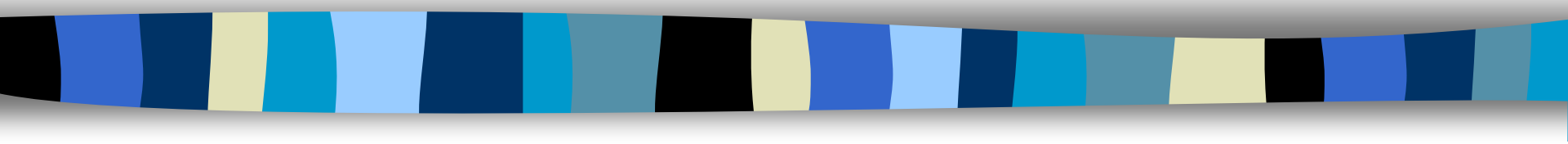
Sistemi Informativi

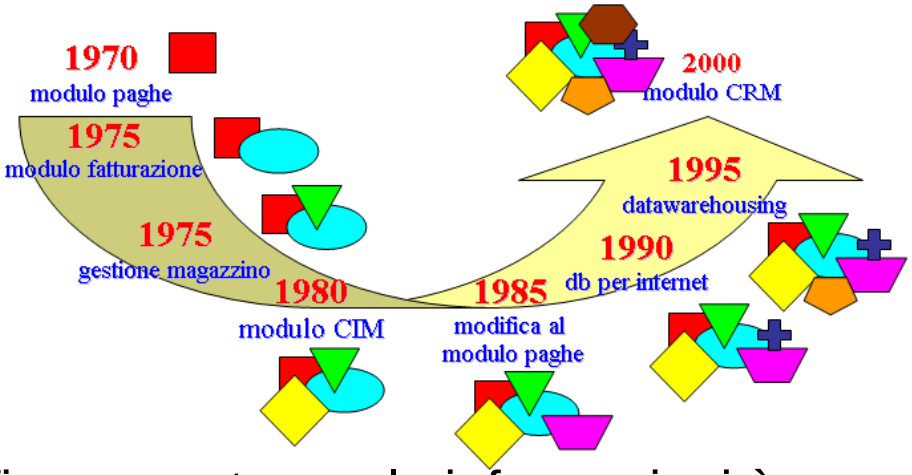
Prof. Matteo Golfarelli
Alma Mater Studiorum - Università di Bologna

Integrazioni di basi di dati



I motivi

- Lo sviluppo di un sistema informativo è di tipo **incrementale** ed **evolutivo** e quindi il suo risultato è un mosaico composto da più frammenti ognuno dei quali utilizza la tecnologia in voga al momento della sua realizzazione e che modella la realtà esistente in quel momento



- Per operare efficacemente con le informazioni è necessario che:
  - Il modello della realtà utilizzato sia consistente e veritiero (problema del *divario percettivo*).
  - I dati siano consistenti e corretti.
- Il raggiungimento di questo ambizioso risultato necessita di un processo di ***integrazione***, ***pulizia*** e ***trasformazione*** dei dati che può richiedere un elevato dispendio di tempo e di risorse.

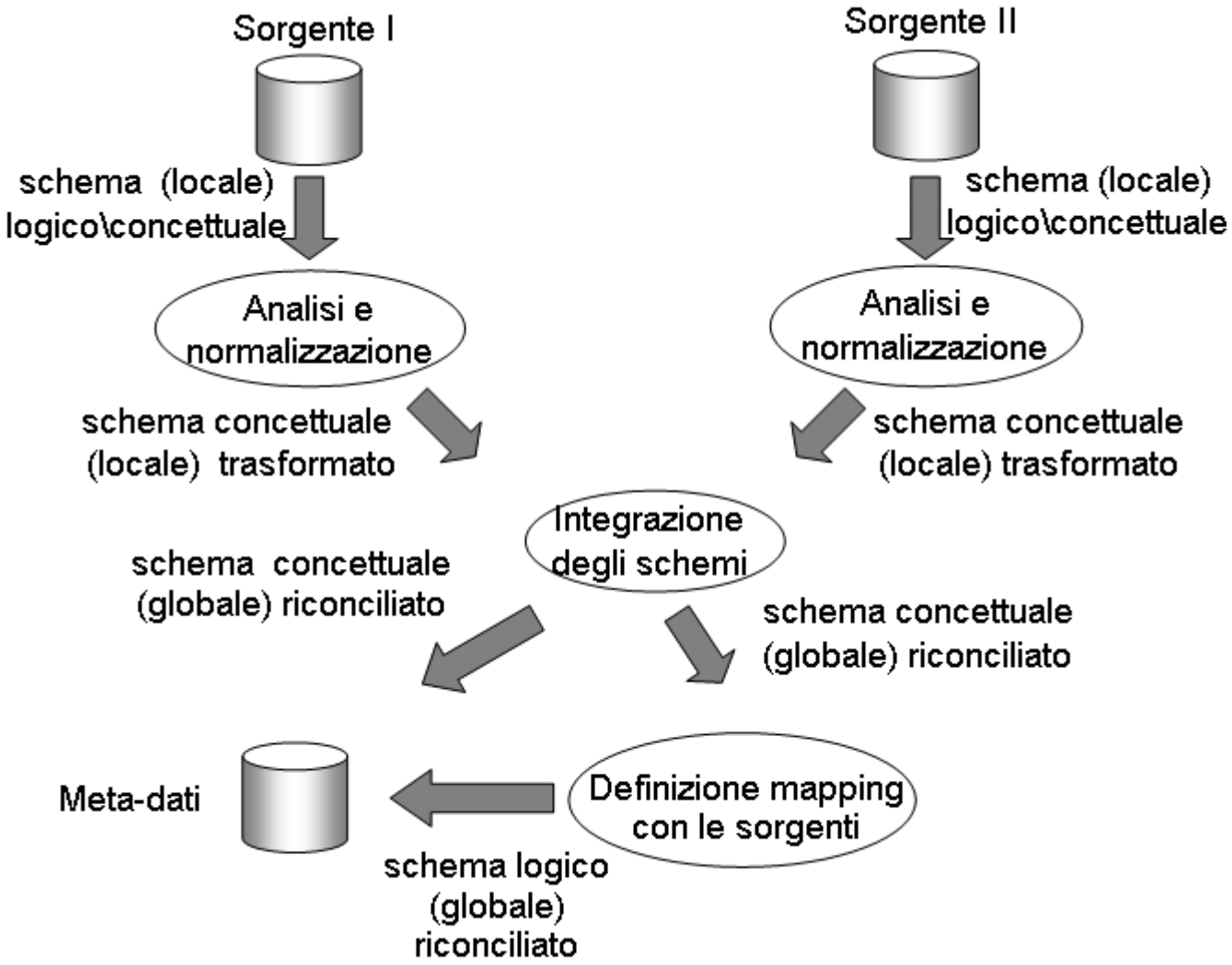
Aspetti intensionali ed estensionali

- Con il termine di **integrazione** si indicano l'insieme di attività atte a costruire una versione integrata e consistente del sistema informatico.
  - **Integrazione dello schema:** operazione a livello intensionale che rende consistenti gli schemi dei diversi moduli.
  - **Trasformazione dei dati:** operazione a livello estensionale che trasforma i dati degli schemi locali nei dati dello schema globale.
  - **Pulizia dei dati:** operazione a livello estensionale che controlla ed elimina eventuali inconsistenze ed errori.

I passi dell'integrazione dello schema

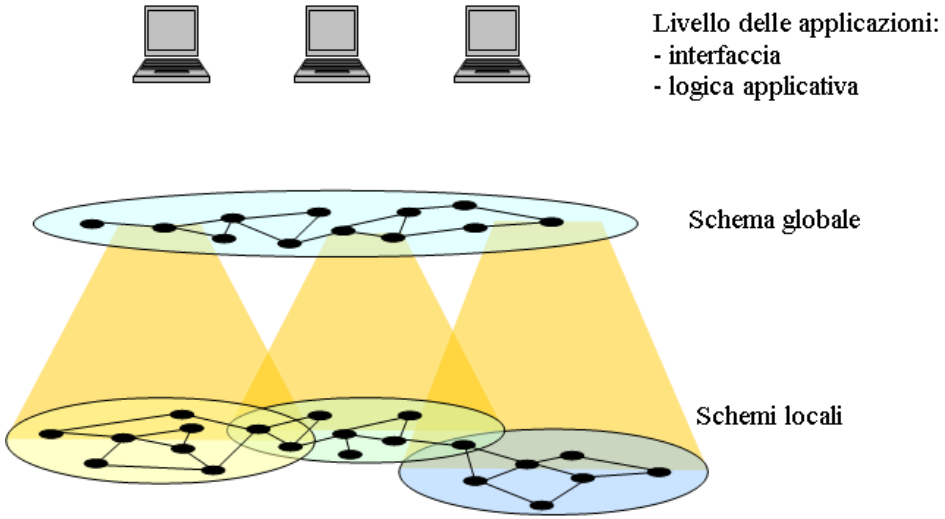
- Analizzando più in dettaglio la fase di integrazione si evidenziano le seguenti operazioni:
  - **Analisi e normalizzazione**: consiste nell'analizzare i diversi schemi locali producendo come risultato un insieme di schemi concettuali localmente completi e consistenti.
  - **Definizione dello schema riconciliato**: è la fase più importante in cui i diversi schemi locali vengono fusi in un unico schema globalmente consistente.
  - **Definizione del mapping**: utilizzando lo schema logico riconciliato si definisce la relazione (mapping) tra concetti degli schemi sorgenti e dello schema riconciliato.

I passi dell'integrazione



Architettura per il SI integrato

- La presenza di un livello di dati integrato modifica l'architettura del SI



- Le applicazioni operano su viste dello schema integrato che può essere:
  - **Virtuale:** viene definita solo la meta conoscenza necessaria a ottenere le informazioni sullo schema globale. Queste saranno create solo quando richieste mediante interrogazioni eseguite sugli schemi locali. Questa soluzione è quella maggiormente utilizzata (es. SI distribuiti, SI confederati, SI eterogenei).
  - **Materializzato:** i dati vengono trasformati e memorizzati in versione duplicata. Questa soluzione viene utilizzata per esempio nei sistemi DW.

Architettura per il SI integrato

- A causa della complessità dell'operazione non è sempre possibile ottenere un unico schema globale; talvolta può essere sufficiente creare poche porzioni integrate a partire da molti schemi locali.

- **Master Data Management (MDM)**: l'integrazione è limitata ai dati aziendali critici

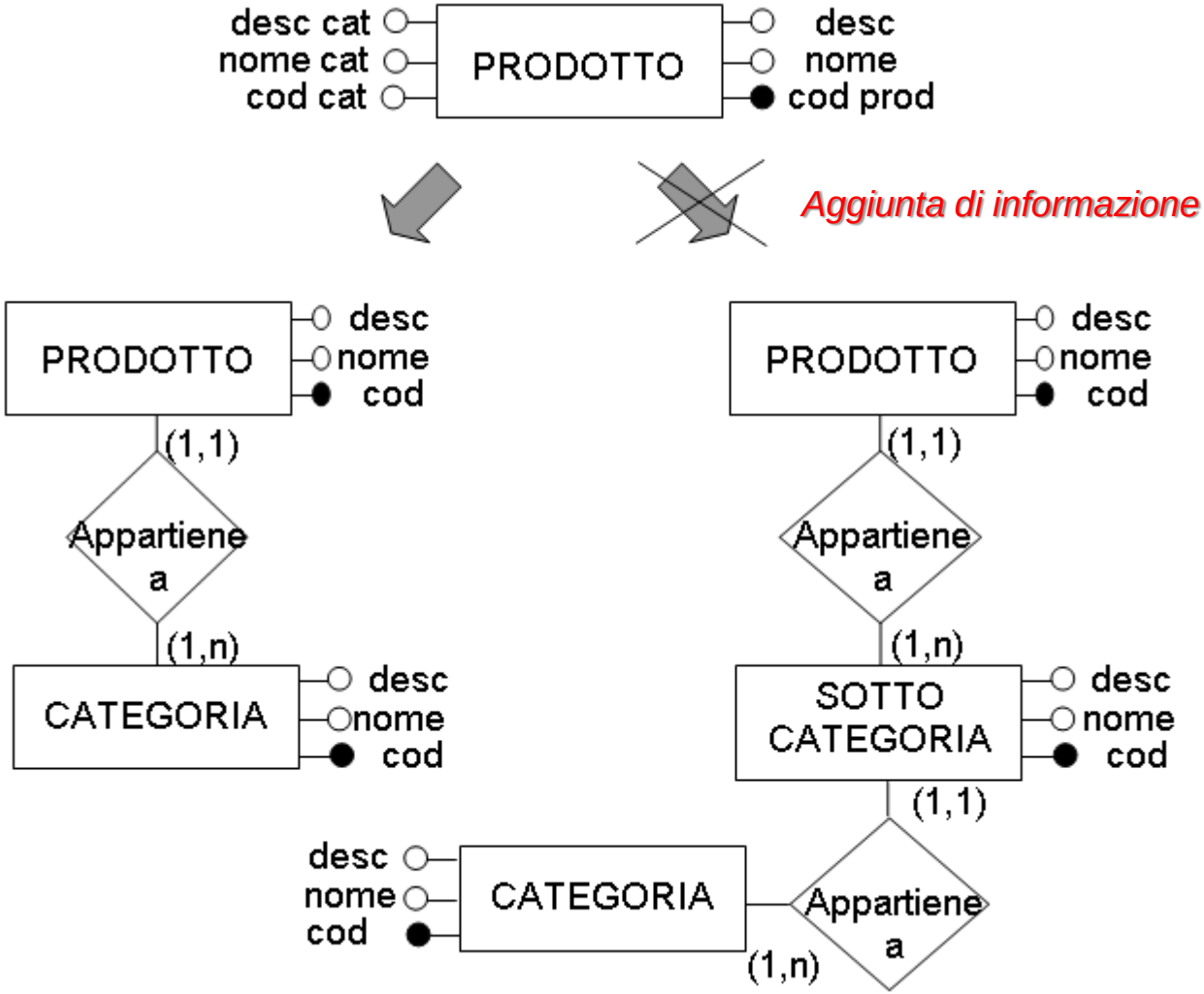
Analisi e normalizzazione

- Durante la fase di analisi il progettista, confrontandosi con gli esperti del dominio applicativo, deve verificare la completezza degli schemi locali, sforzandosi di individuare eventuali correlazioni involontariamente omesse:
  - Esplicitazione di dipendenze funzionali precedentemente tralasciate
  - Individuazione di nuove associazioni tra le entità.

- Le trasformazioni apportate allo schema non devono però introdurre nuovi concetti (assenti nei dati presenti sulle sorgenti), bensì esplicitare tutti quelli ricavabili dai dati memorizzati nelle sorgenti operazionali.

- Lo schema concettuale delle sorgenti rappresenta senz'altro il risultato principale dell'analisi, e deve essere espresso mediante lo stesso formalismo per tutte le sorgenti. Dove assente esso deve essere ricavato per **reverse engineering**.

Analisi e normalizzazione



I problemi da affrontare

- Se le diverse sorgenti dati modellassero porzioni indipendenti e distinte del mondo reale, il problema dell'integrazione non sussisterebbe.
- Ciò non può avvenire in pratica poiché ogni azienda è un universo coeso in cui le diverse unità organizzative partecipano a processi comuni che condividono tutti o parte degli attori.

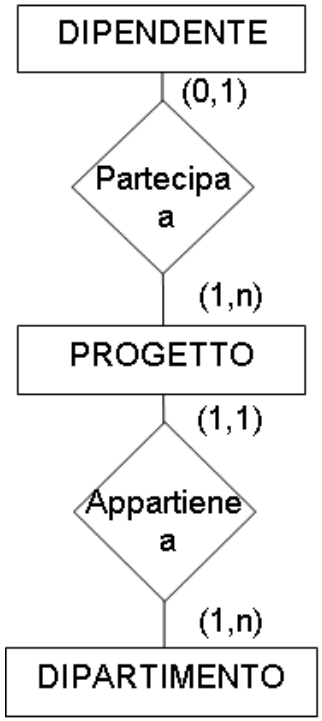
Le diversità nelle rappresentazioni dei concetti comuni sono causate da:

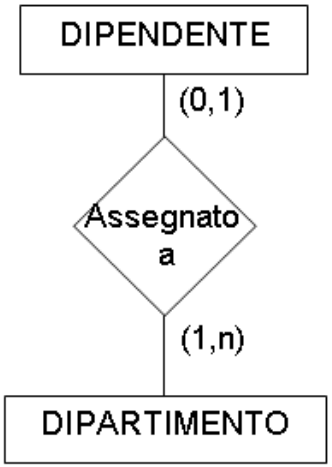
- Diversità di prospettiva.
- Equivalenza dei costrutti del modello.
- Incompatibilità delle specifiche.

- La fase di integrazione non si deve limitare a evidenziare le differenze di rappresentazione dei concetti comuni a più schemi, ma deve anche identificare l'insieme di concetti distinti e memorizzati in schemi differenti che sono correlati attraverso proprietà semantiche (***proprietà inter-schema***).

Diversità di prospettiva

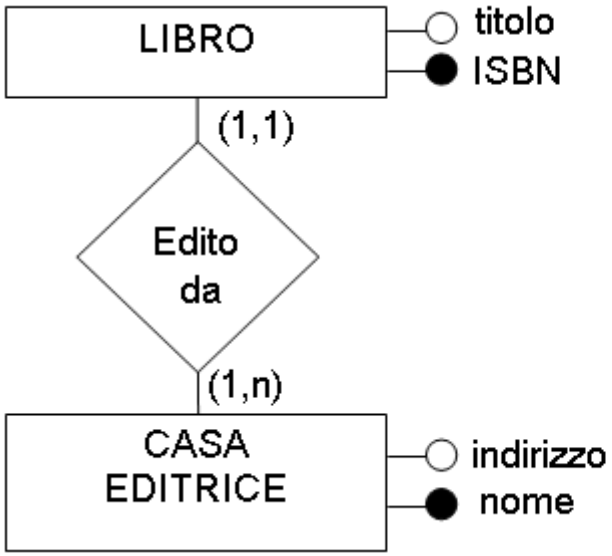
- Il punto di vista rispetto al quale diversi gruppi di utenti vedono uno stesso oggetto del dominio applicativo può differenziarsi notevolmente in base agli aspetti rilevanti ai fini della funzione a cui essi sono preposti.
  - Quello di destra è una porzione dello schema utilizzato per gestire l'organigramma aziendale per il quale non è rilevante la distribuzione dei dipendenti sui vari progetti.

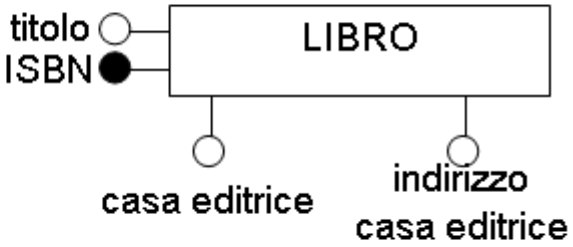




Equivalenza dei costrutti del modello

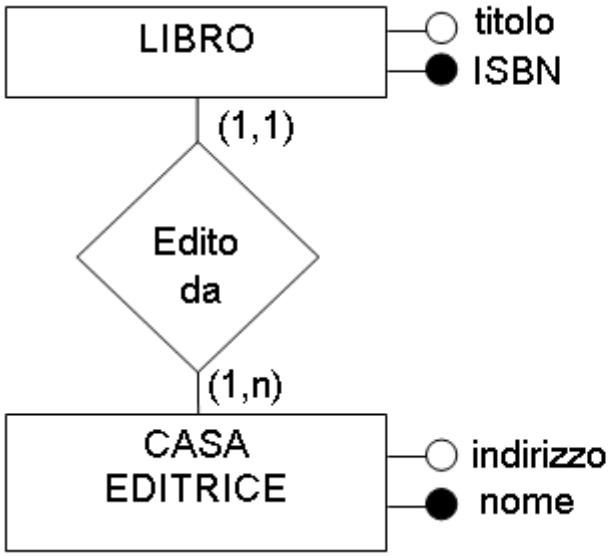
- I formalismi di modellazione permettono di rappresentare uno stesso concetto utilizzando combinazioni diverse dei costrutti a disposizione.
  - La diversità è puramente di tipo sintattico: le due modellazioni sono perfettamente equivalenti.

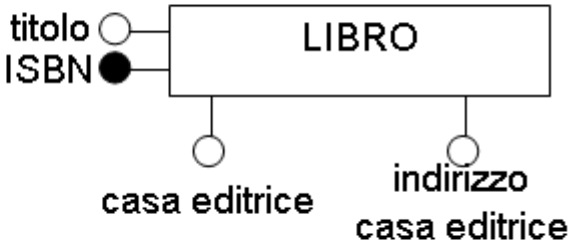




Equivalenza dei costrutti del modello

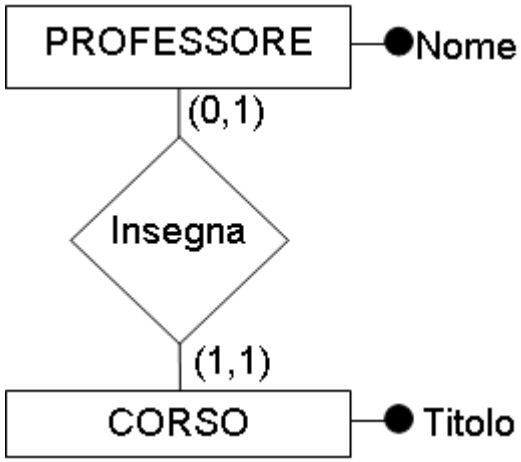
- I formalismi di modellazione permettono di rappresentare uno stesso concetto utilizzando combinazioni diverse dei costrutti a disposizione.
  - La diversità è puramente di tipo sintattico: le due modellazioni sono perfettamente equivalenti.

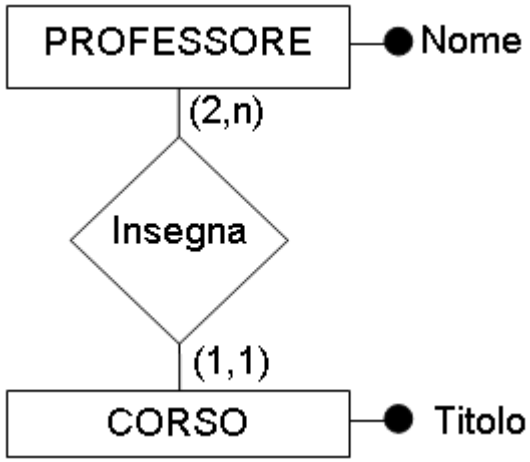




Incompatibilità delle specifiche

- Si verifica quando due schemi modellano una stessa porzione del dominio applicativo e racchiudono concetti diversi, in contrasto tra loro. Tale diversità deriva normalmente da errate scelte progettuali che possono coinvolgere per esempio la scelta dei nomi, dei tipi di dati e dei vincoli di integrità.
  - I due schemi sono incompatibili poiché quello in quello di sinistra un professore può tenere al più un corso mentre in quello di destra deve tenerne almeno due.



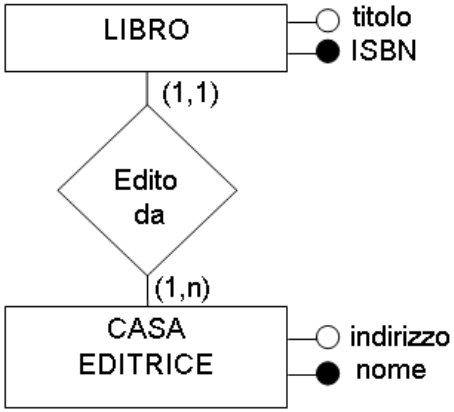


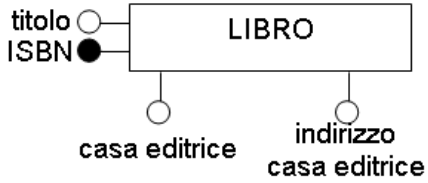
Concetti comuni

- È necessario definire il tipo di relazione semantica esistente tra concetti comuni modellati diversamente in schemi distinti. Quattro sono le possibili relazioni esistenti tra due distinte rappresentazioni R_1 e R_2 di uno stesso concetto:
  - ***Identità:*** vengono utilizzati gli stessi costrutti, il concetto è modellato dallo stesso punto di vista e non vengono commessi errori di specifica; in altre parole quando R_1 e R_2 coincidono.
  - ***Equivalenza:*** R_1 e R_2 non sono esattamente le stesse poiché sono stati utilizzati costrutti diversi (ma equivalenti) e non sussistono errori di specifica o diversità di percezione.
  - ***Comparabilità:*** R_1 e R_2 non sono né identici né equivalenti, ma i costrutti utilizzati e i punti di vista dei progettisti non sono in contrasto tra loro.
  - ***Incompatibilità***: R_1 e R_2 sono in conflitto a causa dell'incoerenza nelle specifiche. In altre parole la realtà modellata da R_1 nega la realtà modellata da R_2.
- A esclusione della situazione di identità, i casi precedenti determinano dei conflitti la cui soluzione rappresenta la componente principale nella fase di integrazione.
-
- **Def.** Si verifica un ***conflitto*** tra due rappresentazioni R_1 e R_2 di uno stesso concetto ogniqualvolta le due rappresentazioni non sono identiche.

Concetti comuni

- **Def.** Due schemi R_1 e R_2 sono ***equivalenti*** se le loro istanze possono essere messe in corrispondenza uno-a-uno.
- L'equivalenza tra schemi implica che a livello estensionale esistono sempre due insiemi di dati, diversi ma equivalenti, che memorizzano le stesse informazioni.





LIBRO

ISBN	titolo	casa editrice
123445	Il DFM	McGrawHill
4354543	Mi Sembra Logico	LettiTutti
4566454	La Giusta Misura	NonSoloLibri
.....	.....	.....

CASA EDITRICE

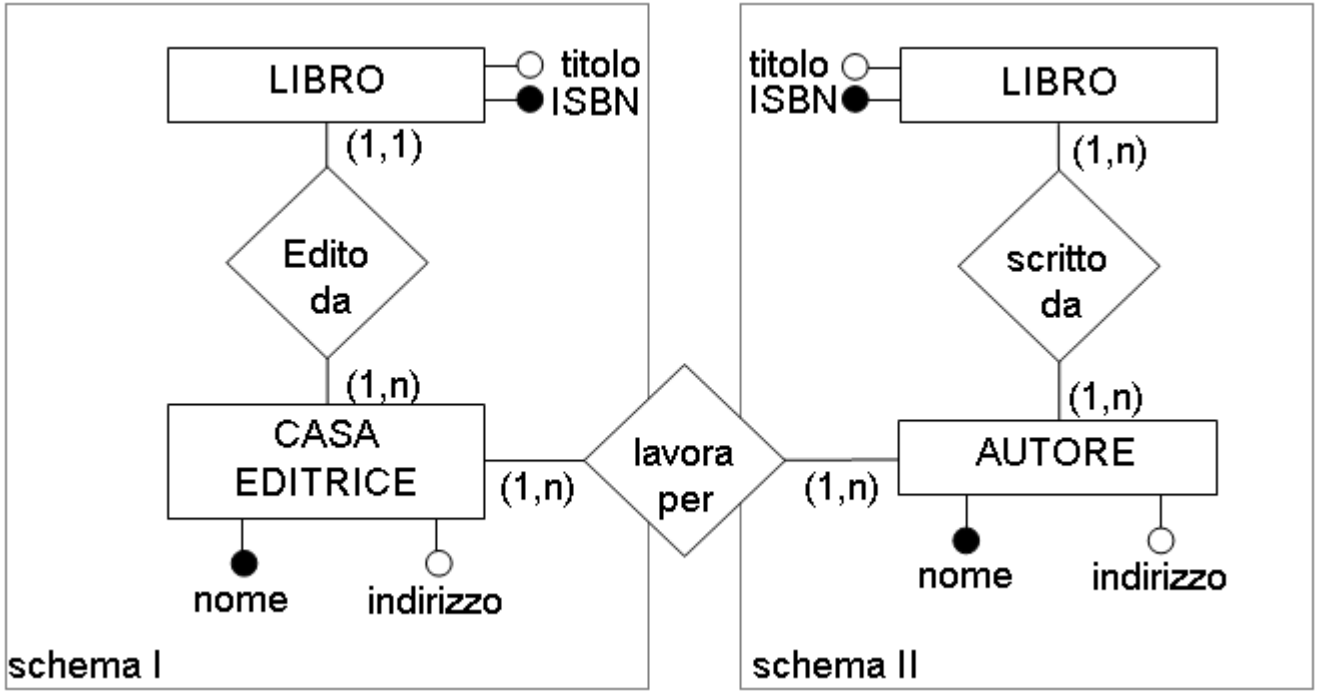
nome	indirizzo
McGraw-Hill	via Ripamonti, 89
LettiTutti	Via dei Brutti, 21
NonSoloLibri	Via Tumedei, 57
.....	.....

LIBRO

ISBN	titolo	nome c.e.	indirizzo c.e.
123445	Il DFM	McGraw-Hill	via Ripamonti, 89
4354543	Mi Sembra Logico	LettiTutti	Via Brutti, 21
4566454	La Giusta Misura	NonSoloLibri	via Rossi, 43
.....	.....	.....	.....

Concetti correlati

- A seguito dell'integrazione, molti concetti diversi ma correlati verranno a trovarsi nello stesso schema dando vita a nuove relazioni che non erano percepibili in precedenza. Tali relazioni sono dette ***proprietà inter-schema*** e devono essere identificate e rappresentate esplicitamente.

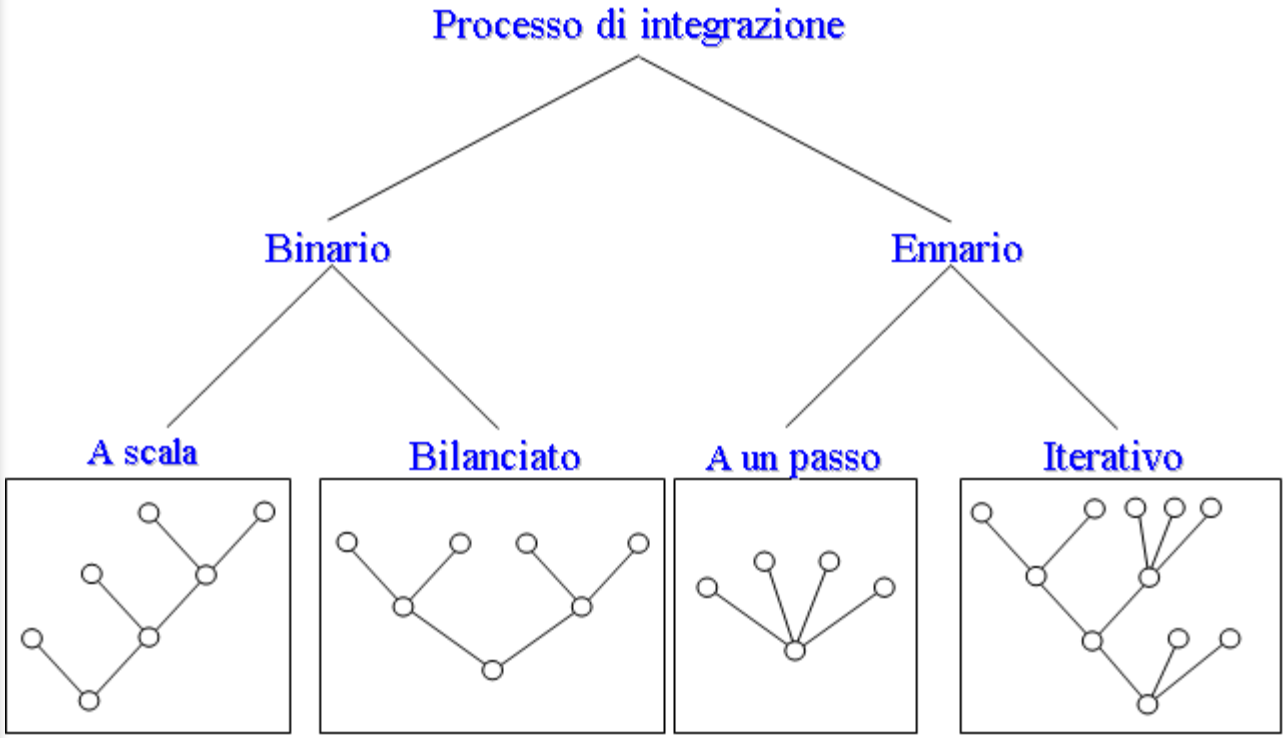


Le fasi dell'integrazione

- Risolvere i problemi fin qui elencati ed evidenziare le proprietà che emergono a seguito dell'integrazione degli schemi locali richiede un complesso insieme di operazioni, per la cui corretta gestione è necessario adottare una metodologia. Le molte metodologie proposte in letteratura concordano sulla sequenza di passi che devono essere svolti e che possono essere così sintetizzati:
  - **Preintegrazione:** comprende la fase di analisi e di scelta della strategia di integrazione.
  - **Comparazione degli schemi:** durante questa fase gli schemi locali vengono confrontati per evidenziare i conflitti e i concetti correlati.
  - **Allineamento degli schemi:** durante questa fase il progettista deve risolvere i conflitti precedentemente individuati.
  - **Fusione e ristrutturazione degli schemi:** gli schemi resi coerenti possono essere fusi a creare un unico schema globale.

Strategie di integrazione

- Indicano l'ordine con cui si procederà all'integrazione degli schemi.
  - ***Tecniche ennarie*:** processo di integrazione considera più di due schemi contemporaneamente
  - ***Tecniche binarie:*** il processo di integrazione considera sempre coppie di schemi. Le tecniche binarie sono dette ***a scala*** quando i nuovi schemi sono integrati allo schema temporaneo ***determinato*** fino a quel momento.



Strategie di integrazione

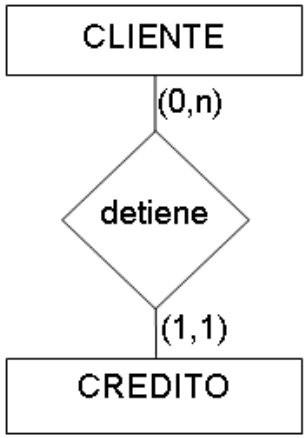
- **L'approccio binario** rende ogni passo di integrazione più semplice grazie al ridotto numero di concetti coinvolti contemporaneamente.
- Con **l'approccio ennario** ogni concetto viene analizzato avendo a disposizione contemporaneamente tutte le informazioni che lo caratterizzano; inoltre con le tecniche ennarie si diminuisce il numero totale di comparazioni tra concetti poiché ognuno di essi viene analizzato una sola volta.
- Utilizzando una **tecnica binaria a scala** è possibile definire l'ordine con cui i diversi schemi sorgenti andranno esaminati e aggiunti allo schema riconciliato; normalmente si preferisce iniziare dalle sorgenti che costituiscono il "cuore" del sistema informativo e la cui integrazione definirà lo scheletro dello schema riconciliato.

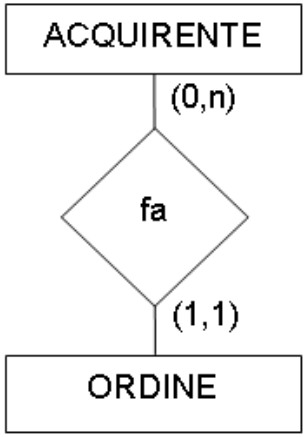
Comparazione degli schemi

- Questa fase consiste in un'analisi comparativa dei diversi schemi che mira a identificare le correlazioni e conflitti tra i concetti in essi espressi. La sua efficacia dipende dalle conoscenze acquisite dal progettista rispetto al dominio applicativo e alla struttura delle sorgenti informative.
- I conflitti che possono essere evidenziati appartengono a 4 categorie:
  - ***Conflitti di eterogeneità***: indicano le discrepanze dovute all'utilizzo di formalismi con diverso potere espressivo negli schemi sorgenti (es. E/R e UML).
  - ***Conflitti sui nomi:*** si verificano a causa delle differenze nelle terminologie utilizzate nei diversi schemi sorgenti:
    - **Omonimie:** lo stesso termine viene utilizzato per denotare due concetti diversi
    - **Sinonimie:** due nomi diversi denotano uno stesso concetto.
- Mentre le omonimie possono essere individuate da una semplice comparazione dei concetti che presentano gli stessi nomi in schemi diversi, per riconoscere eventuali sinonimi è necessaria una conoscenza approfondita del dominio applicativo.

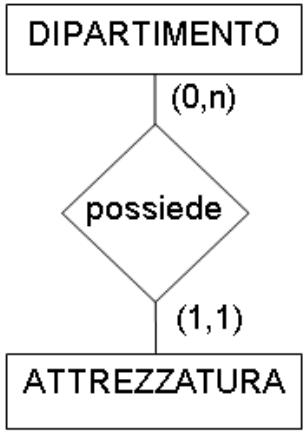
Comparazione degli schemi

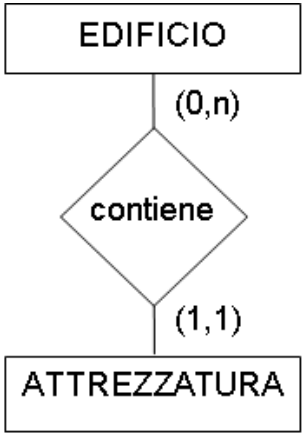
Sinonimie….





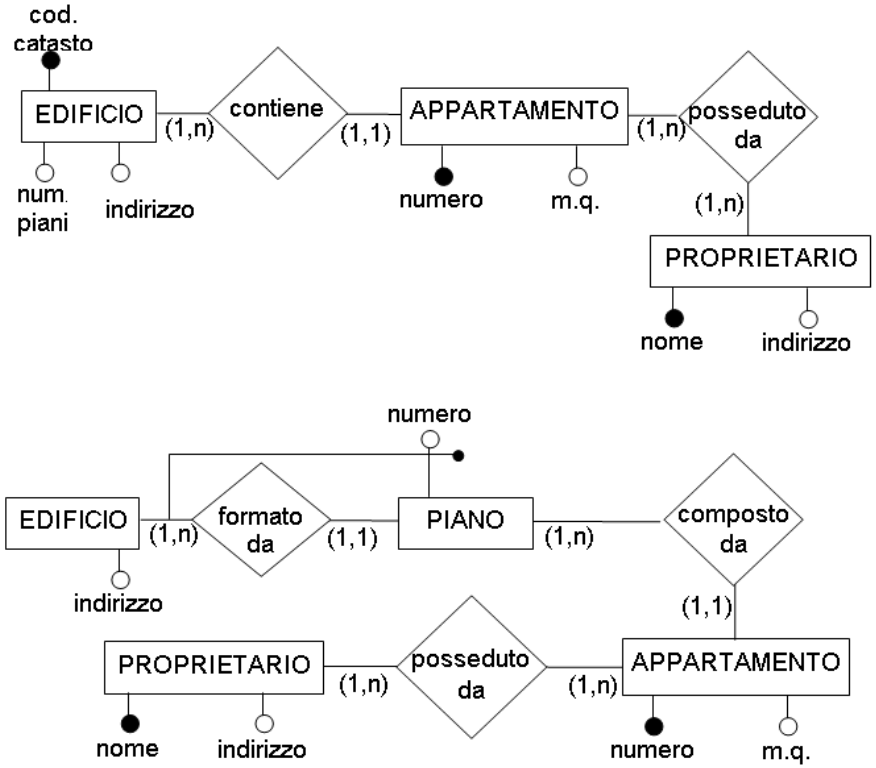
Omonimie….





Comparazione degli schemi

- I ***conflitti semantici*** si verificano quando due schemi sorgenti modellano la stessa porzione di mondo reale a un diverso livello di astrazione e dettaglio.



- Il livello di dettaglio adottato negli schemi locali è diverso e di conseguenza anche l'insieme di informazioni rappresentabili cambia.

Comparazione degli schemi

- I ***conflitti strutturali*** sono causati da scelte diverse nella modellazione di uno stesso concetto, oppure dall'applicazione di differenti vincoli di integrità. Possono essere classificati in quattro categorie.
  - I **conflitti di tipo** si verificano quando uno stesso concetto è modellato utilizzando due costrutti diversi.
  - I **conflitti di dipendenza** che si verificano quando due o più concetti sono correlati con dipendenze diverse in schemi diversi.
  - I **conflitti di chiave** si verificano quando per uno stesso concetto vengono utilizzati identificatori diversi in schemi diversi.
  - I **conflitti di comportamento** si verificano quando diverse politiche di cancellazione/modifica dei dati vengono adottate per uno stesso concetto in schemi diversi.

Allineamento e fusione degli schemi

- Con il termine allineamento degli schemi si intende l'eliminazione dei conflitti in essi presenti tramite l'applicazione di trasformazioni agli schemi sorgenti o allo schema riconciliato temporaneamente definito:
  - Cambio dei nomi
  - Cambio dei tipi degli attributi (intero, float, testo).
  - Modifica delle dipendenze funzionali.
  - Modifica dei vincoli esistenti sugli schemi.

- La soluzione dovrà essere fatta in modo da evitare perdite di informazioni:

intero vs float $\rightarrow$ float

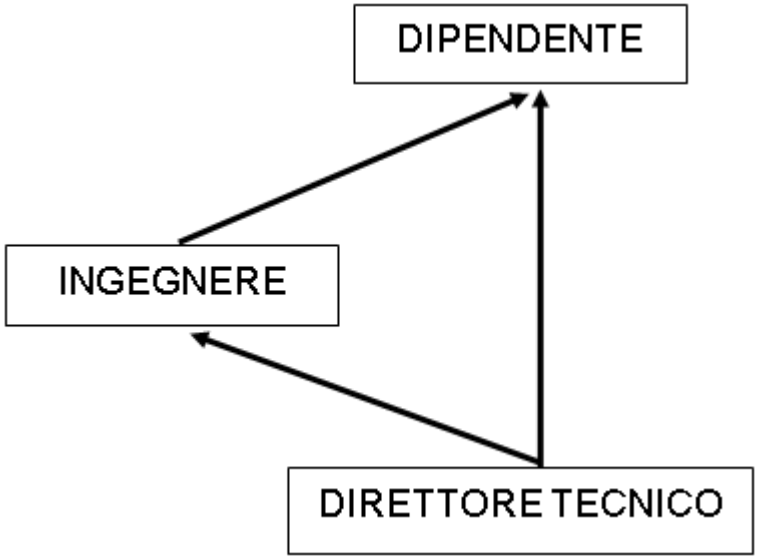
uno-a-uno vs uno-a-molti $\rightarrow$ uno-a-molti

Allineamento e fusione degli schemi

- Attenzione però a non eliminare importanti vincoli del sistema per poter includere situazioni impossibili.
- In caso di incertezza è conveniente preferire le trasformazioni che avvantaggiano gli schemi che si ritengono più centrali nella struttura del futuro schema riconciliato. La strategia migliore è allora quella **binaria a scala**, in cui è possibile iniziare l'integrazione a partire dagli schemi ritenuti più importanti che formeranno il nucleo dello schema riconciliato.
-
- Non sempre i conflitti possono essere risolti poiché derivano da inconsistenze di base del sistema informativo; in questo caso la soluzione deve essere discussa con gli utenti che dovranno fornire indicazioni su quale interpretazione del mondo reale basarsi.

I principi della fusione

- ***Completezza***: dopo la sovrapposizione degli schemi sorgenti risulteranno evidenti ulteriori proprietà inter-schema che non si erano evidenziate in precedenza. Il progettista deve quindi esaminare lo schema fin qui costruito alla ricerca di nuove proprietà che potranno essere esplicitate inserendo nuove associazioni, gerarchie di generalizzazione.
- ***Minimalità***: la sovrapposizione di più schemi può generare una forte ridondanza dei concetti che, sebbene espressi in modo univoco negli schemi sorgenti, risultano duplicati o comunque derivabili l'un l'altro in quello riconciliato. Altre comuni fonti di ridondanza sono le relazioni cicliche tra i concetti e gli attributi derivati

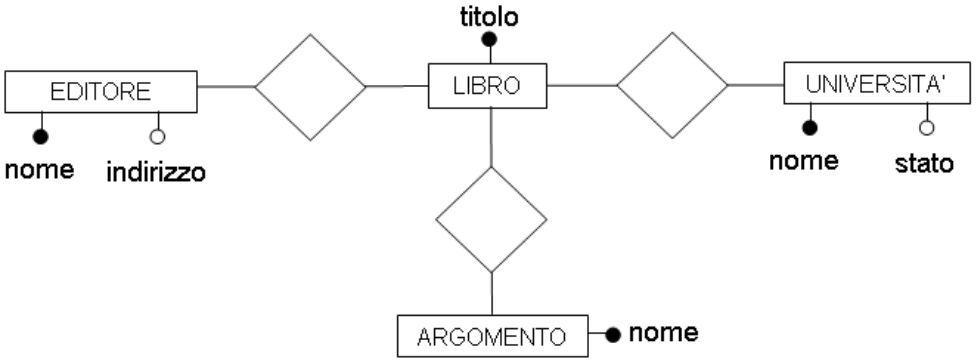


I principi della fusione

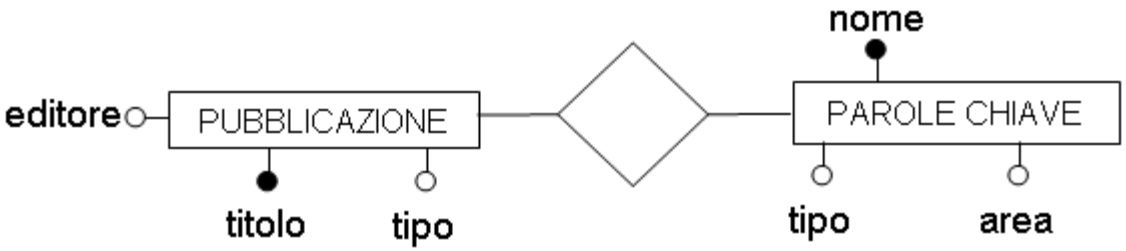
- ***Leggibilità***: il miglioramento della leggibilità dello schema facilita e sveltisce le successive fasi di progettazione. Sebbene sia difficile misurare la differenza di leggibilità tra i due schemi, la nozione qualitativa del termine è relativamente semplice.

Integrazione di schemi: un esempio

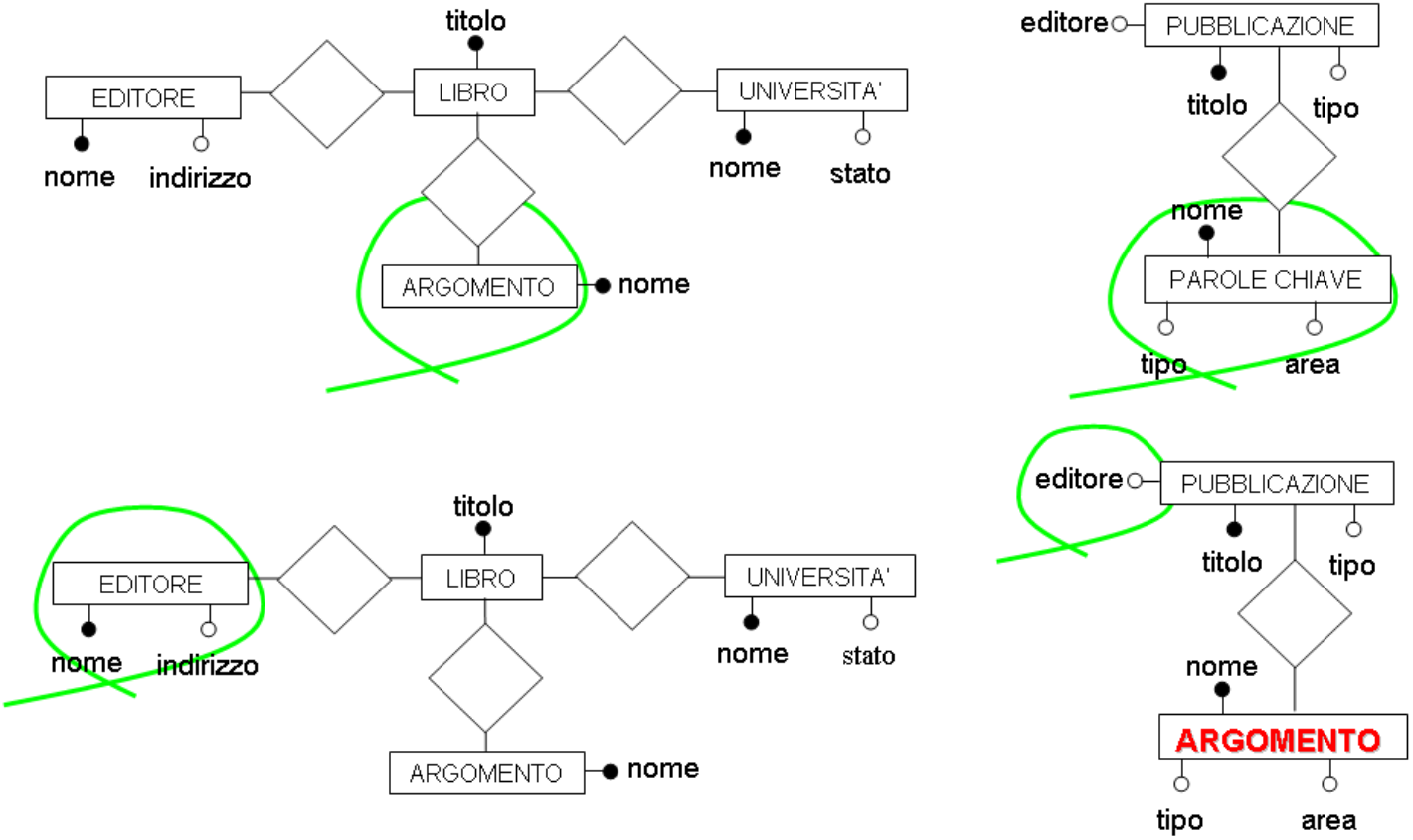
- *".....I dati riguardano i libri. I libri hanno un titolo. Sono pubblicati da editori che hanno un nome e un indirizzo. I libri sono adottati da Università che hanno un nome e appartengono a uno Stato. Ogni libro è relativo ad alcuni argomenti....."*



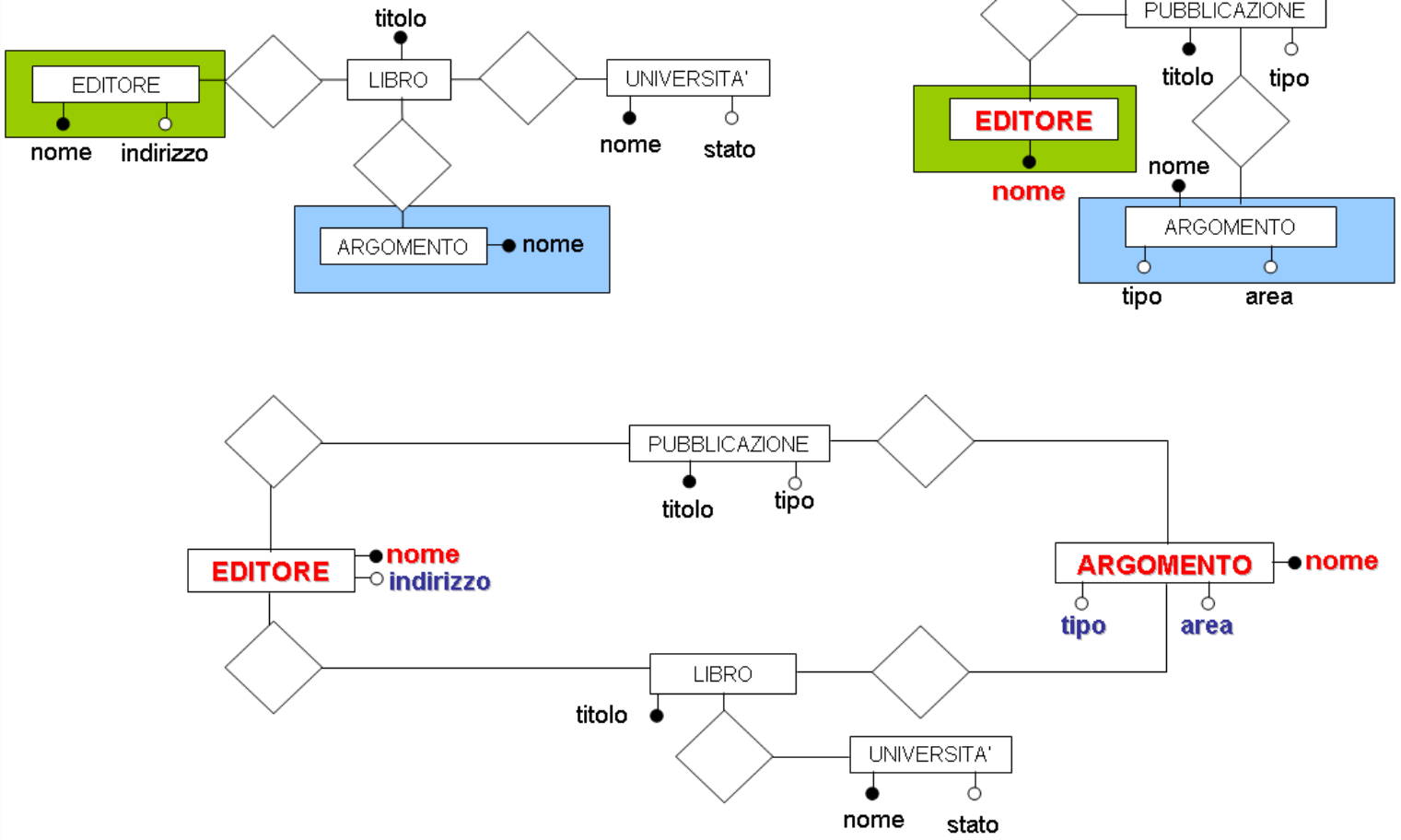
- *".....I dati riguardano delle pubblicazioni di diversi tipi. Ogni pubblicazione ha un titolo un editore e una lista di parole chiave. Ogni parola chiave è composta da un nome, un codice e un'area di ricerca....."*



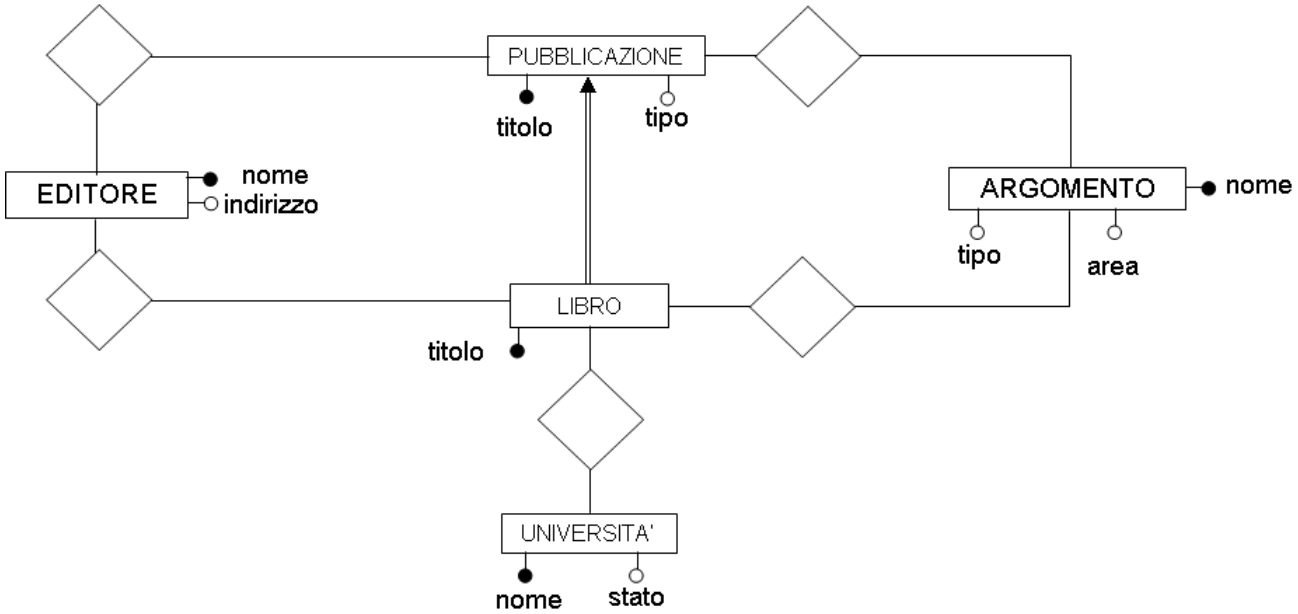
Integrazione di schemi: un esempio



Integrazione di schemi: un esempio



Integrazione di schemi: un esempio



Il mapping

- Il risultato dell'analisi delle sorgenti operazionali finalizzata all'integrazione è composto da due elementi: lo schema riconciliato e il **mapping** ossia l'insieme di corrispondenze tra gli elementi presenti negli schemi sorgenti e quelli dello schema destinazione.

- Le funzioni di mapping vengono utilizzate ogniqualvolta si debba eseguire sui database locali una interrogazione espressa sullo schema riconciliato.

- ***GAV (Global-As-View):*** a ogni concetto dello schema globale deve essere associata una vista il cui significato è definito in base ai concetti che risiedono sugli schemi sorgenti.
- Riduce l'estensibilità dello schema riconciliato poiché l'aggiunta di una nuova sorgente richiederà la modifica di tutti i concetti dello schema globale che la utilizzano.
- Facilita la modalità di definizione delle interrogazioni poiché per capire quali concetti degli schemi sorgenti sono coinvolti sarà sufficiente sostituire a ogni concetto dello schema globale la definizione della vista che lo definisce rispetto ai concetti sugli schemi locali (*unfolding*).

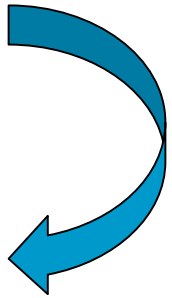
Il mapping

- **LAV (*Local-As-View*):** lo schema globale è espresso indipendentemente dalle sorgenti, i cui concetti saranno invece definiti come viste sullo schema globale
- Richiede trasformazioni complesse (*query rewriting*) per capire quali elementi degli schemi sorgenti devono essere presi in considerazione per ricreare il concetto espresso nello schema globale.
- Favorisce l'estensibilità dello schema riconciliato e la sua manutenzione, per esempio l'aggiunta di una nuova sorgente al sistema richiederebbe solo la definizione della vista sullo schema globale che non verrebbe quindi necessariamente modificato

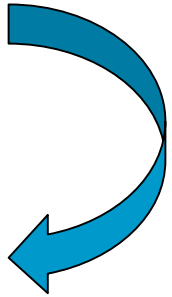
GAV: un esempio

// DB1 Magazzino
ORDINI2001(chiaveO, chiaveC, data ordine, impiegato)
CLIENTE(chiaveC, nome, indirizzo, città, regione, stato)
.....

// DB2 Amministrazione
CLIENTE(chiaveC, piva, nome, tel, fatturato)
FATTURE(chiaveF, data, chiaveC, importo, iva)
STORICOORDINI2000(chiaveO, chiaveC, data ordine, impiegato)
.....



```
CRETE VIEW CLIENTE AS
SELECT    CL1.chiaveC, CL1.nome, CL1.indirizzo, CL1.città,
                    CL1.regione,          CL1.stato, CL2.tel, CL2.fatturato
FROM                DB1.CLIENTE AS CL1, DB2.CLIENTE AS CL2
WHERE     CL1. chiaveC = CL2. chiaveC;
```



```
CREATE VIEW ORDINI AS
SELECT * FROM DB1.ORDINI2001
UNION
SELECT * FROM DB2.STORICOORDINI2000;
```

LAV: un esempio

// DB Riconciliato
ORDINI(chiaveO, chiaveC, data ordine, impiegato)
CLIENTE(chiaveC, piva, nome, indirizzo, città, regione, stato, tel,
fatturato)
.....

// DB1 Magazzino
CRETE VIEW CLIENTE AS
SELECT chiaveC, nome, indirizzo, città, regione, stato
FROM DB.CLIENTE;

CREATE VIEW ORDINI2001 AS
SELECT * FROM DB.ORDINI
WHERE data > '31/12/2000' and data > ''1/1/2002''

La definizione delle sorgenti locali è semplice ma come faccio a esprimere le interrogazioni dallo schema globale a quello locale ?

Pulizia dei dati

- Con il termine **pulizia dei dati** (*Data Cleaning* o *Data Cleansing* o *Data Scrubbing*) si intende l'insieme delle operazioni atte a garantire la consistenza e la correttezza dei dati presenti nel livello riconciliato.

- Le principali cause di inconsistenza nei dati sono le seguenti:
  - **Errori di battitura.** Sono sempre possibili se al momento dell'inserimento non sono previste procedure di controllo dei valori inseriti (per esempio, il controllo di correttezza del codice fiscale).
  - **Differenza di formato dei dati dello stesso campo.** Si verifica ogniqualvolta l'informazione memorizzata in un campo non sia rigidamente strutturata.

    "I", "IT", "Italia"

    "The Coca-Cola Company", "Coca Cola", "Coca-Cola Co."
  - **Inconsistenza tra valori e descrizione dei campi.** Si verifica a causa dell'evoluzione del modo di operare in azienda o dal variare delle abitudini di vita che rendono indispensabile una informazione di cui non era prevista la memorizzazione.

Pulizia dei dati

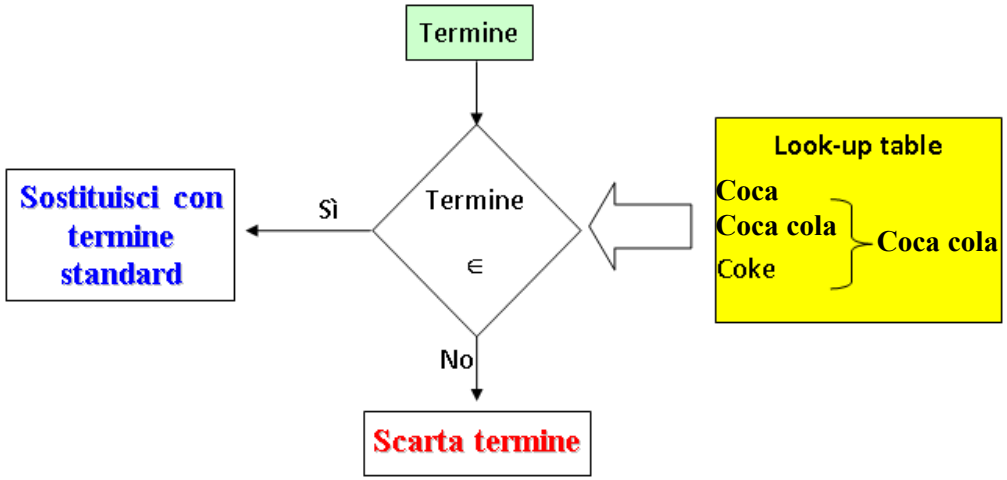
- **Inconsistenza tra valori di campi correlati.** Si verifica quando i valori presenti in due o più campi sono singolarmente corretti ma tra loro inconsistenti.

città = 'Bologna', regione = 'Lazio‘

- **Informazioni mancanti.** Quando le applicazioni del sistema operazionale consentono di non riempire alcuni dei campi, la fretta e lo scarso interesse verso un dato spingono spesso l’operatore a tralasciarne l’inserimento.
- **Informazioni duplicate.** Si presentano quando due sorgenti forniscono la stessa informazione ma i record utilizzano chiavi diverse.

Tecniche basate su dizionari

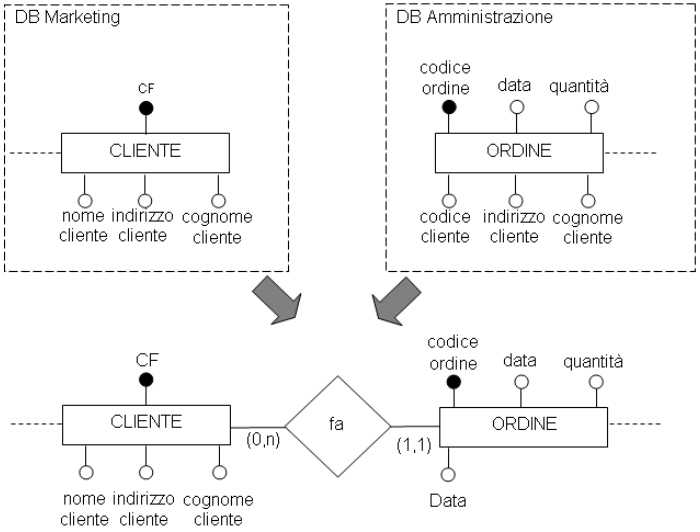
- Con questo termine si indicano le tecniche di verifica della correttezza dei valori di un campo basate su tabelle di riferimento (*look-up table*) e dizionari per la ricerca di abbreviazioni e sinonimi.
- Sono applicabili solo quando il dominio del campo è conosciuto e limitato.



- Le tecniche basate su dizionari possono essere applicate anche a più campi contemporaneamente al fine di verificare eventuali inconsistenze tra due campi correlati. Per esempio, avendo a disposizione una tabella di riferimento che riporta le regioni italiane e, per ognuna di esse, l'elenco delle città che vi appartengono, è possibile verificare se i campi città e regione di un indirizzo sono tra loro consistenti.

Tecniche basate su dizionari

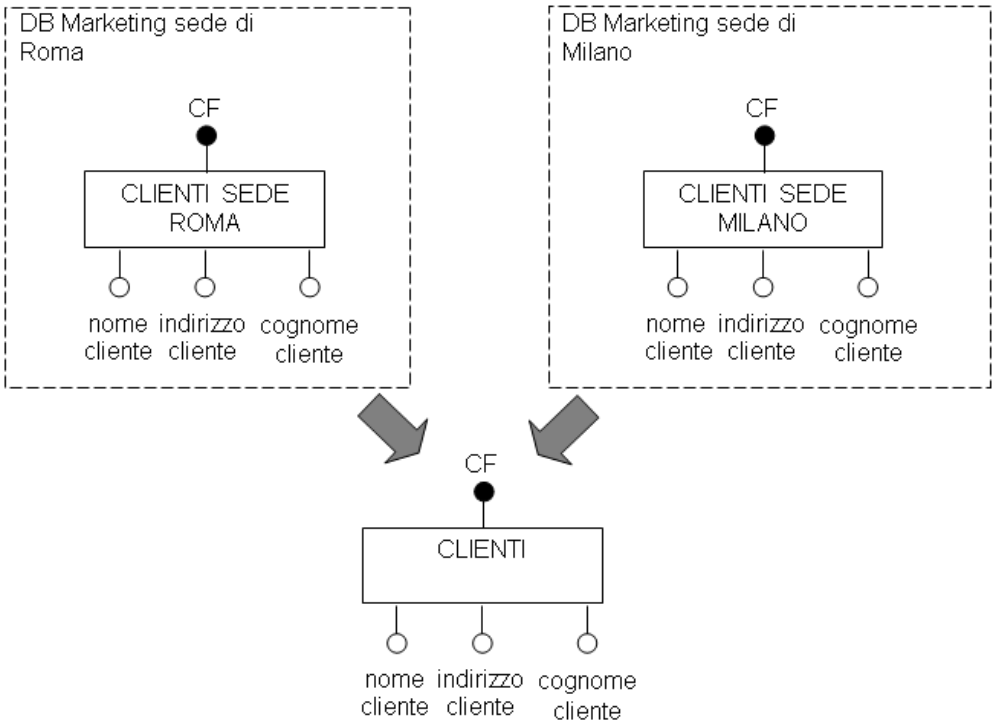
- Ogniqualvolta si debbano combinare in un'unica destinazione dati provenienti da sorgenti diverse senza una chiave comune, nasce la necessità di identificare i record corrispondenti.



- È stata scoperta un'associazione implicita tra l'entità DBMARKETING. CLIENTE , DBAMMINISTRAZIONE.ORDINE. Per esplicitare l'associazione è necessario individuare quale cliente ha inoltrato un particolare ordine. Se i due database identificano diversamente i clienti e quindi il join dovrà essere eseguito sulla base dei campi comuni (indirizzo cliente, cognome cliente) che non rappresentano un identificatore per il cliente. In questo caso si parla di ***join approssimato*** poiché non c'è certezza nella definizione dei record corrispondenti.

Tecniche basate su dizionari

- Quando due istanze diverse di uno stesso schema devono essere fuse assieme si parla di **purge/merge problem**. Le due istanze non sono disgiunte (un cliente può acquistare sia a Roma che a Milano), inoltre anche nei singoli database i clienti potrebbero essere inseriti più volte a causa di errori di battitura che non hanno permesso di verificare un precedente inserimento.



Similarità tra record

- **Tecniche basate su funzioni di similarità:** comparano le sottostringhe di $A_i \in A$ e $B_i \in B$, calcolando la similarità in base alla formula ricorsiva:

$$affinità(A,B) = \frac{1}{|A|}\sum_{i=1}^{|A|}\max_{j=1}^{|B|} affinità(A_i, B_j)$$

- dove la comparazione tra i sottocampi elementari è basata sul confronto tra stringhe e sulla verifica della presenza di abbreviazioni che vengono ricercate in base a pattern ben definiti:
  - Un'abbreviazione può essere un prefisso di una sottostringa, per esempio "Univ." è un prefisso per "Università".
  - Un'abbreviazione può combinare un prefisso e un suffisso di una sottostringa, un esempio è "Dott.ssa" che abbrevia "Dottoressa".
  - Un'abbreviazione può essere un acronimo di una stringa, per esempio DEIS è un acronimo per "Dipartimento di Elettronica Informatica e Sistemistica".
  - Un'abbreviazione può essere la concatenazione di più prefissi contenuti nella stringa, un esempio "UNIBO" che abbrevia "Università di Bologna".

Similarità tra record

- **Tecniche basate su gruppi di regole:** normalmente vengono valutate caratteristiche quali la edit-distance tra due stringhe o la differenza di valore tra due campi numerici.
- A titolo di esempio si propone, sotto forma di pseudocodice, alcune delle regole, proposte da Hernandez (1998), per la soluzione del purge/merge problem applicato a record contenenti informazioni relative a persone. L'idea di base è di calcolare la similarità tra coppie di record; tutti quelli la cui similarità supera una certa soglia vengono poi fusi in un unico record.

Similarità tra record

- **Tecniche basate su gruppi di regole:** normalmente vengono valutate caratteristiche quali la edit-distance tra due stringhe o la differenza di valore tra due campi numerici.
- A titolo di esempio si propone, sotto forma di pseudocodice, alcune delle regole, proposte da Hernandez (1998), per la soluzione del purge/merge problem applicato a record contenenti informazioni relative a persone. L'idea di base è di calcolare la similarità tra coppie di record; tutti quelli la cui similarità supera una certa soglia vengono poi fusi in un unico record.

Similarità tra record

- // Input: due record P1, P2 formati dai campi
  (ssns, nome, fname, addr, city, zip)
  // Output: VERO = P1 e P2 sono lo stesso record, FALSO = P1 e P2 non
  // sono lo stesso record,

  codiciSimili = comparaCodici(P1,P2);
  nomiSimili = comparaNomi(P1,P2);
  indirizziSimili = comparaIndirizzi(P1,P2);
  cittàSimili = comparaCittà(P1,P2);
  CAPSimili = comparaCAP(P1,P2);
  statiSimili = comparaStati(P1,P2);

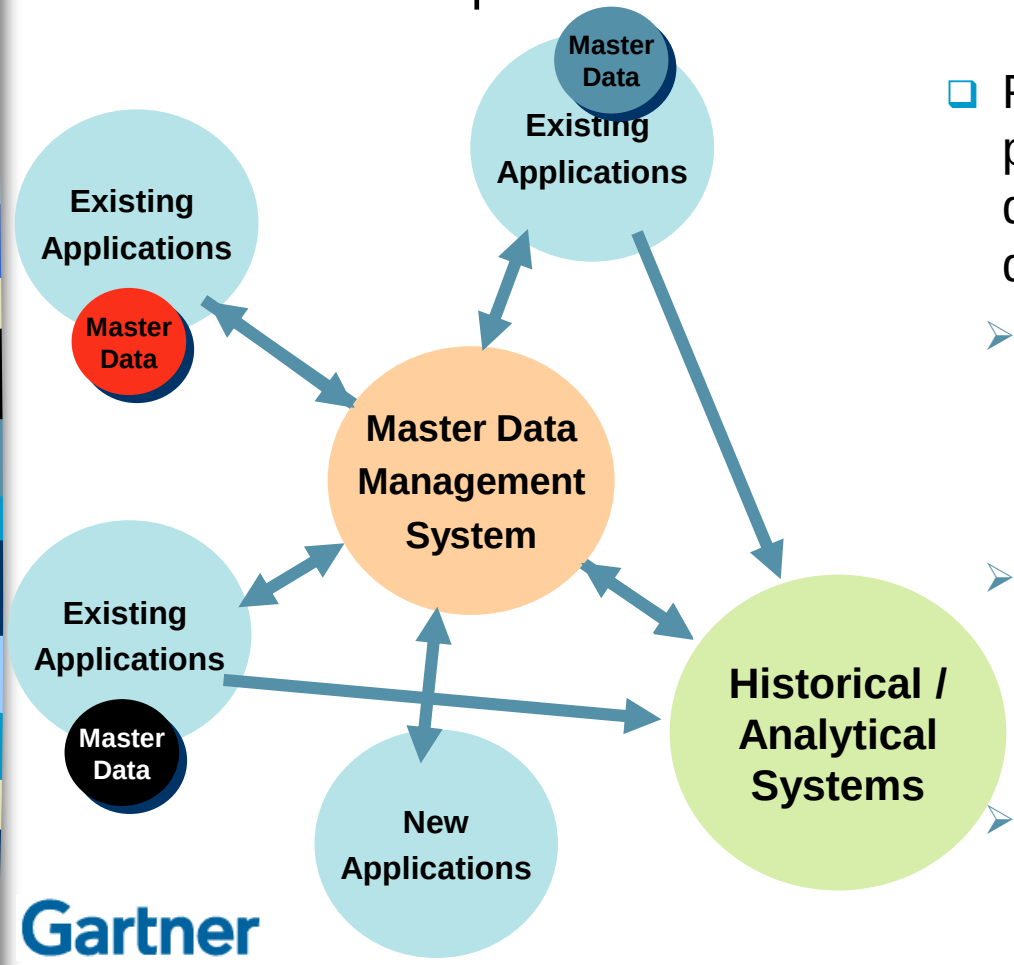
  se (codiciSimili **e** nomiSimili) **allora**
  restituisci VERO;
  indirizziMoltoSimili = indirizziSimili **e** cittàSimili **e** (CAPsimili **o** statiSimili);
  se ((codiciSimili **o** nomiSimili) **e** indirizziMoltoSimili) **allora**
  restituisci TRUE;
  ……

Il Master Data Management

- Con il termine “master data” si indica l’insieme dei dati che identificano e descrivono entità (prodotti, clienti, fornitori, locazioni…) fondamentali per il business di una azienda, che vengono gestiti attualmente da varie applicazioni, sia di tipo operazionale che analitico.
- Definiamo **Master Data Management (MDM)** l’insieme di discipline, tecnologie e soluzioni in grado di creare e mantenere consistenti, aggiornati, accurati e completi i dati di **importanza critica** e di fornirne una visione unica ad utenti, applicazioni e processi sia all’interno sia all’esterno dell’azienda.
- Nell’ambito dell’MDM si distinguono generalmente due aree più specifiche:
  - **PIM** (Product Information Management): incentrata sui Master Data di **Prodotto**
  - **CDI** (Customer Data Integration): incentrata sui Master Data relativi a **clienti/fornitori/utenti**

Il Master Data Management: idea di base

- La dispersione e la ridondanza delle informazioni di importanza cruciale su diverse applicazioni provoca problemi di consistenza e di inefficienza nei processi aziendali



- Per risolvere questi problemi si procede spostando i MD al di fuori delle singole applicazioni. Ciò ha diverse implicazioni
  - Viene creata una nuova base dati “master” sincronizzata con quelle esistenti mediante tecniche di data integration in tempo reale
  - Si modificano i processi di business che toccano i MD, per alimentare e sfruttare al meglio la nuova base dati ”master”
  - Va definita la “ownership” dei MD e dei processi di alimentazione e gestione che li riguardano.

Scegliere i master data

- Tipologie dei dati
  - **Non Strutturati:** E-mail. Documenti pdf, pagine web.
  - **Transazionali:** ordini, fatture, trouble ticket.
  - **Metadati**: descrizione di attributi, glossario aziendale
  - **Gerarchici:** descrivono le relazioni tra i concetti
  - **Master**: persone (clienti, impiegati, fornitori) cose (prodotti, materie prime,negozi, proprietà), luoghi e concetti.
- Caratteristiche dei MD
  - **Quantità:** la creazione di un'architettura per la gestione dei MD si giustifica solo per quantità elevate di informazioni
  - **Lifetime:** i MD tendono a essere meno volatili dei dati transazionali. In base a ciò aziende diverse possono fare scelte per la stessa informazione
    - I contratti possono essere considerati MD da un'azienda se la loro durata è pluriennale
  - **Valore:** maggiore è il valore di una informazione, maggiore la probabilità che sia considerato un MD
  - **Complessità:** per dati con limitate problematiche di gestione è poco utile allestire un meccanismo di MDM

Scegliere i master data

- Caratteristiche dei MD
  - **Riusabilità:** un concetto riutilizzato da più sistemi aziendali richiede con elevata probabilità una gestione basata su un sistema di MDM
  - **Centralità:** i MD sono centrali a più applicazioni e vengono da essi creati, modificati, letti e cancellati

	Customer	Product	Asset	Employee
Create	Customer visit, such as to Web site or facility; account created	Product purchased or manufactured; SCM involvement	Unit acquired by opening a PO; approval process necessary	HR hires, numerous forms, orientation, benefits selection, asset allocations, office assignments
Read	Contextualized views based on credentials of viewer	Periodic inventory catalogues	Periodic reporting purposes, figuring depreciation, verification	Office access, reviews, insurance-claims, immigration
Update	Address, discounts, phone number, preferences, credit accounts	Packaging changes, raw materials changes	Transfers, maintenance, accident reports	Immigration status, marriage status, level increase, raises, transfers
Destroy	Death, bankruptcy, liquidation, do-not-call.	Canceled, replaced, no longer available	Obsolete, sold, destroyed, stolen, scrapped	Termination, death
Search	CRM system, call-center system, contact-management system	ERP system, orders-processing system	General Ledger tracking, asset DB management	HR LOB system

La gestione dei Master Data

- Il MDM non è semplicemente un problema tecnologico, in molti casi impatta pesantemente i processi aziendali e determina problemi di natura politica e organizzativa
  - Chi è il proprietario dei dati?
  - Chi è responsabile della loro manutenzione e pulizia?
- Il MDM include sia la creazione, sia la manutenzione dei MD di conseguenza un progetto di MDM non può essere considerato un'attività una tantum poiché determina una porzione di SI che deve essere mantenuta nel tempo.
- Le principali fasi di un progetto di MDM sono:
  1. **Identificare le sorgenti dati.** Tipicamente il numero dei DB che memorizzano MD è superiore alle aspettative.
  2. **Specificare produttori e consumatori di MD.** Il tipo di operazioni e la numerosità degli attori che operano sui MD incidono sulla complessità del processo di gestione
  3. **Raccogliere meta dati sui MD.** E' fondamentale definire univocamente i MD sia a livello sintattico (formati, tipi di dati), sia a livello semantico.

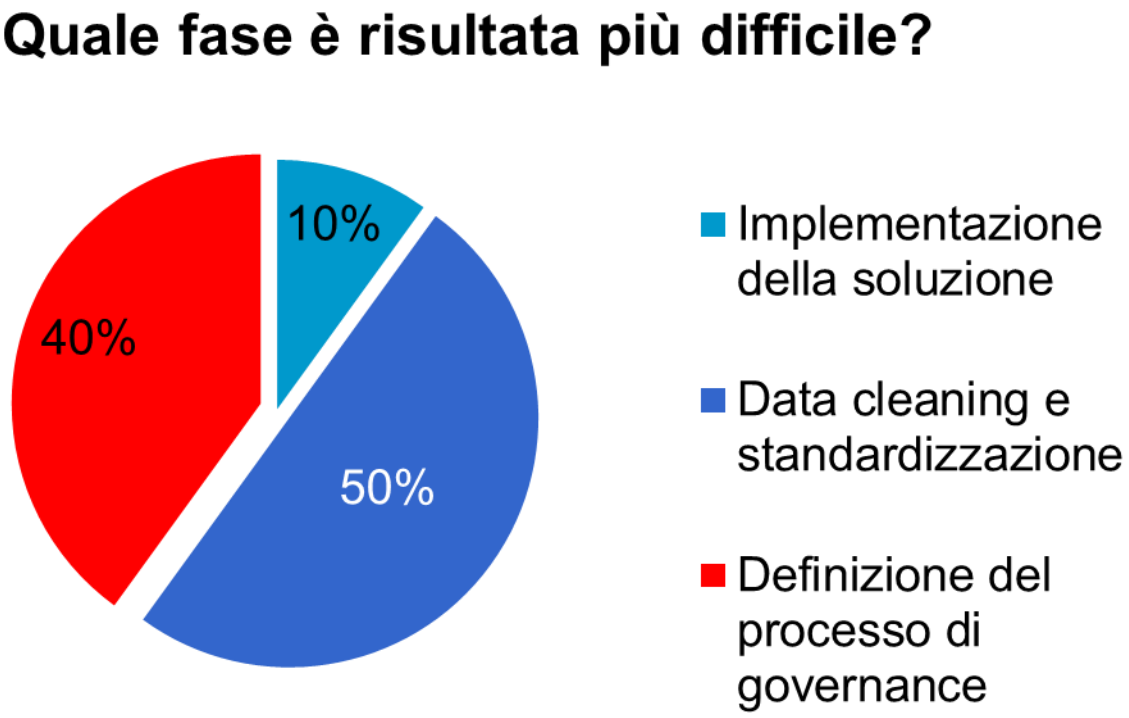
La gestione dei Master Data

- Le principali fasi di un progetto di MDM sono: (..continua)
  4. **Identificare gli esperti dei MD.** Per ogni sorgente si identificano persone in grado di descrivere come i MD debbano essere trasformati per renderli compatibili con la versione condivisa.
  5. **Definire il processo di data-governance e il relativo gruppo di lavoro.** Il gruppo di lavoro deve avere le competenze e l'autorità per prendere decisioni relatiamente a come gestire i MD.
  6. **Sviluppare un modello di gestione dei MD.** Dal punto di vista tecnico è la fase più importante del progetto poichè in essa si decide:
     - Quale architettura utilizare
     - Quale sarà il formato dei MD
     - Quali record saranno inclusi/esclusi
  7. **Scegliere uno strumento /tool.** Nel caso in cui il sistema di gestione dei MD non sia sviluppato internamente, il tool viene scelto in base al tipo di MD gestiti e alle scelte fatte nella fase precedente.

La gestione dei Master Data

- Le principali fasi di un progetto di MDM sono: (..continua)
  8. **Progettare e implementare l'infrastruttura di gestione.** L'infrastruttura si occupa di collezionare e rendere disponibili alle applicazioni esistenti i MD.
  9. **Generare e testare i master data.** Questo processo iterativo permette di fare evolvere la soluzione grezza al fine di considerare progressivamente tutte le specificità dei dati non catturate durante la prima fase di analisi.
  10. **Modificare le applicazioni produttrici e consumatrici di MD.** In base alle interfacce messe a disposizione e in base alle poliche di gestione prescelte le applicazioni che modificano o utilizzano I meta dati dovranno essere modificate.
  11. **Definire il processo di manutenzione.** Il processo di manutenzione deve mettere in grado gli esperti dei MD di verificarne la qualità e di correggere gli errori. Il processo deve quindi definire ruoli, modi e tempi per la manutenzione dei MD ma deve anche prevedere gli strumenti di attuazione. Un tool per il controllo dei MD deve per esempio:
      - Mostrare quali inconsistenze si sono verificate sui dati
      - Mostrare quale sorgente e quale utente è responsabile dell'incoerenza (data tracing e data auditing)
      - Suggerire correzzioni

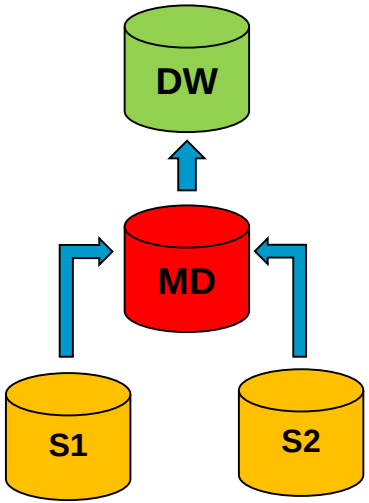
Complessità di un progetto MDM



Fonte: AMR Research - MDM Strategies for Enterprise Applications (April 2007)

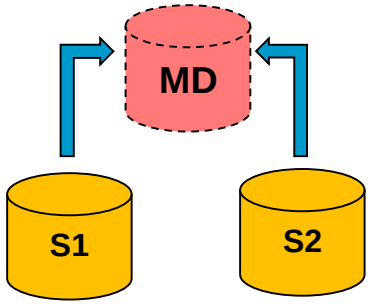
Architetture per MDM: consolidamento

- L'architettura viene fisicamente istanziata mediante un hub centrale che contiene i "golden record"
  - La proprietà dei dati rimane delle applicazioni sorgenti che li possono modificare in autonomia
  - L'aggiornamento dei MD non è sincrono con gli eventi che li creano/aggiornano conseguentemente l'architettura non garantisce che i MD siano aggiornati
  - I MD sono utilizzati principalmente per le attività reportistica direzionale
  - I MD ***possono*** essere scaricati periodicamente sulle applicazioni che li utilizzano come versione di riferimento (evolvendo in un'architettura di coesistenza)
    - Comporta una modifica alle sorgenti che devono pervedere come integrare il dato di ritorno
  - I MD sono utilizzati per armonizzare il comportamento tra più applicazioni ma non sono utilizzati durante le transazioni dei sistemi sorgente



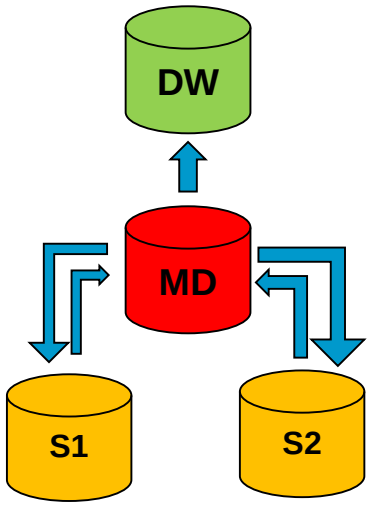
Architetture per MDM: a registro

- Viene costruito un registro centrale che collega le versioni locali dei dati di cui si è verificata la corrispondenza
  - Le sorgenti pubblicano i propri dati e l'hub contiene un riferimento ad essi
  - La proprietà dei dati rimane delle applicazioni sorgenti che li possono modificare in autonomia
  - L'hub esegue gli algoritmi di pulizia e di match tra i record, assegna un identificatore univoco a ogni record distinto e gli associa le versioni corrispondenti nelle diverse applicazioni
  - Nessun aggiornamento è inviato alle sorgenti
  - Il miglioramento di qualità raggiungibile è limitato
  - La logica per la ricostruzione del dato è complessa poiché deve agire su più applicazioni
  - L'impatto sulle applicazioni e sui processi è limitato
  - I MD sono utilizzati per armonizzare il comportamento tra più applicazioni ma non sono utilizzati durante le transazioni dei sistemi sorgente
    - Il cliente 123 del modulo Ordini corrisponde al cliente 257 dell'applicazione Fatturazione



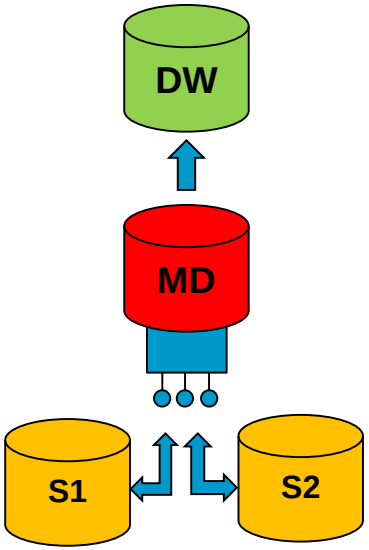
Architetture per MDM: coesistenza

- Viene costruito un hub centrale che mantiene una versione aggiornata dei dati che viene in seguito (con modalità non sincrona) riversata sulle sorgenti
  - La proprietà dei dati rimane delle applicazioni sorgenti che li possono modificare in autonomia
  - L'aggiornamento dei MD non è sincrono con gli eventi che li creano/aggiornano conseguentemente l'architettura non garantisce che i MD siano aggiornati
    - Normalmente le regole e le tempistiche di aggiornamento sono definite autonomamente dalle singole applicazioni
  - Gli MD sono utilizzati per armonizzare il comportamento tra più applicazioni e come punto di riferimento centralizzato



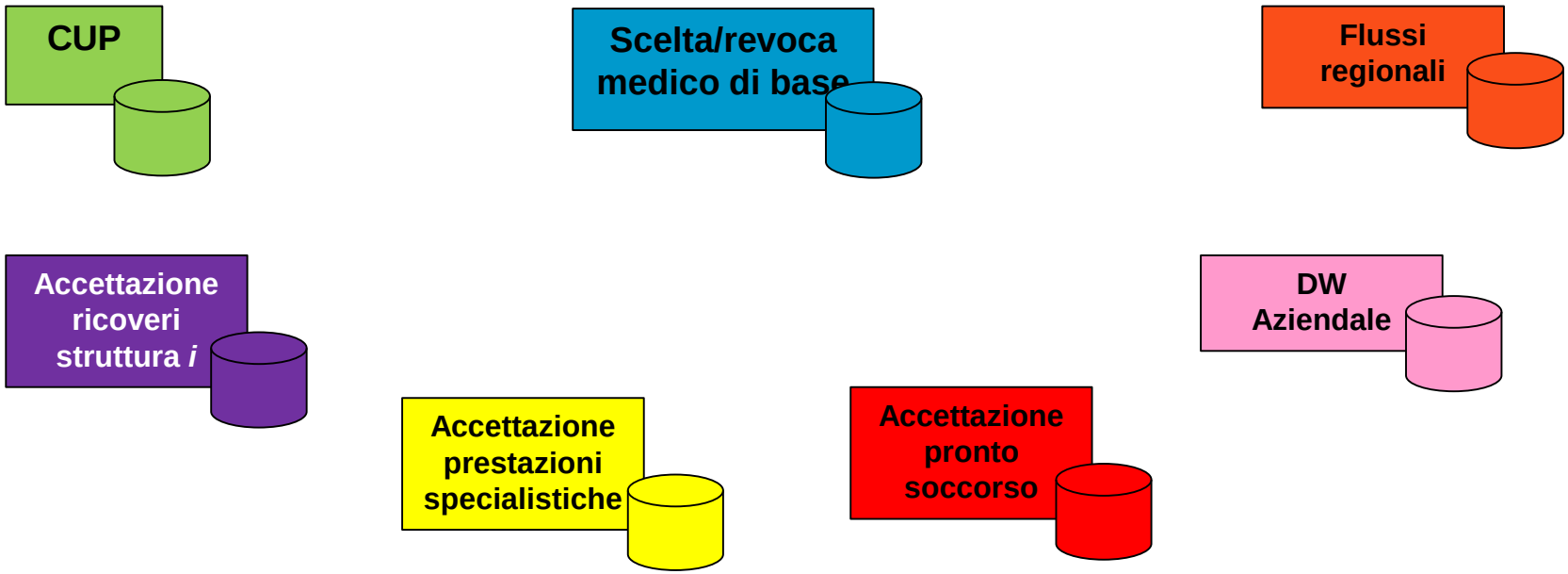
Architetture per MDM: Transazionale

- Viene costruito un hub centrale che mantiene una versione aggiornata utilizzata in modo sincrono da tutte le applicazioni
  - La proprietà dei MD è trasferita all'hub che è l'unico soggetto che può modificarli sulla base delle segnalazioni effettuate dalle sorgenti
  - Il dato aggiornato deve essere reso disponibile in modo sincrono alle sorgenti
    - Si sfrutta principalmente il meccanismo di publish-subscribe dei web service
    - Dove possibile è l'hub stesso che aggiorna i db degli applicativi sorgenti in modo da limitare le modifiche alle applicazioni
    - L'hub utilizza politiche transazionali per garantire la consistenza delle informazioni
  - Gli MD sono utilizzati sia nella normale operatività delle applicazioni operazionali sia per armonizzare il comportamento tra più applicazioni



Un esempio concreto: il caso dell'AUSL

- Un'AUSL vuole valutare possibili soluzioni all'annoso problema dell'anagrafica assistiti
  - I residenti nella provincia di riferimento dell'AUSL
  - I non residenti che per scelta o necessità usufruiscono di prestazioni (es. un ricovero, una visita specialistica) presso gli ospedali, cliniche dell'AUSL
- Al momento i dati anagrafici degli assistiti sono mantenuti da più applicazioni, tra loro parzialmente incoerenti e ridondanti



AUSL: descrizione della situazione attuale

- **Scelta/Revoca medico di base:** è l'ufficio che si occupa, di associare a tutti i residenti un medico di base/pediatra. La scelta/revoca deve essere fatta forzatamente presentandosi all'ufficio
- **CUP - Centro Unico Prenotazioni:** è l'ufficio che gestisce le prenotazioni per tutte le prestazioni specialistiche. La prenotazione può essere fatta allo sportello oppure telefonicamente utilizzando il codice della ricetta del medico di base
- Ogni struttura dispone di un ufficio accettazioni che regola ricoveri e dimissioni
- Il reparto di **pronto soccorso** ha un proprio sportello di accettazione vista l'urgenza dei ricoveri
  - I pazienti possono giungere al PS in stato confusionale o in stato di incoscenza e devono comunque essere registrati
- Ogni struttura dispone di più punti di accesso che regolano la fruizione delle prestazioni specialistiche (es. accettazione radiologia, accettazione prelievi del sangue)
- I flussi **regionali** permettono di comunicare con le altre AUSL le prestazioni svolte su assisiti non residenti
  - In uscita sono spedite tutte le prestazioni/ricoveri effettuate da NON residenti fatte presso l'AUSL
  - In entrata vengono ricevute tutte le prestazioni/ricoveri effettuate da residenti della provincia (e quindi associati all'AUSL) fatte in altre AUSL

AUSL: descrizione della situazione attuale

- **Come anagrafica principale è stata scelta quella dell'ufficio** scelta/revoca del medico di base poichè tale scelta rappresenta una operazione obbligatoria **per tutti e soli i residenti** e da fare di persona
  - Anche gli assistiti che non fanno prestazioni/ricoveri sono in anagrafica!
- L'anagrafica principale aggiornata è resa disponibile (in formato .csv) sull'intranet dell'AUSL agli altri uffici con frequenza mensile
  - Non esiste una logica unificata per la fusione degli archivi
  - Gli archivi locali sono spesso più aggiornati visto che è raro recarsi all'ufficio scelta/revoca
- Ogni applicativo ha una propria struttura relazionale
  - I campi si differenziano nelle diverse applicazioni in base alle funzioni svolte (es. la data dell'ultimo ricovero è presente solo nel db dell'ufficio accettazione ricoveri)
  - I formati non sono standardizzati
- Per gli assistiti extra-AUSL fanno fede la tessera sanitaria e i dati inseriti al primo contatto (per ogni applicativo)
  - Per gli stranieri senza CF viene inserita una codifica a parte
- I flussi regionali in entrata sono sempre ritardati di un mese

AUSL: diagnosi della situazione attuale

- Ogni applicazione può inserire nuovi assistiti. Di conseguenza un errore nell'inserimento del codice sanitario comporta la duplicazione dei dati.
  - Difficile tracciare tutte le prestazioni effettuate da un paziente nelle diverse strutture.

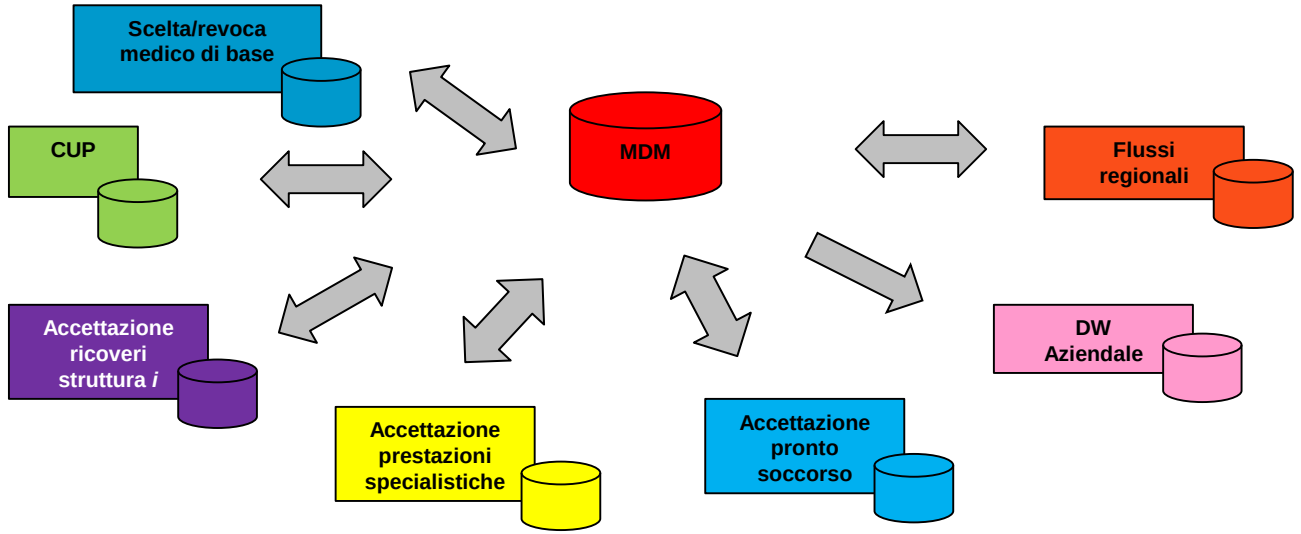
- L'incoerenza dei dati comporta problemi nelle attività operative
  - Un assistito fa una prenotazione al CUP. La prenotazione e il codice dell'assistito (ma non tutti i suoi dati) sono mandati all'ufficio di accettazione della struttura.
    - Se il codice sanitario è imputato male non risulterà alcuna prenotazione
    - Se l'indirizzo dell'assistito è cambiato i risultati non saranno ricevuti
    - Se il numero di telefeno è cambiato non sarà possibile comunicare eventuali variazioni nella prenotazione

AUSL: diagnosi della situazione attuale

- Solo gli aggiornamenti comunicati direttamente all'ufficio scelta/revoca vengono diffusi a tutte le applicazioni
- Ogni applicazione deve prevedere un propria procedura di fusione dei dati
- Il livello di affidabilità delle applicazioni varia fortemente
  - Ufficio scelta/revoca contiene spesso dati obsoleti e parziali (mancanza degli assistiti extra-AUSL)
  - Il Pronto Soccorso è il reparto con dati meno attendibili
  - Il CUP è l'ufficio con i dati più aggiornati perchè è l'ufficio in cui transitano la maggior parte delle prestazioni. I dati sono però meno attendibili se le prenotazioni sono fatte telefonicamente.
- La frequenza degli aggiornamenti è troppo bassa

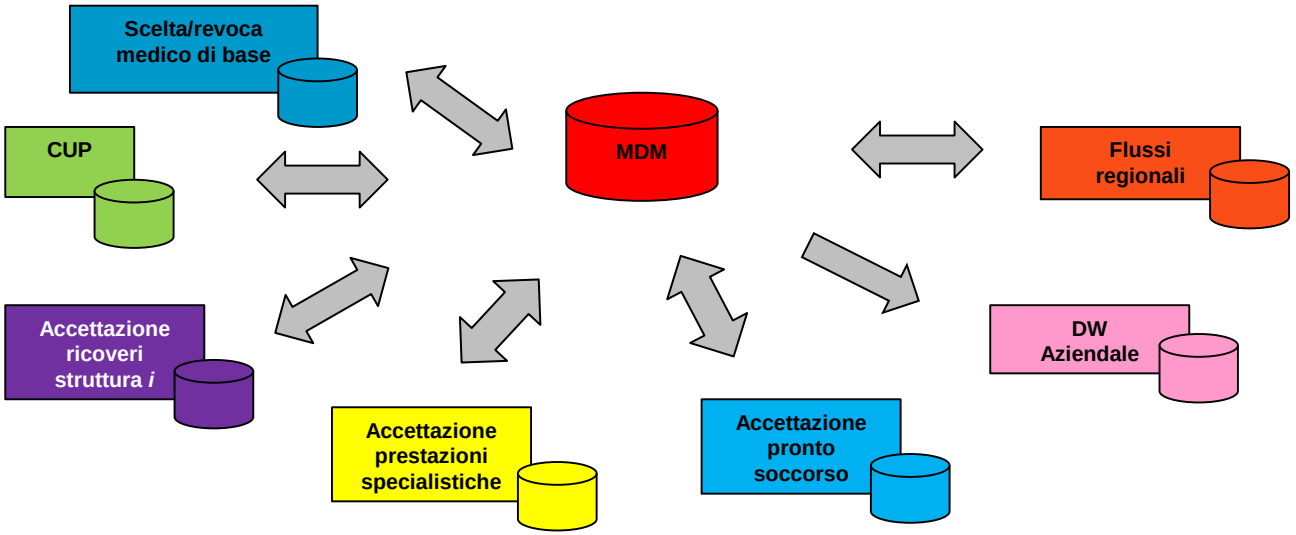
AUSL soluzione coesistenza

- Si costruisce un nuovo DB per i master data e si definiscono i flussi di alimentazione da e verso le applicazioni
  - A tutte le applicazioni deve essere aggiunta una procedura per l'integrazione dei dati in ingresso
  - I processi operazionali non sono modificati ma operano semplicemente su dati di qualità migliore (più aggiornati, più completi, più corretti)



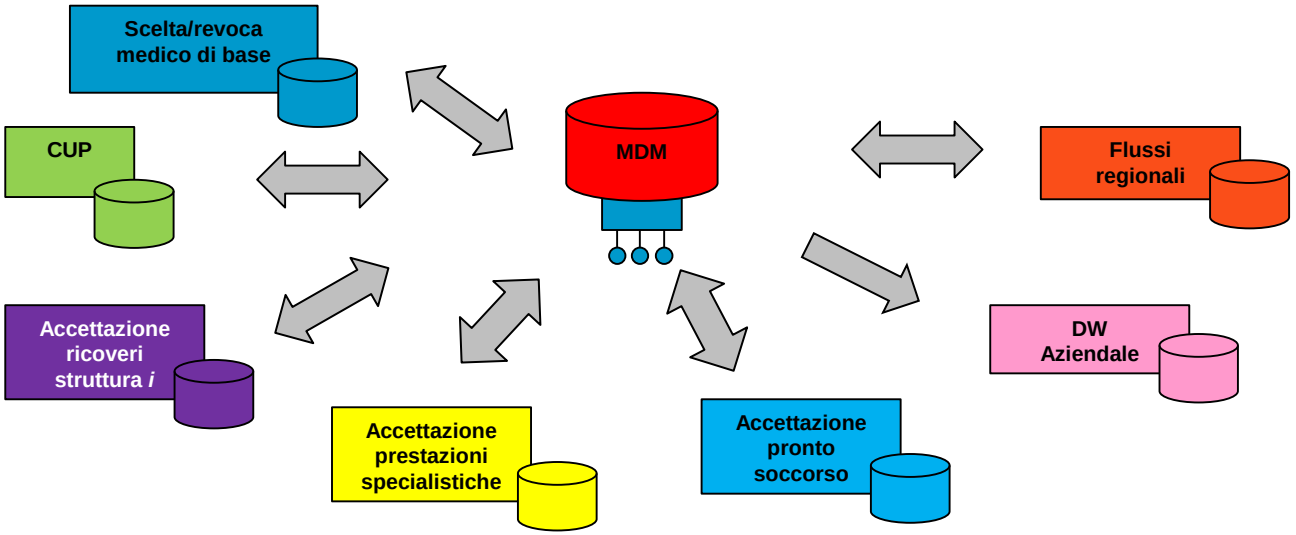
- Va definito l'insieme dei dati master lasciando sulle applicazioni locali quelli utili solo alla specifica applicazione

AUSL soluzione coesistenza



- Va definita la struttura standard dei dati e le regole di standardizzazione
  - Il formato deve soddisfare tutte le esigenze informative delle sorgenti ossia deve permettere di alimentarle
- Vanno definite le politiche di pulizia dei dati
  - Come comportarsi se un indirizzo non esiste?
- Va definita una politica per il merge dei dati presso l'hub
  - Se la sorgente mantiene la data dell'ultima modifica va considerato il dato più aggiornato
  - Informazioni discordanti vanno valutate in base al livello di affidabilità delle sorgenti
- Per aumentare la qualità dei dati sarebbe utile concordare la modalità e la tempistica di aggiornamento dei dati presso le sorgenti

AUSL soluzione transazionale



- Si costruisce un nuovo DB per i master data e si modificano tutte le applicazioni per permettere la gestione transazionale dei dati
  1. L'utente del CUP invoca il web service del sistema di MDM richiedendo i dati anagrafici della tessera sanitaria 1344
  2. Il web service restituisce l'ultima versione committed dei dati
  3. L'utente del cup richiede di modificare l'indirizzo di residenza
  4. Se nessuna applicazione sta utilizzando il record e le politiche di qualità sono soddisfatte il record è locked e l'utente cup può apportare le modifiche
     - Solo alcuni uffici possono effettuare le modifiche
  5. Il sistema verifica la qualità del dato e accetta/rifiuta l'aggiornamento
  6. Il lock viene è rilasciato e il dato diventa visisibile a tutti

AUSL transazionale vs coesistenza

- Contro transazionale
  - Maggiore impatto sulle applicazioni
  - Possibile solo solo se non esistono applicazioni legacy
    - Se ciò non è verosimile è necessario studiare una strategia mista transazionale-coesistenza
  - Assume che tutti i sistemi siano sempre in rete
    - Se temporaneamente un'applicazione non in rete può sfruttare una copia locale dei dati ma non deve permettere l'aggiornamento
  - Richiede di definire chi è l'owner del processo di gestione dei MD
    - Non tutti i problemi sono risolvibili da un processo automatico che comunque deve essere supervisionato, fatto evolvere, ecc.

- Pros transazionale
  - Garantisce sempre l'allineamento dei dati tra le applicazioni interne (circolarità dell'informazione)
  - Centralizza la gestione dei dati e le politiche di aggiornamento

What's Next: Integration, integration integration

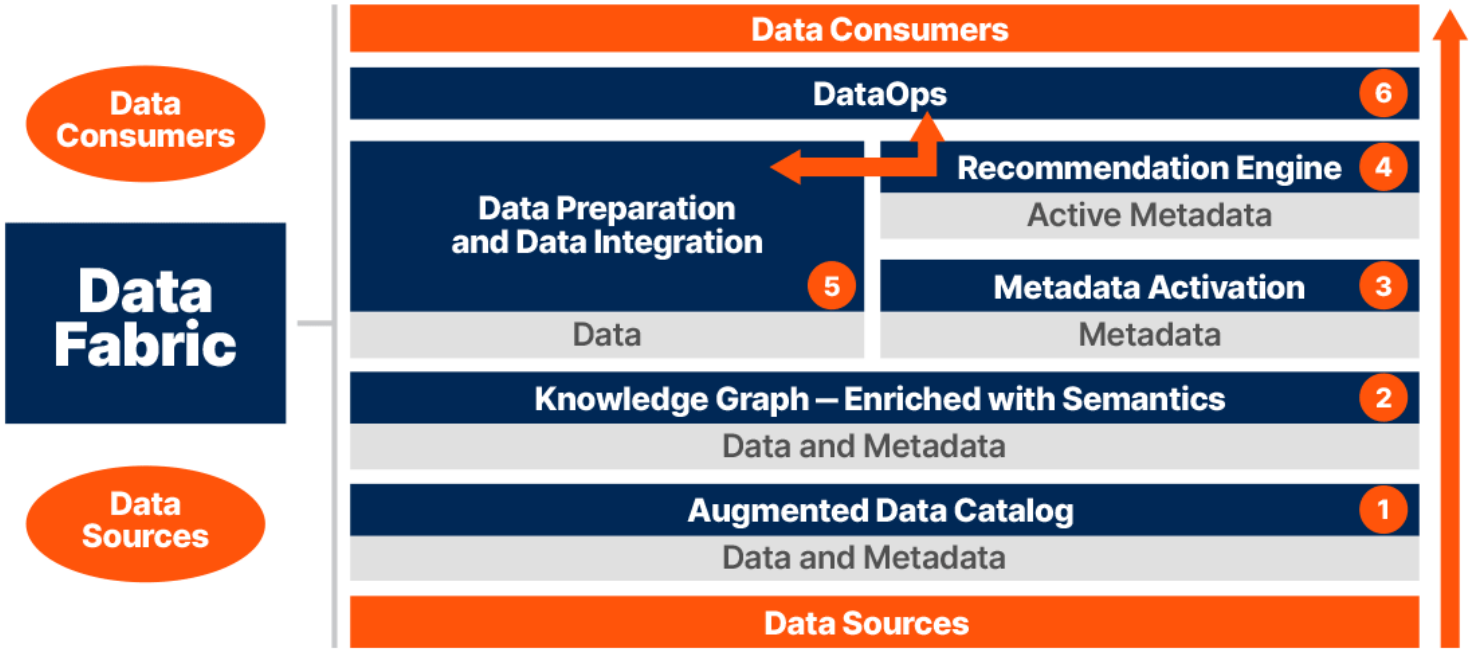
- L'integrazione dei dati è un obiettivo a tendere che i sistemi informativi inseguono dagli anni '70 perché con essa determina:
  - Qualità del dato
  - Ottimizzazione di processo
  - Capacità di analisi
- Con il procedere della digitalizzazione aumenta il numero di sorgenti dati e diventa sempre più complesso mantenerle integrate e sincronizzate

What's Next: Data Fabric

- Il **Data fabric** permette un accesso e una condivisione in un ambiente distribuito anche multi-cloud
  - Consente un quadro di gestione dei dati unico e coerente, che permette l'accesso ai dati e l'elaborazione senza soluzione di continuità a sorgenti che altrimenti sarebbero a silo
  - Sfrutta sia le capacità umane che quelle delle macchine per accedere ai dati sul posto o per supportarne il consolidamento laddove necessario
  - Identifica e collega continuamente i dati provenienti da applicazioni diverse scoprendo relazioni rilevanti per il business tra le sorgenti dati disponibili
- È un'architettura unificata con un insieme integrato di tecnologie e servizi
  - Progettata per fornire **dati integrati e arricchiti** - al momento giusto, nel metodo giusto e al giusto consumatore di dati - a supporto dei carichi di lavoro sia operativi che analitici
  - Combina le tecnologie chiave di gestione dei dati, come il catalogo dei dati, la governance dei dati, l'integrazione dei dati, il pipeline dei dati e l'orchestrazione dei dati

Gartner, 2019 https://www.gartner.com/en/newsroom/press-releases/2019-02-18-gartner-identifies-top-10-data-and-analytics-technolo
Gartner, 2021 https://www.gartner.com/smarterwithgartner/data-fabric-architecture-is-key-to-modernizing-data-management-and-integration
K2View Whitepaper: What is a Data Fabric? The Complete Guide, 2021

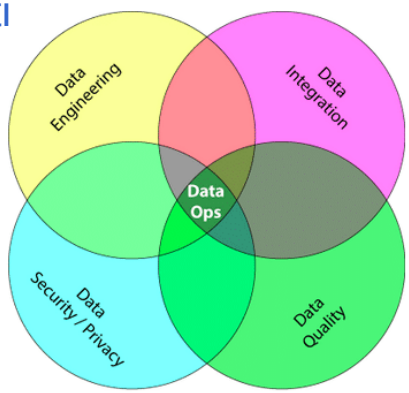
What's Next: Data Fabric



https://www.irion-edm.com/data-management-insights/gartner-data-summit-irion-representative-vendor-for-data-fabric-technology/

What's Next: Data Ops

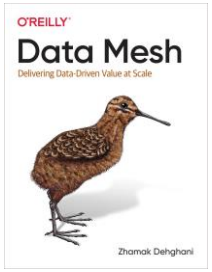
- DevOps deriva dalla contrazione inglese di *development*, "sviluppo", e *operations*, inteso come "messa in produzione" o "*deployment*"
- Dal DevOps al DataOps
  - *"A collaborative data management practice focused on improving the communication, integration and automation of data flows between data managers and data consumers across an organization"*
- Alcune regole di base
  - Stabilire misure di progresso e performance in ogni fase
  - Automatizzare il maggior numero possibile di fasi del flusso di dati
  - Stabilire una disciplina di governance (governance-as-code)
  - Progettare il processo per la crescita e l'estensibilità



Gartner, 2020 https://www.gartner.com/smarterwithgartner/how-dataops-amplifies-data-and-analytics-business-value
Andy Palmer, 2015 https://www.tamr.com/blog/from-devops-to-dataops-by-andy-palmer/
William Vorhies, 2017 https://www.datasciencecentral.com/profiles/blogs/dataops-it-s-a-secret

What's Next: Data Mesh

- *Data mesh is a decentralized sociotechnical approach in managing and accessing analytical data at a scale*



Data Mesh: Delivering Data-Driven Value at Scale

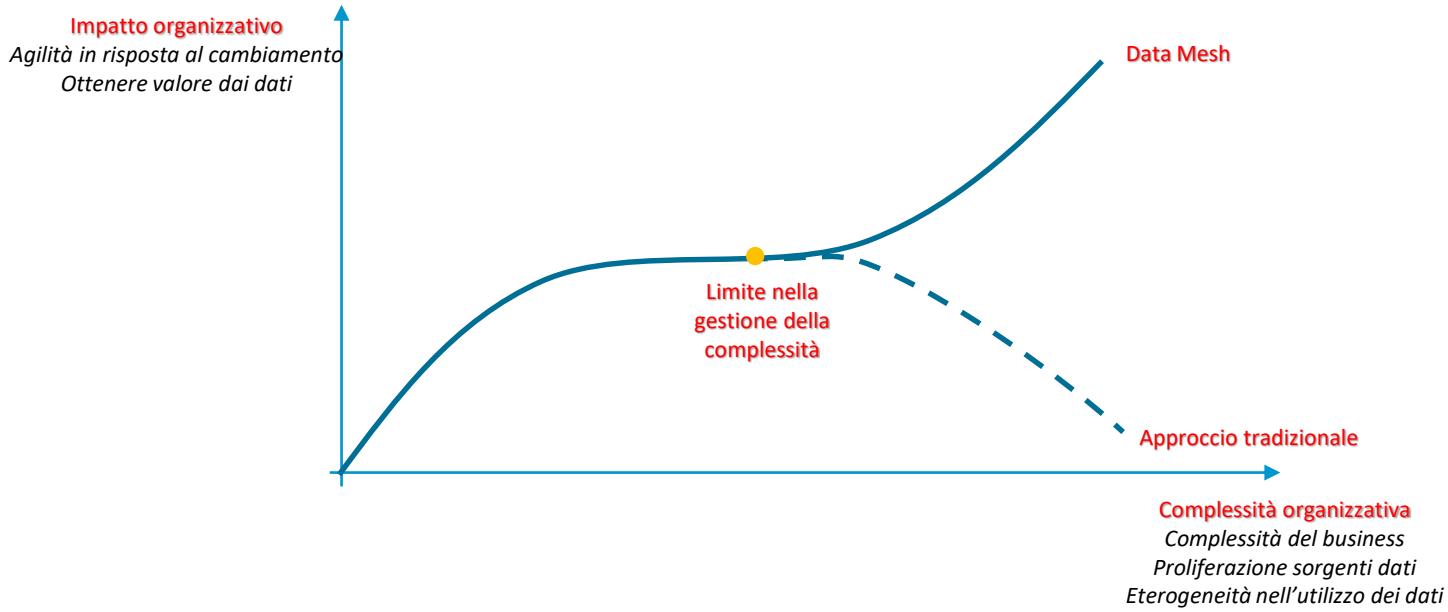
A book by Zhamak Dehghani

/thoughtworks

- Obiettivo: abilitare le organizzazioni a diventare maggiormente Data Driven cambiando il modo in cui queste organizzano i propri team e le proprie architetture dati

- L'approccio data mesh richiede competenze diffuse di Data Architect e Data Science in azienda!

What's Next: Data Mesh



Limiti delle Data Platform

- Col proliferale delle diverse sorgenti informative diventa sempre più complesso **armonizzare i dati all'interno di un modello centralizzato e mantenerlo aggiornato**.

- Le diverse modalità di fruizione del dato e la necessità di eseguire sperimentazioni veloci sul dato, portano all'esigenza di creare **molteplici pipeline di elaborazione difficili da gestire da un team centralizzato**.

- I continui cambiamenti nelle esigenze di business portano ad elevata complessità nella gestione della **sincronizzazione** e **prioritizzazione** delle pipeline di elaborazione.

- I team di lavoro sono organizzati per **competenza tecnologica** (non per competenza di business): rischio di difficoltà nel comprendere a pieno le esigenze del business e il business ha scarsa fiducia nel dato.

- Difficoltà nell'individuazione di chi ha **l'ownership sul dato**.

I principi del data mesh

- Domain ownership
- Data as a product
- Self-serve data platform
- Federated computational governance

Domain ownership: domain-oriented decentralization

- Nel paradigma Data Mesh l'infrastruttura dati è responsabile della fornitura delle tecnologie utili alla gestione e al processamento dei dati, ma sono i **domini** ad essere responsabili delle pipiline di ingestion, pulizia e aggregazione dei dati al fine di generare assets (Data Products) utilizzabili e consumabili da applicazioni e/o altri domini.

- **Ogni dominio è responsabile** quindi di gestire e manutenere le proprie pipeline (ETL) e un set di funzionalità applicate ad ogni dominio si occupano di salvare, catalogare e gestire gli accessi ai dati di terzi.

- Una volta che i dati sono stati forniti e in seguito trasformati dal rispettivo dominio, gli owners di dominio possono quindi sfruttare i dati per **esigenze analitiche** o **operazionali**.

- La decentralizzazione domain-oriented richiede che i team sui domini abbiano competenze pratiche nella gestione e analisi del dato. I **data scientist sono decentralizzati**.

Data as a product

- Le caratteristiche più importanti che un Data Product deve mappare sono:
  - Discoverable e Addressable: il Data Product deve fornire intrinsecamente e intenzionalmente tutte le informazioni utili a trovarlo facilmente oltre ad un indirizzo unico di accesso al dato (per fini operazionali e analitici), ad esempio tramite un Data Marketplace.
  - Understandable, Trustworthy e Accessible: i Data Products devono esporre informazioni semantiche che permettano la facile comprensione del dato, logiche di quality check, integrity test, lineage e tecniche di accesso native che ne estendano l'usabilità.
  - Interoperable e Secure: perché un data product sia interoperabile occorre utilizzare formati aperti e strutture dati stabili che rispettino le policy definite a livello globale, che prevedono anche aspetti di security e privacy.

- Il principio del «Data as a product» è pensato per indirizzare problematiche di data quality e dei vecchi data silo, o ancora meglio quello che Gartner chiama dark data.

Data as a product

- Le caratteristiche più importanti che un Data Product deve mappare sono:
  - Discoverable e A… e e intenzionalment… n indirizzo unico di accesso… mite un Data Marketplac…
  - Understandable… rre informazioni ser… , logiche di quality check, in… estendano l'usabilità.
  - Interoperable e Secure: perché un data product si… e occorre utilizzare formati aperti e strutture dati stabili che rispettino le po… a livello globale, che prevedono anche aspetti di security e privacy.

- Il principio del «Data as a product» è pensato per indirizzare problematiche di data quality e dei vecchi data silo, o ancora meglio quello che Gartner chiama dark data.

Gartner definisce i **dark data** come gli asset informativi che le organizzazioni raccolgono, elaborano e archiviano durante le normali attività aziendali, ma che generalmente non riescono a utilizzare per altri scopi (per esempio, analisi, relazioni commerciali e monetizzazione diretta). Simili alla materia oscura in fisica, i dati oscuri spesso comprendono l'universo delle risorse informative della maggior parte delle organizzazioni. Quindi, le organizzazioni spesso conservano i dati oscuri solo per scopi di conformità. Conservare e mettere in sicurezza i dati comporta tipicamente più spese (e a volte un rischio maggiore) che valore.

Federated Computational Governance

- Il Data Mesh segue i principi dei sistemi distribuiti, attraverso una collezione di Data Product indipendenti, con una propria gestione del lifecycle messa direttamente in mano ai team di dominio. Al fine di rendere questi Data Product interoperabili tra loro (generando quindi ulteriore valore all'interno dell'organizzazione), un'implementazione di Data Mesh richiede un **modello di governance** che abbracci allo stesso tempo:
  - La decentralizzazione in domini e l'ampia autonomia dei team nella definizione di Data Product.
  - La definizione di un set di **regole globali** da applicare a tutti i Data Product e alle loro interfacce di comunicazione.
- Un fattore critico di successo per il Data Mesh è quindi riuscire a trovare e mantenere **un giusto equilibrio tra centralizzazione e decentralizzazione**, definendo **quali decisioni debbano essere prese localmente e quali invece vadano definite globalmente**, quali Data Product decentralizzare maggiormente e quali mantenere sotto un'ownership più centrale.

Self-Service Platform

- Per sviluppare, distribuire, eseguire, monitorare e, infine, accedere ai nostri Data Product occorre un'infrastruttura (Data Platform) adeguata. Le competenze necessarie per definire e sviluppare un ecosistema infrastrutturale di questa complessità sono altamente specializzate ed è necessario un coordinamento e una governance architetturale centralizzata.

- La Data Platform deve fornire i servizi utili ai team di dominio al fine di poter gestire l'intera filiera dei propri Data Product.

- È necessario quindi predisporre una self-serve data infrastructure: una Data Platform che espone servizi e funzionalità che abilitano i team di dominio a gestire il ciclo di vita dei Data Product in autonomia.

Il Data Mesh non fa per voi se....

- Pensate al Data Mesh come a una tecnologia e non come a un modello operativo
- Pensate al Data Mesh come a una soluzione a scaffale
- Non è stata definita una Data strategy aziendale
- Non emergono business-case (data driven) che portano valore alla business unit (dominio)
- Manca il mindset culturale rispetto a processi di decision-making in modalità bottom-up
- C'è scarsità di risorse *data-addicted* all'interno dell'organizzazione e dei team (data analyst, data scientist, data engineer)

Data Mesh vs Data Fabric

- Data fabric e data mesh definiscono architetture per accedere ai dati attraverso più tecnologie e piattaforme
  - Data fabric
    - Tenta di centralizzare e coordinare la gestione dei dati
    - Affronta la complessità basandosi pesantemente su metadati
  - Data mesh
    - Enfasi sulla decentralizzazione e sull'autonomia del dominio dei dati
    - Si concentra sul cambiamento organizzativo, sulle persone e sui processi
- Defiscono approcci risolutivi, non soluzioni pratiche
  - Non si escludono a vicenda
  - Sono framework architetturali, non architetture
    - I framework devono essere adattati e personalizzati alle vostre esigenze, ai vostri dati, ai vostri processi e alla vostra terminologia
    - Gartner stima che il 25% dei fornitori di gestione dei dati fornirà una soluzione completa di data fabric entro il 2024, rispetto all'attuale 5%.

Alex Woodie, 2021 https://www.datanami.com/2021/10/25/data-mesh-vs-data-fabric-understanding-the-differences/
Dave Wells, 2021 https://www.eckerson.com/articles/data-architecture-complex-vs-complicated