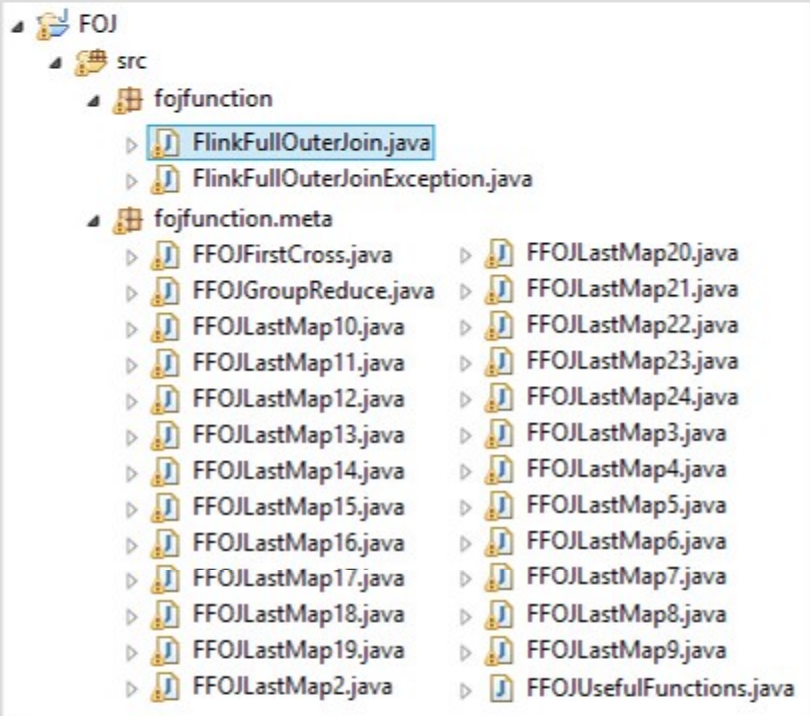


Red	Red	AAAA	Purple	Red	BBB	Purple	Red	CC	Purple	Grey
Blue	Blue	AAAA	Purple	Blue	BBB	Purple	Grey	Grey	Grey	Grey
Green	Green	AAAA	Purple	Grey	Grey	Grey	Green	CC	Purple	Grey
Yellow	Grey	Grey	Grey	Grey	Grey	Grey	Yellow	CC	Purple	Grey

Le vie in eccesso
(25 - N) sono
eliminate. Questa
operazione non è
necessaria per la
creazione di un
corretto Full Outer
Join



Flink Full Outer Join (implementazione)



È possibile inoltre inserire la specifica della select list di output



Flink Full Outer Join (implementazione)

```
final ExecutionEnvironment AMBIENTE = ExecutionEnvironment.getExecutionEnvironment();
```

```
DataSet</*Tupla di 25 Oggetti*/>
```

```
    SORGENTE_1 = [...],
```

```
    SORGENTE_2 = [...],
```

```
    SORGENTE_3 = [...],
```

```
    [...]
```

```
FlinkFullOuterJoin FULL_OUTER_JOIN = new FlinkFullOuterJoin(AMBIENTE);
```

```
FULL_OUTER_JOIN.setInput(0, SORGENTE_1);
```

```
FULL_OUTER_JOIN.setInput(1, SORGENTE_2);
```

```
FULL_OUTER_JOIN.setInput(2, SORGENTE_3);
```

```
[...]
```

```
Vector<Integer> VETTORE_DI_CHIAVI = new Vector<Integer>();
```

```
/*Settare le posizioni delle chiavi*/
```

```
FULL_OUTER_JOIN.setKeys(VETTORE_DI_CHIAVI);
```

```
DataSet<Tuple2<String, /*Tupla di N Tuple di 25 Oggetti*/>>
```

```
    RITORNO = FULL_OUTER_JOIN.get_nx25_Foj();
```

```
AMBIENTE.execute();
```



- Integrazione tecnologica tra Flink, Docker e Momis non ancora messa in atto
- Flink non è l'unico framework per la gestione dei Big Data: es. Spark
- Sviluppo futuro: confronto delle performance tra Flink, Spark e altri framework (il DBGroup ha già utilizzato Spark per l'Entity Deduplication)

Accoppiamento tra Data Integration e piattaforme gestione Big Data per l'estrazione dell'informazione utile

Flink è stato scelto per il suo potente motore stream-based

Grazie per l'attenzione