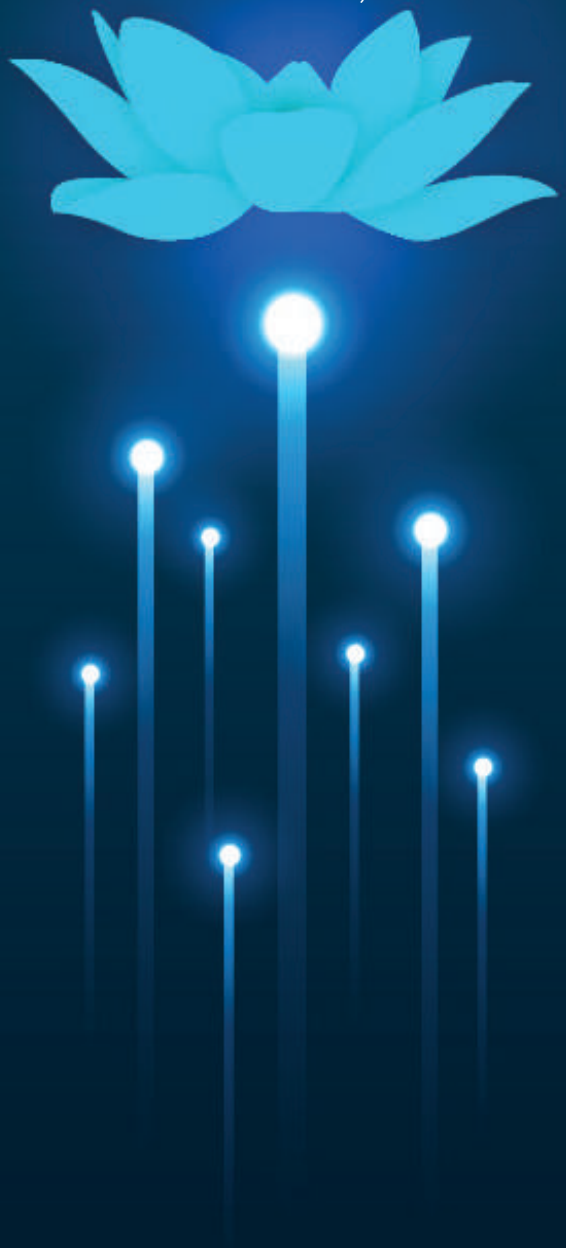
Tecnologie open e modelli dati comuni nei sistemi informativi manifatturieri del futuro

Standard tecnologici e architetturali

Autori:
Giovanni Miragliotta (Politecnico di Milano)
Marco Tessarin (CEO, SMC Treviso)
Mauro Mariuzzo (Senior Software Architect, SMC Treviso)

Data: Marzo 2021



DIGITAL Lake

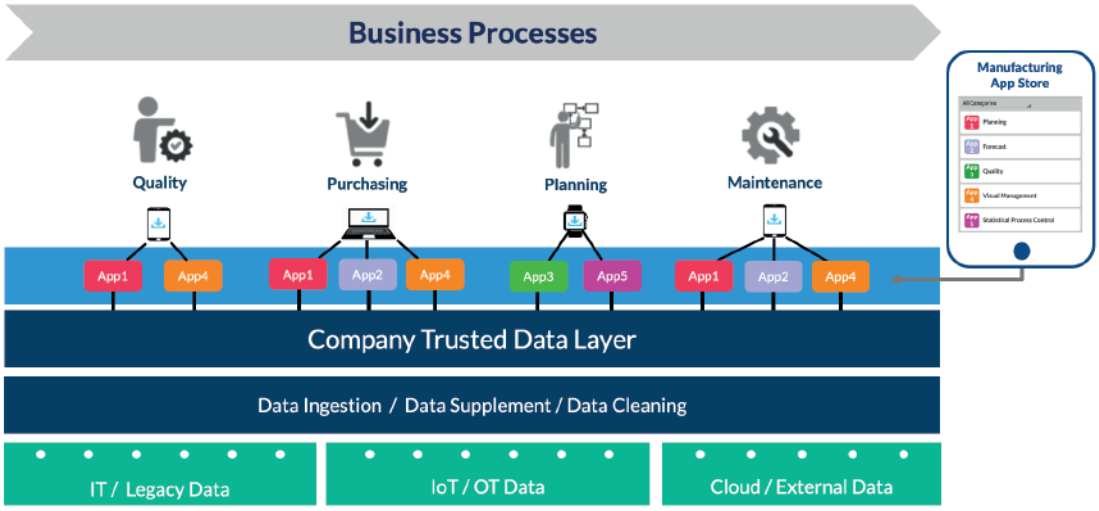
Indice

Prosegue il percorso di approfondimento sull'evoluzione dei sistemi informativi aziendali e sul loro ruolo a supporto dei processi di business. Con questa analisi di **Giovanni Miragliotta**, **Marco Tessarin**, **Mauro Mariuzzo** si focalizza l'attenzione sul ruolo degli standard tecnologici e architetturali.

Per tutte le imprese del mondo manifatturiero la progettazione della traiettoria di evoluzione del proprio sistema informativo è un tema che incide direttamente sulla competitività delle imprese. Nell'analisi del **sistema informativo e del perché è importante ridisegnarlo e renderlo a prova di futuro**, si è esaminata l'evoluzione dei sistemi informativi aziendali e il loro ruolo a supporto dei processi di business. Con questo nuovo contributo si affrontano i temi legati alle tecnologie open e ai modelli dati comuni nei sistemi informativi manifatturieri del presente e del futuro.

Premessa

Nella precedente memoria abbiamo descritto, e giustificato, come sarà costituita in futuro l'architettura di un sistema informativo, pensando ad una impresa generica ma in particolare avendo in mente un'impresa industriale manifatturiera.
Nella visione che avevamo presentato (cfr. Figura 1), grazie alla crescente maturità di diverse tecnologie, l'azienda potrà strutturare ed utilizzare appieno il proprio patrimonio informativo, sia quello residente nei propri sistemi transazionali classici, sia quello che genera attraverso i propri sistemi / prodotti connessi al mondo reale, e arricchirlo grazie alle sorgenti dati esterne (web / cloud). Su queste sorgenti dati l'impresa appoggerà un insieme di applicazioni a supporto del business (un vero e proprio catalogo di Business App) capaci di rispondere ai bisogni dei propri manager, di fronte alle complesse decisioni che li attendono.



Ma come si realizza, tecnicamente, questa architettura? Questa seconda memoria sarà dedicata a rispondere a questa domanda. Saranno due gli elementi approfonditi: le **Tecnologie Open** che sostengono lo stack architetturale del **Trusted Data Layer**, in particolare quelle dedicate al collegamento tra mondo fisico e mondo digitale, ed i **Modelli di Rappresentazione dei Dati**, in particolare quelli basati su standard e riferimenti comuni.

Prima di entrare nel merito di queste tematiche, che con la loro tecnicalità possono rappresentare un ostacolo per taluni, è bene richiamare l'impatto che hanno sul business: in un recente evento pubblico, il General Manager di una grande assicurazione italiana sosteneva che oggi non si può aspirare ad un ruolo manageriale di rilievo, in nessuna compagnia, senza delle solide basi di Information Technology, e citava ad esempio la conoscenza del ruolo e della natura della API come di una tecnologia che ha riscritto il modo in cui si possono realizzare, e dunque immaginare, nuove applicazioni per il personale ed i clienti. Facciamo dunque insieme questo sforzo, perché è davvero importante che alcuni concetti, pur con le loro tecnicalità, entrino a far parte del bagaglio di conoscenza di manager e dipendenti la cui vocazione sia più vicina al business.

Approccio aperto

Nella costruzione di un Trusted Data Layer dobbiamo pensare alla necessità di avere un sistema aperto. Usiamo questo temine nell'accezione più ampia. Un patrimonio informativo che sia pensato e costruito con logiche di medio-lungo respiro deve essere pensato in modo indipendente da applicativi (ERP, CRM, etc.) e macchinari (automazione classica, IoT, etc.) o loro evoluzioni e release, seguendo una logica di "No Vendor Lock-in" che deve permeare tutte le scelte di progettazione e di investimento. L'attenzione verso piattaforme e soluzioni Open Source ne è una naturale conseguenza perché esse forniscono maggiori garanzie proprio per la loro natura.

Tecnologie Open Source

Il termine "Open Source" è utilizzato, in modo generico, per definire una filosofia e un sistema di valori che celebrano lo scambio aperto, la partecipazione collettiva, la trasparenza, la meritocrazia e lo sviluppo della comunità. Tale filosofia applicata nell'ambito del software consente agli utenti il libero accesso e l'utilizzo del codice sorgente, che può essere adattato, modificato e ridistribuito. Tale metodologia è considerata rivoluzionaria rispetto a quella tradizionale del software commerciale coperto da copyright, in quanto coinvolge sviluppatori distribuiti in tutto il mondo che evolvono il codice correggendo bug ed ottimizzandolo nel tempo.

Le tecnologie Open Source, soprattutto nell'opzione Enterprise (supportate e mantenute con livelli di servizio definiti), si sono completamente liberate dall'idea dell'approccio "geek": le piattaforme Open Source sono oggi del tutto paragonate per qualità, solidità, robustezza e visione a quelle commerciali, e come conseguenza infrastrutture critiche altamente rilevanti (come la gran parte dei server internet) oggi girano su software Open Source[1]. Non è, però, solo una questione di apertura del codice sorgente, bensì è una filosofia di apertura e fruizione del dato che, attraverso la presenza di API ben documentate, apre alla possibilità di accedere alle informazioni in modalità aperta e senza vincoli. In questo paragrafo ci limitiamo ad indicare solo alcune (tra le tante) caratteristiche che spingono alla creazione di Trusted Data Layer basati su piattaforme Open Source:

- L'Open Source è più rispettoso degli standard ufficiali e de-facto, perché indipendente e quindi in grado di garantire il massimo livello di intercambiabilità. Certamente vi potranno essere dei vincoli relativi a singoli prodotti specifici ma, con un costo abbastanza basso, si potrà sostituire un elemento dell'architettura con uno diverso e compatibile.
- L'Open Source è, salvo rari casi, più ricettivo a suggerimenti nell'evoluzione del prodotto, anche a costo di divergenze tra la comunità che potrebbe produrre delle biforcazioni nel codice (fork). Anche questo aspetto ne diventa una valida azione di cambiamento libero ed evolutivo verso nuove funzionalità e possibilità delle piattaforme.
- Se il Trusted Data Layer rappresenta il patrimonio informativo aziendale è necessario poterlo gestire in totale autonomia e libertà. L'Open Source permette questo anche a figure professionali interne in grado di acquisire e padroneggiare la soluzione. Va inoltre ricordato che con le piattaforme commerciali l'approntamento di workaround tattici è difficile, se non impossibile, e c'è sempre il rischio di violazione del copyright.

1 E. Brynjolfsson, A. McAfee, 2017, Machine, Platform, Crowd, W. W. Norton & Co.

- Le soluzioni Open Source, almeno quelle community, sono spesso esenti da vincoli di scalabilità, aspetto da tenere fortemente in considerazione laddove il Trusted Data Layer abbia la necessità di crescere negli anni al crescere dell'azienda.

API

Se conserviamo la visione inspiratrice dell'ecosistema smartphone ed in particolare la centralità dello strato di applicazioni (e delle comunità di sviluppatori), appare chiara l'importanza di dare accesso ai dati e a servizi offerti dagli strati inferiori, a favore delle funzionalità che dovranno essere realizzate dagli strati applicativi superiori.
Riprendendo l'esempio della stima del rischio di fornitura, l'applicazione dovrà poter accedere ai dati di difettosità generati in tempo reale dai macchinari, per individuare una situazione di potenziale perdita di controllo del fornitore sui materiali forniti, e quindi ai sistemi IT gestionali per acquisire dati in merito ai prodotti finiti che ne saranno impattati, e con quali ricadute sui margini attesi a budget.
Questo non è un problema nuovo: nell'approccio best of breed si sono sempre dovute realizzare interfacce applicative per permettere ad applicazioni di colloquiare tra loro, usando workaround tecnologici. Secondo quell'approccio, le interfacce venivano realizzate ad hoc e non vi erano mai garanzie di completezza, documentazione, e soprattutto di apertura verso terze parti.

Pensiamo ad esempio a importanti soluzioni applicative Cloud come SalesForce, Facebook, Linkedin (e nel passato MySpace): solo attraverso una completa e aggiornata documentazione delle loro interfacce di collegamento (Application Protocol Interface-API) è consentito a qualunque software terzo (sviluppato con logiche e linguaggi diversi) di interagire con le loro funzionalità avendo garanzia che queste, le API, dureranno stabilmente nel tempo.

Ma cosa sono nello specifico le API? Le API rappresentano un contratto tra due componenti applicative. Esso descrive nel dettaglio le azioni di interfacciamento consentite ed i requisiti/vincoli per utilizzarle. Tale "impegno contrattuale" prevede che qualora vi fossero evoluzioni delle API, vi sia un adeguato intervallo temporale durante il quale le vecchie API restano funzionanti ma in uno stato "deprecato" e il software terzo si possa adeguare cosi alle nuove funzionalità e ai nuovi vincoli/requisiti dichiarati.

Le API possono essere:

- **tecniche**, sotto forma di Client Library, che consentono lo sviluppo di estensioni applicative; plugin che vengono eseguiti in un contesto controllato dall'applicazione principale;
- **di integrazione** che consentono il dialogo tra attori esterni e distinti, che possono essere sviluppati con un diverso linguaggio di programmazione.

Perché è importante avere sia API tecniche che di integrazione?

Le API tecniche sono utilizzate laddove il dialogo tra le applicazioni possa implicare costi troppo elevati. In alcuni casi infatti, la logica business potrebbe richiedere "on edge" una mole informativa elevata che dovrebbe transitare nella rete con la possibilità di problemi legati alla sicurezza, alla banda e al volume di dati. Ecco quindi che l'App verticale (sempre nell'esempio, l'App dedicata alla gestione del rischio) potrà lasciare parte della business logic in un "plugin applicativo" installato nel Trusted Data

Layer, che permetterà un accesso privilegiato, sicuro ed ottimizzato ai dati; ciò potrà produrre informazioni fruibili da più App (fattore comune) e sfruttare la scalabilità del Trusted Data Layer.
Le API di integrazione curano invece lo scambio di dati tra App diverse. Esistono diversi standard di definizione e documentazione delle API di integrazione, i più comuni sono OpenAPI e GraphQL. Entrambi gli standard hanno vantaggi e svantaggi che li rendono non mutualmente esclusivi. Con OpenApi si descrivono delle API REST puntuali, con una struttura rigida e per questo ottimizzata per lo scopo. La fase di apprendimento è breve soprattutto se le diverse API sono realizzate sfruttando un naming ed un pattern condiviso. GraphQL rappresenta più un linguaggio di integrazione che permette di ottenere solo le informazioni necessarie, riducendo il numero di interazioni tra le parti ed il traffico di rete, a volte con un sovraccarico della componente back-end. Come detto entrambi gli standard prevedono la documentazione come passo fondamentale nella realizzazione dell'API stessa.

Digital Experience Platform

Per approcciare a queste tecnologie e a questi metodi di fruizione delle informazioni, senza dover iniziare ogni progetto da zero, ci viene in aiuto una nuova categoria di piattaforme digitali che stanno sotto l'acronimo di DXP, ovvero Digital Experience Platform. Una DXP è una piattaforma software aziendale che ha come missione quella di offrire una migliore esperienza utente.

Gartner definisce una piattaforma di esperienza digitale (DXP) come un insieme integrato di tecnologie, basato su una piattaforma comune, che fornisce a un'ampia gamma di pubblico un accesso coerente, sicuro e personalizzato alle informazioni e alle applicazioni attraverso molti punti di contatto digitali. Le DXP gestiscono il livello di presentazione in base al ruolo, ai privilegi di sicurezza e alle preferenze di un individuo, combinano e coordinano applicazioni, tra cui gestione dei contenuti, ricerca e navigazione, personalizzazione, integrazione e aggregazione, collaborazione, flusso di lavoro, analisi, supporto mobile e multicanale.

Le DXP possono essere composte da un unico prodotto o da una suite di programmi che interagiscono, e forniscono alle aziende un'architettura per digitalizzare le operazioni di business, fornire esperienze utente connesse ed ottenere una vera e propria customer insight.

Elementi base del Trusted Data Layer

Una volta introdotti i fondamenti alla base del Trusted Data Layer (Open Source, API e DXP) andiamo ad esplorare i passi tecnici per la sua realizzazione.

Approccio a Prodotto

Negli scenari in cui è necessario raccogliere grandi volumi dati provenienti da fonti diverse uno degli strumenti che viene messo in campo è il Data Lake. Esistono diverse soluzioni Data Lake, che hanno seguito un percorso evolutivo nel tempo e che comunque sono accomunate da un approccio general purpose. Digital Lake, pur ispirandosi alle soluzioni Data Lake, introduce elementi che lo avvicinano ad un approccio a prodotto che garantisce rilasci definiti a salvaguardia delle Business App realizzate.

Figura 2 – Blocchi logici architetturali di cui è composto un TDL.

Architettura

Prima di approfondire gli elementi che compongono un Trusted Data Layer, guardiamone dall'alto l'architettura tipica, che come anticipato si ispira alle soluzioni Data Lake. Essa prevede i seguenti blocchi logici: Data Sources, Data Ingestion, Storage Tier, Data Processing & Enrichment, Data Analysis & Exploration. Per "**Data Sources**" si intendono le fonti dati aziendali e non che contengono il sapere che si ritiene utile far confluire nel Trusted Data Layer per fargli assumere il ruolo di "Single Source Of Truth". Sono eterogenei e tipicamente lontani dal contenere solo dati nella qualità attesa.

Con "**Data Ingestion**" si intendono gli strumenti e/o i processi che "travasano" i dati dai singoli "Data Source" al sistema di storage del Trusted Data Layer. Oltre a trasportare i dati, è possibile anche agire con delle azioni di "risanamento".
Con "**Object Store**" si intende l'ambiente dove i dati sono gestiti come veri e propri oggetti. Ciascuno pensato come un pacchetto informativo che al suo interno contiene dati, ma anche tutti i metadati liberamente selezionabili e un ID univoco di ricerca che consente di avere una struttura piatta di memorizzazione.
Il "**Data Processing & Enrichment**" è l'insieme dei processi che hanno il compito di effettuare trasformazioni ed arricchimenti. Leggono da "Object Store" e persistono in "Object Store".
La "**Data Analysis & Exploration**" può essere parte del Trusted Data Layer, ma tipicamente, si tratta di una o più applicazioni esterne (Business App) le quali possono delegare al Trusted Data Layer una quota parte della logica di back-end.

Procediamo ora in un approfondimento di alcuni concetti:

- la standardizzazione dello schema dati (Common Data Model);
- la fase di ingestione dei dati (Data Ingestion);
- la memorizzazione dei dati (Data Lake);

Data Lake

Definizione e breve storia

Il Data Lake è un sistema o un contenitore che memorizza i dati nel loro formato grezzo e ne fornisce l'accesso sia da codice programma (API) sia tramite query SQL per gli scopi necessari: Business App di natura operativa, decisionale, analisi statistiche, AI, etc. Le informazioni salvate nel Data Lake possono includere dati strutturati provenienti da database relazionali (righe e colonne), dati semi-strutturati (CSV, logs, xml, json), dati destrutturati (email, documenti, pdf) e dati binary (immagini, audio, video).
Lo scopo primario del Data Lake è di diventare l'unica fonte della verità ("Single Source of Truth" o SSOT). Ovvero di contenere informazioni attendibili e quindi: **Corrette, Consistenti, Complete**.

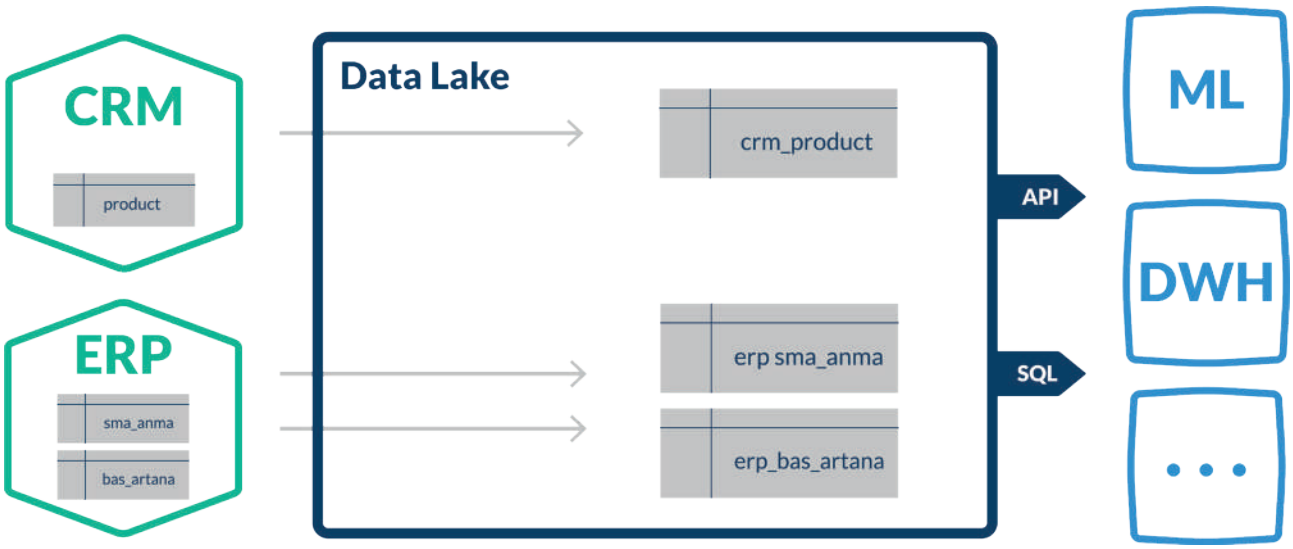


Figura 3 – Schema base architetturale di un Data Lake

Evoluzione

È opinione diffusa che la prima generazione di Data Lake abbia fallito i suoi obiettivi, essa rispondeva infatti solo al paradigma di contenitore di dati grezzi con il risultato che:

- nel Data Lake confluivano anche dati non necessari o non del tutto corretti;
- tutte le logiche di filtro e di arricchimento del dato erano demandate alle singole App;
- la quantità di dati che transitavano dal Data Lake alle App era elevata richiedendo, per il Data Lake, una infrastruttura hardware corposa e costosa.

La seconda generazione di Data Lake ha cercato di rispondere meglio al paradigma di "**contenitore di dati attendibili**" attraverso un approccio: "Virtual DataSet" e/o "Storage Stages".

Virtual DataSet

Rappresenta un approccio al Data Lake attraverso piattaforme di Data Virtualization che, attraverso viste logiche al database, operano per semplificare l'approccio alla banca dati e il reperimento dei dati. Alcune piattaforme (come ad esempio Dremio) sfruttano il concetto di "Virtual Dataset"; una specie di vista semplificata al database attraverso la quale è possibile applicare regole di aggregazione e trasformazione dei dati sorgente per ottenere informazioni finali utili alle App, mascherando la complessità del database in termini di disegno e volume dei dati.

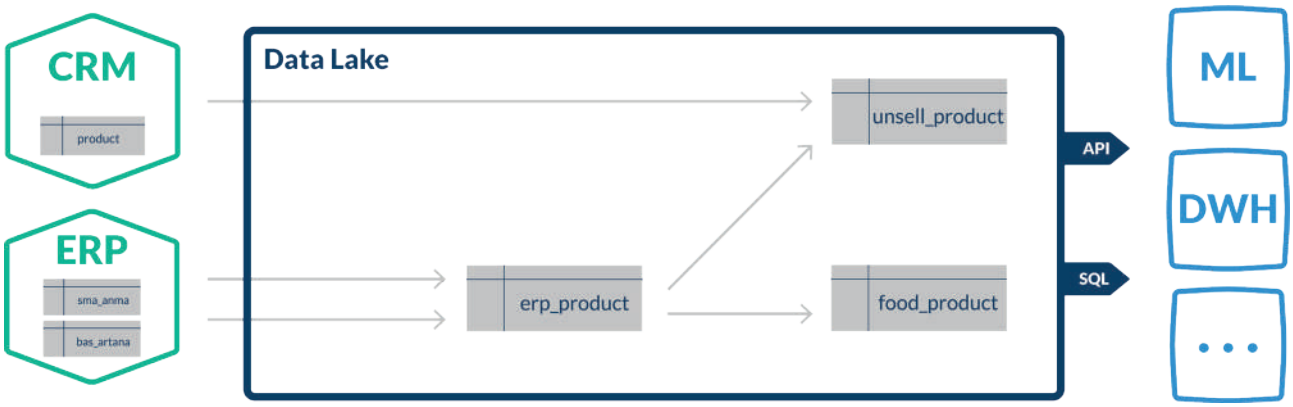


Figura 4 – Vista logica in un Data Lake

Sebbene si possano creare Virtual Dataset da altri Virtual Dataset, le logiche che possiamo gestire sono talvolta limitate o molto onerose in termini di complessità e di trasferimento dati.
Non di rado si necessita di strumenti esterni (Spark, Kafka, etc.) per creare stadi intermedi con dati più affidabili, oppure per gestire i remapping dei valori. Queste piattaforme hanno il grande pregio di attingere ai dati sorgente in modo diretto, senza duplicarli al loro interno. Questo può avere comunque implicazioni in termini:

- **di integrità** - posso fondere dati provenienti da due query eseguite in momenti diversi che riflettono commit diversi sul database sorgente;
- **di performance** - sebbene siano previsti meccanismi di caching, la piattaforma di Data Virtualization sollecita la fonte contaminandosi e contaminandone le performance.

Storage Stages

Un diverso approccio (potendolo mixare al precedente) è rappresentato dal sistema ad "Aree di Stoccaggio" (Storage Stages).

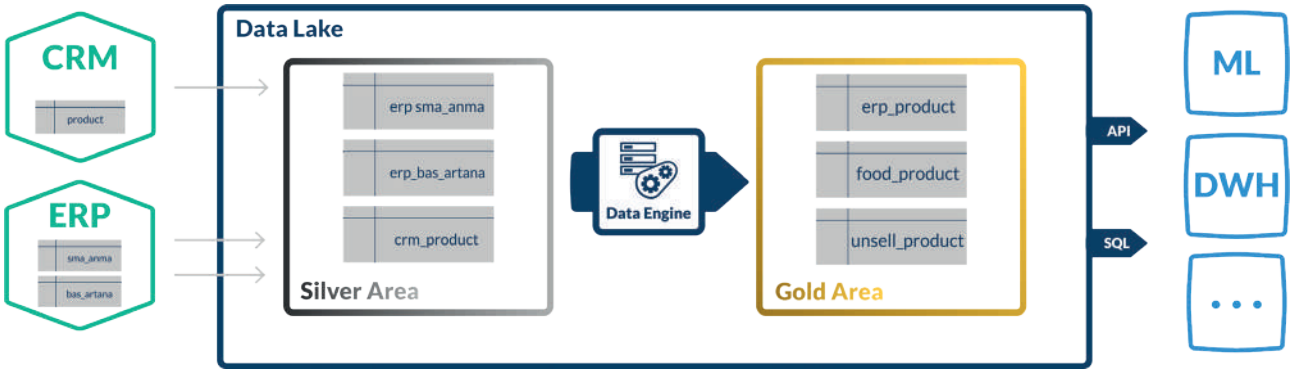


Figura 5 – Storage Stages in un Data Lake

In questo caso l'approccio alla memorizzazione del dato all'interno del Data Lake avviene secondo stadi ed arricchimenti successivi. Infatti:

- i dati grezzi vengono storicizzati in una prima area di storicizzazione chiamata "**Silver Area**";
- specifici processi (task, o job) si occupano di analizzare questi dati applicando filtri di trasformazione necessari per renderli affidabili ed omogenei;
- i dati così manipolati ed arricchiti vengono storicizzati in una nuova area chiamata "**Gold Area**".

La "Silver Area" può assumere anche una visione "generazionale", per gestire la storia delle ingestioni:

- in modo da evitare il trattamento di ingestioni parziali;
- per consentire logiche di trasformazione anche su cambiamenti storici.

Common Data Model

Nei capitoli precedenti abbiamo compreso gli elementi tecnologici che sostengono l'acquisizione e la conservazione di dati (fisici o virtuali) nel Trusted Data Layer dell'impresa, e abbiamo analizzato le tecnologie che si possono utilizzare perché sia garantita l'accessibilità ai dati per il loro successivo utilizzo. Ora è importante spostare il focus verso gli aspetti più attinenti la completezza e la interpretabilità dei dati presenti nel Trusted Data Layer.

Il Common Data Model (CDM) è un **contratto sui dati** condiviso da uno o più attori. Il suo obiettivo è definire le strutture dati che rappresentano il punto di partenza e/o il punto di arrivo di una applicazione, di un processo, di una attività.

Poiché chiunque può definire un "CDM", e perchè il perimetro del modello dati può essere definito secondo necessità, non esiste un CDM in assoluto migliore, ma va valutato e scelto rispetto gli obiettivi che ci si pone. Alcuni CDM interessanti sono:

- "**schema.org**"
- "**OpenInitiative Common Data Model**"
- "**IBM Tivoli Common Data Model**"
- "**Observational Health Data Sciences and Informatics**"

Standard di rappresentazione dei dati

Ancora una volta, facciamo riferimento all'esempio dell'applicazione di valutazione del rischio di fornitura: abbiamo capito come riportare nel Trusted Data Layer i dati necessari per il funzionamento delle App e come accedervi in modo semplice tramite API. Ma come possiamo avere garanzia che i dati esposti verso le applicazioni siano completi e correttamente interpretati rispetto a quello che l'applicazione / algoritmo si aspetta per calcolare la probabilità di accadimento di un evento o il costo atteso dello stesso? E come potrebbe una azienda utente scegliere, in un ipotetico App Store, quella App che sia più precisa, completa, usabile ed economica e con la certezza che vi sarebbe la stessa interoperabilità applicativa che oggi sperimentiamo nel mondo delle applicazioni mobile?

L'approccio di trasferire / virtualizzare nel Trusted Data Layer tutte le informazioni di cui l'impresa dispone, ed il ricorso ad API ben documentate e esposte secondo standard, consente di raggiungere il primo obiettivo, ovvero, quello dell'accesso al dato, ma non il secondo, perché le API non sono di per sé garanzia di completezza delle informazioni disponibili o di interoperabilità universale tra applicazioni.
Per raggiungere il secondo obiettivo è necessario introdurre un alto livello di standardizzazione nei dati gestiti nel Trusted Data Layer (master data, dati transazionali, dati catturati dal mondo fisico, o da internet) in modo tale che diversi sviluppatori possano "contare" su un substrato informativo completo e standard e far nascere così una App economy, anche nel contesto industriale manifatturiero.

Relativamente alle prime due categorie, ovvero master data e transazioni, sono stati fatti in passato numerosi tentativi di definire degli standard sufficientemente completi. Alcuni di questi sono stati ricompresi in standard de facto. Relativamente ai dati provenienti dal mondo fisico del prodotto (Internet of Things) o del processo (Industry 4.0) connesso, essendo relativamente recente questa materia, questo lavoro è in gran parte da fare mentre per i dati accessibili in internet (open data, data broker, etc.)

la situazione è nel mezzo, con alcuni comparti già dotati di data model consolidati (e.g. weather forecast, quotazioni commodities) ed altri ancora da sviluppare.

Caratteristiche e Architettura

Il Common Data Model descrive le entità che rappresentano la struttura informativa gestita, ovvero necessaria alle applicazioni verticali realizzate per settore merceologico per rispondere alle esigenze delle imprese. Si tratta sia di entità grezze sia di entità derivate da esse (macro-entità, tabelle di sintesi, etc.). Il disegno della struttura dati è frutto di un importante lavoro collaborativo, tra una parte consulenziale/business e una parte tecnica ed il risultato può essere considerato "**statico**". Statico non significa che non possa subire estensioni ed aggiustamenti ma questi devono essere trattati con un approccio "a rilascio", ovvero non possono essere prodotti o applicati in maniera libera, ma attraverso un iter di consolidamento, poiché le evoluzioni del Common Data Model implicano sempre un intervento software. Questo è un aspetto architetturale "accettato" per poter garantire la "stabilità" della struttura dati alle applicazioni verticali che ne fruiscono.

Dal punto di vista architetturale il CDM rappresenta:

- il punto di arrivo dell'ingestione;
- il punto di partenza per la fruizione.

Questo significa che ogni entità informativa da memorizzare dovrà essere:

- creata nel database nel rispetto delle specifiche;
- gestita da API (Rest / GraphQL) per recuperare le sue informazioni come da specifiche;
- pronta per consentire una fruizione via SQL in sola lettura;
- protetta affinché la sola componente di ingestione possa agire per la variazione dei dati.

Data Ingestion

Il CDM descrive un proprio set di informazioni la cui struttura non è sovrapponibile (non ha corrispondenze) alle entità delle fonti dati sorgente. In questo capitolo diamo visione delle tecniche che permettono di operare affinché il Trusted Data Layer sia popolato di informazioni di valore attraverso l'azione di Data Ingestion.

Iniziamo comprendendo che l'ingestione può essere:

- **Attiva o Passiva**: nel primo caso nel Trusted Data Layer è presente un componente in grado di recuperare i dati dalla fonte sorgente, mentre nel secondo un componente nel Trusted Data Layer espone una API sfruttata dalla fonte sorgente per fornire i dati;
- **Totale o Incrementale**: nel primo caso l'ingestione coinvolge tutti i dati presenti nelle diverse fonti sorgente, nel secondo l'ingestione è in grado di isolare solo i dati variati rispetto ad un istante di riferimento;
- **Periodica o Puntuale**: nel primo caso i dati vengono recuperati o forniti ad intervalli più o meno regolari, nel secondo i dati sono recuperati nell'istante successivo ad un cambiamento (add, update, delete).

Il motore dell'ingestione

Sin dal primo White Paper abbiamo compreso l'importanza di affrontare ben preparati la grande sfida della persistenza e fruizione di grandi moli informative. Ecco allora che un elemento fondamentale per ottenere un utile Trusted Data Layer (avendo sempre chiara la visione del processo di Business) è il motore di ingestione dei dati.

Tale elemento non si palesa in un solo momento della vita delle informazioni presenti nel nostro Data Lake, anzi, diventa un potente e vivo architrave che sa continuamente migliorare la fruibilità delle informazioni affinché le Business App diano il massimo nella loro fruizione.

Approfondiremo questo argomento nel prossimo White Paper, ma è indubbio che elementi come qualità, fruibilità, semplicità (e molte altre caratteristiche) del dato sono essenziali. È come avere dell'oro poco "indossabile" per un ricevimento. È qui che, come un fine orafo, il motore della Data Ingestion opera per fornire alle Business App dati altamente usabili.

Storage Stages

Sapendo che il CDM è il punto di arrivo, è necessario che il Trusted Data Layer disponga di un "database" strutturato per agevolare l'ingestione. La strutturazione è descritta come un processo di raffinamento successivo del dato, che attraversa vari stadi di lavorazione.

Bronze Stage

Questa fase dell'ingestione è costituita dai seguenti elementi informativi (qui elencati con nomenclatura tecnica): OriginTable, ImportProcess, RecordCheck, RecordData. Attraverso questo set di elementi è possibile modellare una importazione attiva incrementale anche se la sorgente non è in grado di gestirla.
Sarà comunque necessario:

- creare un tool specifico per la fonte (Driver);
- recuperare tutti gli elementi di tutte le basi informative (o tutti quelli modificati dopo una certa data);
- verificare se il dato è variato;
- serializzare i record variati (nuovi, modificati, rimossi) da gestire in "RecordData";
- gestire le dipendenze del record per assicurarsi che l'ingestione non comprometta la consistenza;
- attivare la fase di consolidamento dei cambiamenti all'interno del CDM Storage.

Silver Stage

Nella logica Data Lake la "Silver Area" rappresenta l'area dove i dati vengono puliti e predisposti per il passaggio nella Gold Area. Nel Trusted Data Layer la Bronze e Silver Area si sovrappongono in quanto la maggior parte delle azioni di ingestione sono realizzate attraverso componenti che conoscono sia la sorgente che il Common Data Model di arrivo. Ci sono comunque aspetti da gestire come la fase di filtro sui dati in ingresso, la normalizzazione dei valori, la presenza dello stesso dato in più sorgenti, etc. La risoluzione delle controversie può richiedere la definizione di "aree di mapping" che indichi quale fonte vince per ogni entità o gruppo di entità.

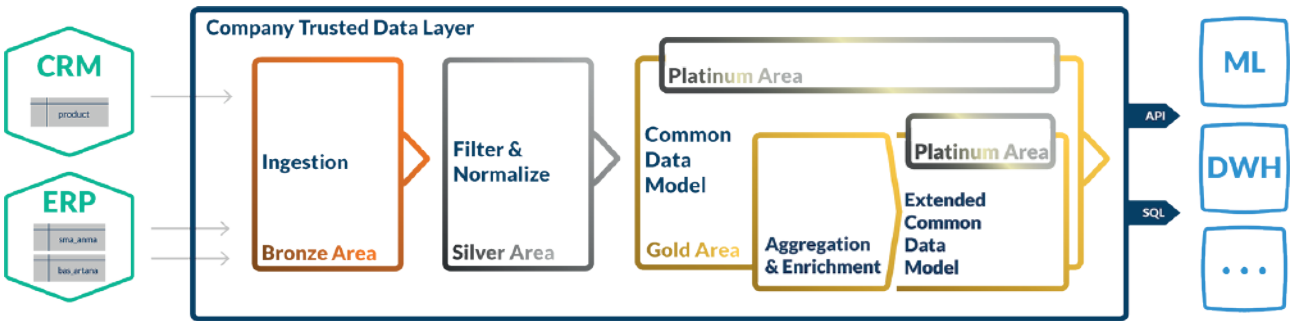


Figura 6 – Common Data Model in TDL

Gold Stage

Nella fase Bronze i dati vengono ingestiti e parcheggiati in un'area di quarantena; nella fase Silver dove i dati sono recuperati dalla quarantena, filtrati, normalizzati ed infine usati per popolare/aggiornare la Gold Area del Common Data model, la quale rappresenta la prima area di collegamento con le Businnes App. I dati sono stati elaborati e pronti ad essere fruiti nello stato più elementare.

Non si devono escludere comunque altri Stage utili a contenere informazioni pronte al consumo delle Business App come:

- una o più **Extended Common Data Model** come fase di ulteriore arricchimento e persistenza di informazioni;
- una o più **Platinum Area** come fase di trasformazione da un Physical Common Data Model verso un Logical (exposed) Common Data Model che possa contenere informazioni con metriche aziendali specifiche.

Le scelte di Digital Lake

Ripercorriamo ora le scelte adottate nella realizzazione di Digital Lake secondo le direttrici in cui ci siamo mossi nei precedenti capitoli. Lo faremo quindi descrivendo:

- le scelte tecnologiche;
- la piattaforma digitale individuata;
- il disegno della base informativa standard;
- il motore di ingestione utilizzato.

Tecnologie Abilitanti

Elenchiamo alcune scelte tecnologiche che sono state adottate per la realizzazione di Digital Lake. L'obiettivo è quello di darne evidenza per consolidare concetti e visioni descritte nei capitoli precedenti senza necessariamente entrare in complessi tecnicismi.

Java

Digital Lake è sviluppato in Java, un linguaggio di programmazione maturo, stabile, versatile, nativamente multi-thread[1].
Java dispone di un ecosistema di librerie e soluzioni che ne abilitano l'utilizzo negli ambiti più disparati.
L'evoluzione si basa sulla preventiva definizione di specifiche (API) implementate da diversi Vendors (importante nell'ottica open di No-Lock-in). Solo a titolo di esempio ricordiamo che la User Interface di Android e le App per Android sono realizzate in Java.

OSGi

L'Open Service Gateway initiative (OSGi) è una specifica ed una tecnologia dell'ecosistema Java che consente lo sviluppo di applicazioni modulari, basate su API e servizi (anche micro). L'aumento della complessità in un prodotto software, sia esso embedded, client o server, richiede codice modulare, ma anche sistemi che siano estensibili dinamicamente. Grazie ad OSGi l'applicazione può essere evoluta ed estesa in modo agevole, e quindi scomposta e distribuita su più unità elaborative. Digital Lake usa OSGi su Apache Karaf (ambiente di runtime) per ottenere una suddivisione omogenea delle funzionalità tecniche in "servizi" che possono essere scalati orizzontalmente e verticalmente. Ancora a titolo di esempio OSGi e Karaf sono le tecnologie di base utilizzate da Netflix.

Multi model database

Il contesto Digital Lake è per definizione un contesto che ci proietta verso la gestione di una mole informativa di tipo "Big Data", gestita generalmente da Database NoSQL. Ma come detto Digital Lake guarda in primis al mondo Enterprise che genera dati di tipo relazionale e, pur arricchendoli di informazioni IoT/OT, la loro fruizione sarà prevalentemente di tipo relazionale.

Digital Lake è stato pertanto pensato per operare con diversi modelli di Base Dati, ed opera con database multimodel, come ad esempio OrientDB, che consentono di gestire basi informative attraverso tabelle con colonne variabili, all'interno delle quali è possibile utilizzare sia relazioni tradizionali (foreign-key) che più moderne ed evolute relazioni a grafo. Operare con OrientDB permette di gestire grandi quantità di dati esplorabili in modo agevole con ottimi livelli di performance.

Apache Calcite

L'evoluzione delle tecnologie back-end e front-end non sempre cresce in maniera organica. Molti applicativi front-end, ad esempio nell'ambito della tecnologia Business Intelligence (BI), operano verso banche dati transazionali. Apache Calcite è una tecnologia che aiuta l'integrazione di questi strumenti di BI fornendo loro una rappresentazione "virtuale" del database reale. L'obiettivo è limitare il costo elaborativo in termini di CPU, rete, letture disco, etc. che gli strumenti BI abbisognano. Tale metodo di "astrazione" viene utilizzato anche nella fase di ingestione dei dati per agevolare/ridurre le interazioni con la base informativa.

1 La programmazione multi-thread è un paradigma di sviluppo che permette di eseguire più sottoprocessi in parallelo. È anche detta programmazione concorrente, e permette di ridurre il tempo di esecuzione e la complessità temporale.

La piattaforma scelta: Liferay DXP

Nella soluzione Digital Lake è stata fatta una scelta importante a fondamento della fase di amministrazione della piattaforma, della gestione dei flussi di ingestione, della fruizione sicura dei dati in un paradigma integrato dell'esperienze utente, attraverso la scelta di Liferay DXP.
Liferay è una straordinaria piattaforma, che Gartner da 11 anni annovera consistentemente nel quadrante delle piattaforme Leader nella Digital Transformation.

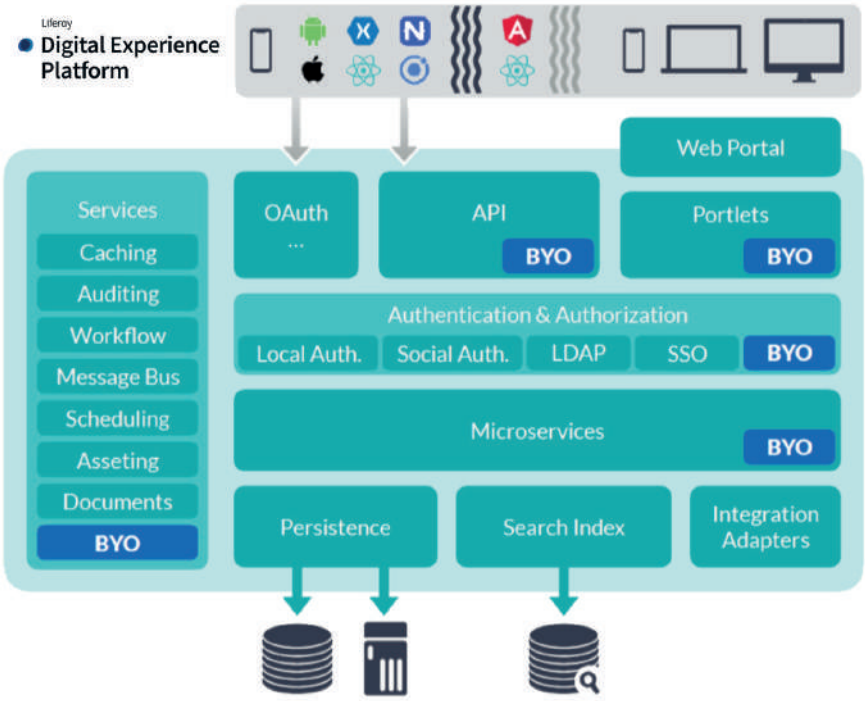


Figura 7 – Schema Architetturale di Liferay DXP

Alcune caratteristiche di Liferay DXP

La piattaforma Liferay, nata inizialmente per rispondere alle soluzioni di tipo Portali Trasversali, con l'uscita della release DXP ha virato completamente verso una piattaforma di Digital Transformation; senza perdere le origini da cui era nata, ma evolvendo fortemente le funzionalità utente centriche.

Ci soffermiamo ora su alcune caratteristiche che fanno di Liferay una piattaforma di trasformazione digitale:

- **Experience Management**: consente di creare librerie di contenuti dedicati a specifici canali riutilizzabili da App esterne;
- **Content Management**: consente di gestire contenuti informativi diversificati ed integrarli con sistemi di Artificial Intelligence/Machine Learning per ottimizzare al massimo le esperienze utente;
- **Business Operations**: abilita la costruzione di App personalizzate utilizzando un generatore di codice di basso livello; automatizza le operazioni dei processi aziendali con App facili da creare basate su workflow;
- **Headless**: permette di creare esperienze tradizionali, headless e multicanale con un approccio API (OpenAPI e GraphQL) per accelerare il processo di fruizione;
- **Identity Management and Access Control**: consente una attenta e protetta modalità di gestione delle informazioni a salvaguardia del patrimonio informativo;
- **Platform Services**: consente un facile accesso alle applicazioni cross-site e

monitorare l'utilizzo delle risorse a livello di sistema, ospitando un numero qualsiasi di istanze separate logicamente configurando il sistema per ognuna di queste;
- **ElasticSearch**: incorporato all'interno della piattaforma, questo importante motore di ricerca è basato su Lucene, con capacità di ricerca full text, con supporto ad architetture distribuite. Tutte le funzionalità sono nativamente esposte tramite interfaccia RESTful, mentre le informazioni sono gestite come documenti JSON.

Liferay DXP per Digital Lake

Ecco allora la scelta di porre come fondamenta solide di Digital Lake una piattaforma robusta, completa e moderna con cui è possibile integrare sia le diverse applicazioni aziendali che alimentano il Trusted Data Layer, sia le Business App che ne sfruttano il contenuto informativo. Liferay DXP sfrutta appieno la tecnologia OSGi, che gli consente espandibilità e scalabilità, ed agevola la gestione dell'ambiente di sviluppo e degli ambienti di test e di produzione garantendo una solida dorsale su cui poggiare il patrimonio informativo aziendale. Tutta la componente Headless garantirà la fruizione nel tempo del catalogo App che potrà essere realizzato con gli strumenti già messi a disposizione dalla piattaforma utilizzando moderne tecnologie di front-end (React, Angular, etc) o con strumenti esterni.

IoT Experience

Rappresenta la soluzione che risponde alle esigenze IoT delle industrie manifatturiere. Sviluppata da SMC appoggiandosi alla soluzione Liferay DXP consente la gestione delle informazioni provenienti dal mondo della fabbrica, dai prodotti connessi (smart-produtcs) e da tutte le informazioni provenienti dai sensori (energia, gas, video-sorveglianza, contapersone, etc.) e le interseca con informazioni di "contesto" provenienti dal mondo (ERP/MES/CRM/Etc.) tradizionale.
IoT Experience è l'ulteriore elemento differenziante rispetto all'approccio Data Lake classico, in quanto fornisce due aspetti utili ad una visione a prodotto di Digital Lake:

- di natura Architetturale: questa soluzione incorpora i paradigmi precedentemente descritti in uno schema architetturale che diventa base informativa pronta per le Business App;
- di natura Tecnologica: i dati IoT sono una delle fonti dati del Trusted Data Layer, e Digital Lake incorpora l'IoT Experience per la completa gestione della base informativa.

Il CDM per l'azienda manifatturiera

Secondo il principio del Common Data Model, nel Trusted Data Layer andrebbero rimappate tutte le informazioni tipiche di una azienda, e di una azienda manifatturiera in particolare, in modo tale che esse siano correttamente accessibili.
Sono stati fatti numerosi tentativi di definire degli standard sufficientemente completi; sin dalla metà degli anni 2000 si sono tentate delle standardizzazioni e delle ontologie dei processi manifatturieri (cfr. standard ISO), di recente richiamate anche da modelli dati e visioni legati all' Industry 4.0 (cfr. RAMI 4.0).

Il problema principale di questi tentativi è il difficilissimo equilibrio tra l'omni-comprensività (da un lato) e la generalità (dall'altro), in un compromesso che le ha rese, di fatto, inapplicabili. Va anche aggiunto, in una chiave di lettura storica, che la forte competizione funzionale tra i diversi software vendor per il mercato manifatturiero, a cavallo degli anni 2000, portava i vendor stessi a sviluppare modelli dati capaci di

cogliere anche minime peculiarità di processo, o enfatizzare features offerte dalle ultime tecnologie, e a discostarsi così dagli standard.

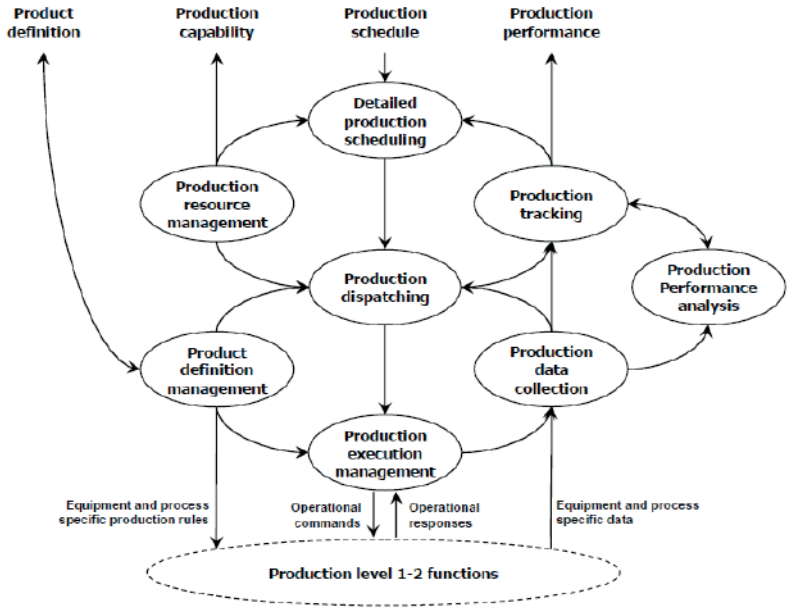


Figura 8 – Esempio di Manufacturing Data Model, © IEC62264-3

Dalla piena maturità di internet, dall'avvento del cloud e con il maturarsi della App Economy, i limiti di un approccio completamente focalizzato sulla performance funzionale di un applicativo sono emersi con chiarezza, a vantaggio di un approccio in cui il dato sia più accessibile, documentato, e a disposizione di un ecosistema di sviluppatori perché il valore generato dal dato stesso possa evolvere secondo i principi dell'Open Innovation.

Forti di questa lettura storica e strategica, quello che è stato fatto nel progetto del Trusted Data Layer di "**Digital Lake**" è stato partire dalla trentennale esperienza di SMC nei software manifatturieri e dalla decennale esperienza di Iq Consulting nell'ottimizzazione di processo, confrontarsi con la mappatura logica dei data model disponibili nei documenti di standardizzazione esistenti, e quindi completarli con processi importanti e solitamente trascurati, e.g. il processo di product design guidato da soluzioni di PDM / CAD[1]. L'ultimo passo è stato, in ossequio ai principi dell'Open Software, documentare i dati posti o virtualizzati nel Trusted Data Layer, di modo tale che sviluppatori (interni o esterni) possano accedere con facilità a tali informazioni.

La soluzione "Digital Lake" fornisce dunque un Common Data Model già **consapevole** delle esigenze che hanno le aziende manifatturiere, radicato nella conoscenza codificata negli standard, ma aumentato alla luce dell'esperienza di gestione e miglioramento di processo: esso combina l'approccio accademico di Iq Consulting al Common Data Model e l'esperienza SMC nell'ambito del software manifatturiero e delle componenti IoT, per fornire un prodotto ed una piattaforma già orientati alle esigenze delle imprese: una soluzione perciò in grado sia di ospitare le informazioni aziendali, sia di essere fonte abilitante per applicazioni verticali.

1 Product Data Management (PD), Computer Aided Design (CAD)

Device Connector Framework

DCF è il modulo della soluzione IoT Experience motore della Data Ingestion per il caricamento, arricchimento e persistenza delle informazioni. Attraverso DCF è possibile disegnare i processi di acquisizione dati da qualsivoglia fonte (sia essa di tipo IT transazionale, M2M / IoT, Internet) e attivare le azioni di trasformazione, validazione correlazione, arricchimento necessari al CDM esteso. DCF risulta il cuore pulsante del Trusted Data Layer che sa "far fluire" in modo ottimale le informazioni aziendali nei diversi stages della banca informativa e nei diversi spazi di arricchimento informativo per le App. Opera collegandosi alle diverse forme di fruizione dei dati come:

- **protocolli** (MQTT, ModBus, OPC, etc.);
- **piattaforme** (Niagara, Kura, etc.);
- **ambienti** (Azure, AWS, Google).

DCF affronta la sfida dell'ingestione dei dati (Velocità, Variabilità, Volumi, Qualità, Scalabilità) attraverso le funzionalità di: Gateway (gestione del collegamento delle tre forme di fruizione dati su descritte), Message Queue (per gestire "ordinatamente" enormi moli di dati), Processor (nella fase di riconoscimento e instradamento), Mangling (nella fase di normalizzazione e arricchimento), Persistence (per la storicizzazione delle informazioni).
Il Device Connector Framework è inoltre il modulo che si occupa di gestire tutta la banca informativa di Digital Lake nei diversi stadi dell'ingestione e nella preparazione delle informazioni utili alle Business App.

Un esempio di applicazione

Un paradigma come quello fino ad ora descritto in tutte le sue sfaccettature presenta inevitabilmente numerosi vantaggi per chi, all'interno delle aziende, vuole far crescere il peso di logiche Data Driven nel prendere le proprie decisioni: nella nostra esperienza, basarsi sui dati, seppur coniugandoli all'esperienza e alle condizioni di contesto, è il modo migliore per rispondere alle complesse problematiche di gestione che affrontano oggi le imprese, quelle manifatturiere in particolare.
La qualità dei dati alla base è l'elemento fondamentale affinché, come recita il principio **garbage in, garbage out,** l'output sia il più affidabile possibile.

Immaginiamo, ad esempio, una decisione di materials management, ovvero definire le logiche di riapprovvigionamento (quando e quanto riacquistare) di uno specifico componente: fino a ieri l'impresa avrebbe avviato uno studio ad hoc (o un progetto, se la decisione avesse riguardato una completa revisione delle politiche di materials management di tutti i componenti), coinvolgendo numerose figure aziendali al fine di dare una risposta al problema, che sarebbe diventata "vecchia" di lì a poco, o per lo meno poco fruibile in una seconda occasione, o su una mole di dati più significativa.

A titolo puramente esemplificativo, proviamo a riassumere, nella seguente tabella, i principali stakeholder di un tale progetto, gli obiettivi di ciascun coinvolgimento e le attività da svolgere:

Funzione coinvolta	Obiettivo	Attività
Vendite	Quantificazione dei volumi attesi di vendita dei prodotti finiti che impiegano quel componente	Analisi delle vendite passate da CRM Analisi dei volumi di previsione da Forecasting Tool
Pianificazione della produzione	Quantificazione dei volumi di produzione pianificata	Accesso alla BoM Estrazione dei dati tecnici di produzione (e.g. tassi di scarto reali) Estrazione da piani di produzione già congelati nel breve termine (Planning tools, o MRP)
Acquisti	Classificazione delle caratteristiche di ciascun componente in termini di costo, criticità, importanza, aleatorietà, consumo. Analisi delle politiche di riordino già implementate, e quindi con processi gestionali già avviati.	Estrazione dell'anagrafica da ERP Estrazione delle caratteristiche di approvvigionamento da sistemi di gestione dei fornitori
IT	Supporto al recupero del dato	Sviluppo query per l'estrazione dei dati da sistemi diversi

Tabella 1: Esempio di processo e relative funzioni volto a identificare i corretti approcci di acquisto dei materiali

Con l'architettura descritta in precedenza, disponendo di un Common Data Model ben documentato, le attività elencate in tabella 1 diventano tutte incredibilmente più semplici, e talune possono essere automatizzate. Non solo: dal momento che le informazioni possono essere facilmente reperite, risorse addizionali possono essere dedicate al miglioramento del metodo di decisione, o a sviluppare strumenti che consentano di simulare anche in tempo reale vantaggi e svantaggi di uno o dell'altro approccio di pianificazione dei materiali, allineando la propria realtà aziendale alla sempre maggiore flessibilità che i mercati chiedono. Grazie all'apertura del dato, diverse realtà esterne possono essere messe a confronto (università, società di consulenza, vendor tecnologici) per cimentarsi, a partire da quel patrimonio informativo, nella proposta di Business App che siano utili ai problemi dell'azienda, proprio come appoggiandosi sulla nostra rubrica, al nostro sensore GPS, alle risorse fisiche del nostro smartphone, sviluppatori di tutto il mondo hanno, con le loro App, migliorato la nostra mobilità, il nostro tempo libero ed il nostro lavoro. Infine, grazie alla persistenza del Common Data Model, la Business App dedicata alla selezione delle logiche di riapprovvigionamento ottimali può essere tenuta costantemente in esecuzione, di modo tale che se vi sia un cambiamento esterno (e.g. incremento di domanda) o interno (e.g. sostituzione di un componente in distinta), l'applicazione possa in modo continuo aggiornare i parametri di gestione del materiale, senza richiedere progetti di consulenza ad hoc e discontinui nel tempo.

Conclusioni

Con queste tecnologie l'architettura che rappresenta la visione del futuro dei sistemi informativi prende forma, diventa robusta alle dinamiche del mercato, rimane aperta, accessibile e sicura.

È il momento di approfondire l'ultimo tassello di Digital Lake, ovvero l'ecosistema di Business App pensate per affiancarsi al lavoro di manager e dipendenti in una visione di Enterprise Business Process. È il tassello più "visibile e fruibile" di quest'opera che porta con sé anche importanti sfide in termini di User eXperience (UX). Di tutto questo parleremo nella terza memoria della serie.

IQ Consulting è il punto di riferimento Coaching & Advisory del Network Digital360.

SMC supporta con competenza le aziende in evoluzione da 40 anni esatti, con sedi in tutto il territorio italiano per garantire accompagnamento e assistenza.

Crediamo profondamente nell'Open Source, quale strumento per raggiungere gli obiettivi aziendali in modo innovativo, efficace e veloce.

SMC

IQ Consulting

SMC è il partner perfetto per la Digital Transformation dell'impresa. Gestisce dati complessi per trasformarli in informazioni vincenti, attraverso soluzioni modulari, integrabili e customizzabili, basate su tecnologie Open Source. SMC affianca il cliente ottimizzando la comunicazione, aumentando la collaborazione tra stakeholder, gestendo l'intera Supply Chain e proponendo soluzioni Industry 4.0. SMC è business partner IBM e Liferay Platinum Partner da oltre 10 anni.

IQ Consulting supporta la direzione aziendale nell'impostazione di strategie per la generazione di valore sostenibile nel lungo termine, attraverso progetti sviluppati in stretta collaborazione con il cliente. Spin-off dell'Università degli Studi di Brescia, l'azienda è in grado di proporre soluzioni ICT solide, innovative, dal valore aggiunto comprovato. IQ Consulting oggi fa parte del Network Digital 360.

www.digitallake.it

in

DIGITAL Lake