



ALMA MATER STUDIORUM A.D. 1088

# UNIVERSITÀ DI BOLOGNA

Scuola

## Ingegneria e Architettura

Corso di studio (laurea magistrale)

## Ingegneria e Scienze informatiche

Sede didattica

Cesena

A.A.

2018/2019

Insegnamento

## Sistemi Informativi

Titolare

Prof. Matteo Golfarelli

## Sommario

Introduzione.....	1
Il sistema informativo .....	2
Ruolo dei sistemi informatici .....	2
Efficacia ed efficienza.....	2
Evoluzione dei sistemi informatici .....	2
Sistemi informativi vs Basi di dati .....	2
Obiettivi del corso .....	3
Figure professionali.....	3
Seminari .....	4
Modalità di esame .....	4
Materiale didattico .....	4
Classificazione dei Sistemi Informativi.....	4
Catena del valore di Porter .....	5
I processi primari.....	5
I processi di supporto e approvvigionamento .....	5
Limiti del modello .....	5
Sistemi informativi operativi.....	7
Sistemi informativi direzionali .....	7
Sistemi informativi analitici.....	8
Sistemi analitici per settori .....	8
Mappa dei Sistemi Informativi.....	8
Un esempio da SAP .....	8
Il portafoglio applicativo aziendale.....	9
Il portafoglio applicativo (un esempio) .....	9
Profilo dei livelli di pianificazione.....	10
Profilo delle attività di esecuzione .....	10
Tipologie di sistemi informativi.....	11
Ingegneria dei processi .....	22
Il ciclo dell'innovazione .....	22
Definizione degli obiettivi .....	23
Innovazione dei Sistemi Informativi: Business Process Reengineering (BPR) .....	24
I processi aziendali .....	25
Linear Responsibility Charting.....	25

---

Balanced scorecard .....	26
Le risorse umane .....	28
Key Performance Indicator .....	28
Proprietà dei KPI .....	29
Esercitazione sui KPI.....	29
Business Process Reengineering .....	31
Passi necessari.....	34
Tecniche BPR e Strumenti analitici .....	34
Information Technology.....	35
Ridisegno dei processi: modello in 6 fasi .....	37
Data Flow Diagram (DFD).....	44
Entità dei DFD .....	45
Sintassi .....	45
Rappresentazione .....	45
Consigli .....	45
Regole .....	45
Vincoli.....	45
Avvertenze .....	46
DFD con flussi di controllo .....	46
DFD multilivello .....	46
Consigli pratici.....	46
Strategie per la costruzione di DFD.....	47
Notazione DFDEditor.....	47
Studio di fattibilità.....	48
Motivazioni e obiettivi .....	48
Struttura di riferimento .....	49
Analisi situazione attuale .....	49
Progetti di massima .....	51
Analisi del rischio .....	52
1. Individuazione dei fattori di rischio.....	52
2. Definizione del rischio di progetto .....	53
3. Modalità di gestione del rischio .....	53
Progetto proposto.....	54
1. Segmentazione del progetto.....	55
2. Riepilogo acquisizioni e realizzazioni previste .....	55

---

3. Modello organizzativo.....	55
4. Piano di progetto .....	56
Analisi costi-benefici .....	66
Stima dei costi.....	67
Individuazione dei benefici .....	69
Raccomandazioni per le fasi realizzative .....	71
Il ciclo di vita di un progetto .....	72
Strategie di acquisizione: “make or buy” .....	72
Outsourcing.....	73
Outsourcing: pro e contro.....	75
Impatto organizzativo .....	75
Compiti e responsabilità della funzione IT.....	76
Fattori chiave e strategie di realizzazione di software applicativo .....	77
Definizione del contratto .....	78
Soggetti coinvolti.....	79
Struttura di un contratto.....	79
Esempio: contratto di manutenzione ed evoluzione dei sistemi informativi.....	79
Integrazione .....	87
Cosa significa integrazione? .....	88
Motivazioni .....	88
Processo di integrazione .....	88
Integrazione dello schema .....	89
Master Data Management (MDM) .....	96
Scegliere i Master Data .....	97
Architetture per Master Data Management.....	98
Seminario: Big Data.....	102
Definizione di Big Data.....	103
Chi genera big data? .....	105
Un’infrastruttura per i dati.....	106
Evoluzione dai DataWareHouse ai Big Data .....	107
Hadoop.....	107
Architetture a confronto.....	107
Scalabilità .....	108
NoHadoop: non solo Hadoop.....	109
Hadoop 1 vs Hadoop 2 .....	109

---

Distribuzioni principali .....	109
Un'analisi disincantata di Hadoop .....	109
Il paradigma Map-Reduce (esempio).....	110
Hadoop 1 vs Hadoop 2 .....	112
Moduli principali .....	113
Aree applicative .....	114
Hadoop Distributed File System .....	114
HDFS Federation .....	115
Topologie di cluster.....	115
Blocchi di dati e repliche .....	115
Località dei dati .....	116
Namespace e Namenode .....	116
High-Availability .....	116
Lettura da HDFS .....	117
Scrittura su HDFS.....	117
Il meccanismo Hear Beat .....	118
YARN .....	118
Application Master.....	118
YARN - Walkthrough .....	119
Hadoop File format .....	119
AVRO .....	119
Parquet.....	120
Storage colonnaire .....	120
Data model.....	120
Unnesting .....	121
Definition level .....	121
Repetition level .....	121
File format.....	123
Unit of parallelization.....	123
Configuration .....	123

## Introduzione

Obiettivi dell'informatica in azienda:

- Aumentare l'efficienza
- Aprire nuove aree di mercato.

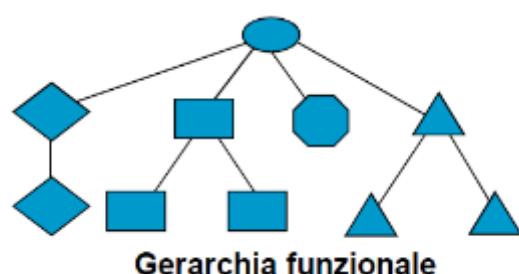
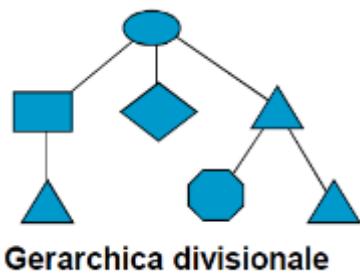
Azienda privata: obiettivi di massimizzazione del ricavo

Azienda pubblica: massimizzare la qualità dei servizi e contenere i costi

Elementi strutturali di un'azienda:

- Obiettivi
- Struttura organizzativa  
Suddivisione dell'impresa in differenti gruppi a cui sono attribuiti compiti, obiettivi e responsabilità
- Processi funzionali

Nella figura si rappresenta l'idea per cui la **gerarchia divisionale** (basata sul mercato o sul prodotto) include, dal punto di vista dei gruppi operativi, diverse funzioni ed è contrapposta alla **gerarchia funzionale** che mantiene separate le competenze (le forme rappresentano competenze omogenee).



La gerarchia funzionale è maggiormente appropriata per aziende di dimensioni piccole o medie.

Nel caso di aziende di grandi dimensioni è maggiormente utile specializzare le competenze per linee di prodotti o aree di mercato.

Spesso si sfrutta un'organizzazione a matrice progetti/funzionalità:

	Funzione 1	Funzione 2	Funzione 3	Funzione 4
	Analista	Sistemista	DBAdministrator	Sviluppatore
Progetto 1	FTE	FTE	FTE	FTE
Progetto 2	FTE	FTE	FTE	FTE
Progetto 3	FTE	FTE	FTE	FTE

La gerarchia è solitamente strutturata su tre macro livelli:

- Direzionale
- Manageriale o tattico
- Operativo

**Processo:** insieme delle attività tra loro interrelate, finalizzate alla realizzazione di un risultato definito e misurabile che contribuisce al raggiungimento della missione aziendale.

## Il sistema informativo

I processi necessitano conoscenza riguardo le risorse con le quali interagiscono.

**Information Processing Capacity:** adeguatezza di un'organizzazione rispetto alle *necessità di elaborare*.

**Sistema informativo:** insieme delle informazioni gestite dai processi aziendali e delle modalità di gestione.

**Sistema informatico:** porzione del sistema informativo che utilizza tecnologie informatiche.

Esempi di sistemi informatici innovativi in azienda: Robotic Process Automation (RPA)

[https://en.wikipedia.org/wiki/Robotic\\_process\\_automation](https://en.wikipedia.org/wiki/Robotic_process_automation)

## Ruolo dei sistemi informatici

L'informatica, nata come strumento a supporto dei processi aziendali, oggi cambia ruolo e diventa elemento di innovazione e trasformazione dei processi.

### Efficacia ed efficienza

Efficienza = Output effettivo / Input

Efficacia = Output effettivo / Output atteso

L'innovazione tecnologica incide positivamente sull'efficienza e indirettamente sull'efficacia.

## Evoluzione dei sistemi informatici

Fase	Descrizione	Esempi	Caratteristiche	Benefici
I	Automazione attività di raccolta, archiviazione, reperimento dati di natura operativa	Stipendi, gestione ordini, fatturazione, ...	<ul style="list-style-type: none"><li>Ripetitività delle operazioni da controllare</li><li>Natura struttura di processi da gestire</li><li>Grandi quantità di dati</li></ul>	<ul style="list-style-type: none"><li>Riduzione tempi e costi di elaborazione</li><li>Riduzione probabilità di errore</li></ul>
II	Automazione delle attività di controllo e valutazione dell'andamento aziendale e delle scelte direzionali	Controllo della produzione, analisi what-if, gestione del budget	<ul style="list-style-type: none"><li>Il sistema assume un ruolo non solo passivo ma anche di controllo attivo supportando i dirigenti e i quadri intermedi</li></ul>	<ul style="list-style-type: none"><li>Riduzione costi di coordinamento e controllo</li><li>Possibilità di analisi manualmente impossibili</li></ul>
III	Integrazione tra applicazioni settoriali, informatizzazione delle comunicazioni		<ul style="list-style-type: none"><li>Sistemi integrati orientati ai dati</li><li>Assenza ridondanze e incoerenze</li><li>Cooperazione e lavoro a distanza</li></ul>	<ul style="list-style-type: none"><li>Aumento qualità dei dati</li><li>Globalizzazione dell'azienda</li></ul>

## Sistemi informativi vs Basi di dati

Sistemi informativi e basi di dati sono concetti differenti.

Sistema informativo riguarda processi aziendali, informazioni necessarie ai processi, struttura aziendale.

Progettare un sistema informativo richiede competenze in ambito di organizzazione aziendale, economia, psicologia.

Basi di dati è parte del sistema informativo.

### Tre livelli:

- Presentazione: rapida evoluzione
- Applicazione
- Dati: bassa evoluzione. Tuttavia sta cambiando molto in questo ambito (integrazione di database relazionali con document based...)

### Obiettivi del corso

- Studiare le tipologie e composizione dei sistemi informativi
- Pianificare i sistemi informativi
  - Analizzare e capire i processi aziendali
  - Riprogettare i processi aziendali e i sistemi informatici
  - Analizzare i rischi e costi/benefici della riprogettazione e degli investimenti in tecnologia
  - Scrivere o rispondere a un capitolo tecnico
- Condurre un progetto informativo nel settore SI
  - Project management

*“Le figure professionali caratterizzate da un mix di competenze tecniche e manageriali/organizzative sono quelle che meglio riusciranno a resistere all'obsolescenza e alla concorrenza del personale proveniente dai Paesi emergenti”.*

Thomas W. Malone,(MIT- School of Management, 2005)

### Figure professionali

**Responsabili di sistemi informativi:** dirigenti preposti alla direzione e pianificazione di sistemi informativi in aziende di medio/grande dimensione.

- Visione strategica e di alto livello del Sistema Informativo
- Capacità di comprendere le problematiche di ordine organizzativo ed economico
- Capacità di pianificazione e progettazione della soluzione tecnologica
- Capacità di intermediazione tra i committenti (non informatici) e i progettisti/implementatori
- Conoscenza delle soluzioni presenti sul mercato
- Capacità di effettuare valutazioni tecnico-economiche sulle diverse soluzioni

**Consulenti/progettisti** in software house di grandi dimensioni sovrintendono a progetti di informatizzazione delle aziende, partecipano alle decisioni strategiche e dirigono il team dei progettisti e degli implementatori

- Capacità di condurre un progetto
- Capacità di rapportarsi con il cliente
- Analizzare i rischi e costi/benefici della riprogettazione e degli investimenti in tecnologia

### Data architect

Da Wikipedia: A data architect is a person responsible for ensuring that the data assets of an organization are supported by a data architecture that aids the organization in achieving its strategic goals. The data architecture should cover databases, data integration and the means to get to the data. Usually the data architect achieves his/her goals via setting enterprise data standards. A Data Architect can also be referred to as a Data Modeler, although the role involves much more than just creating data models.

### Data Scientist

Da IBM: A data scientist represents an evolution from the business or data analyst role. The formal training is similar, with a solid foundation typically in computer science and applications, modelling, statistics, analytics and math. What sets the data scientist apart is strong business acumen, coupled with the ability to communicate findings to both business and IT leaders in a way that can influence how an organization approaches a business challenge. Good data

scientists will not just address business problems, they will pick the right problems that have the most value to the organization.

## Seminari

- 8 aprile 2019 **KERING**: processo di gestione delle MEV (modifiche evolutive)
- To be defined **Technogym**: change management e integration APTTUS Quotation Tool in Technogym
- To be defined **Engineering**: la gestione d un progetto informatico complesso con SAP: Fasi, Ruoli e Cricitità.

## Modalità di esame

Interrogazione orale:

- Parti teoriche
- Casi di studio svolti a lezione
- Interventi esterni quando supportati dalle dispense

Parte fondamentale della valutazione dello studente sarà legata alla:

- Padronanza del linguaggio tecnico e terminologia specifica
- Capacità di collegare gli argomenti trattati in una visione olistica della materia

Data concordata (con circa 10gg di anticipo) con il docente.

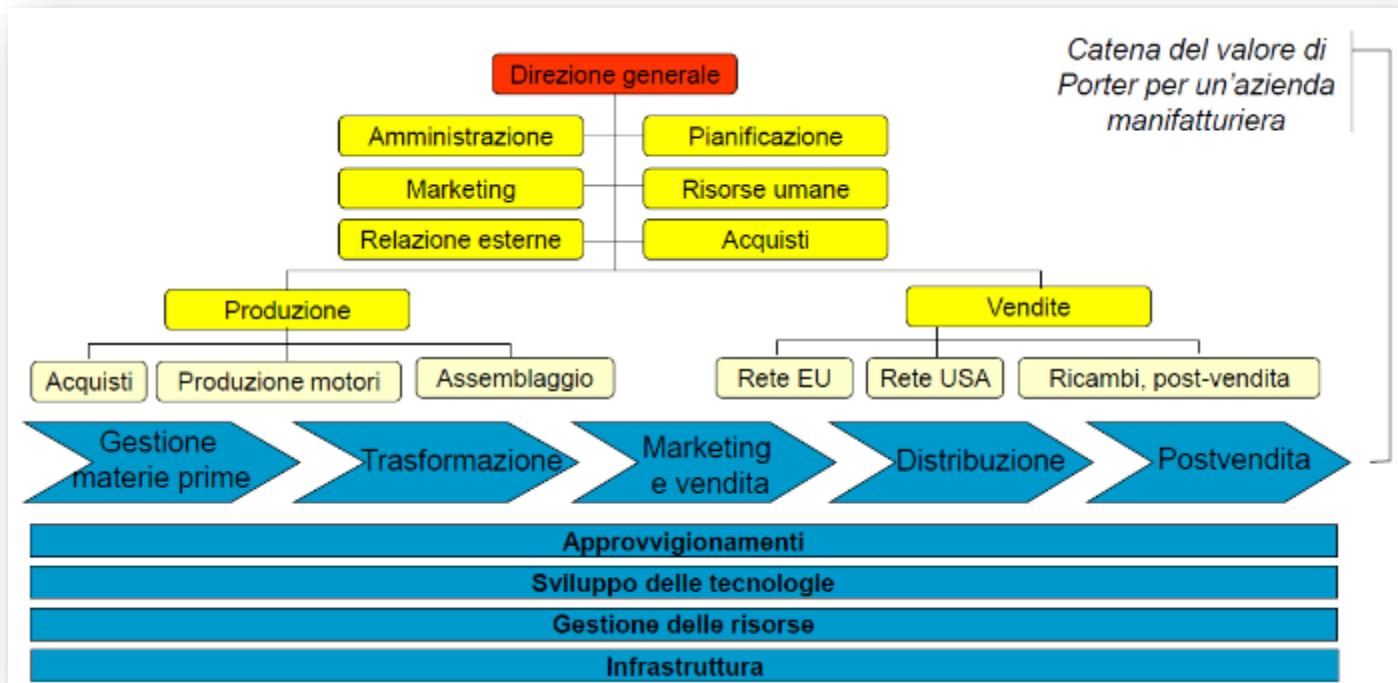
## Materiale didattico

<http://bias.csr.unibo.it/gofarelli/SISPEC/>

## Classificazione dei Sistemi Informativi

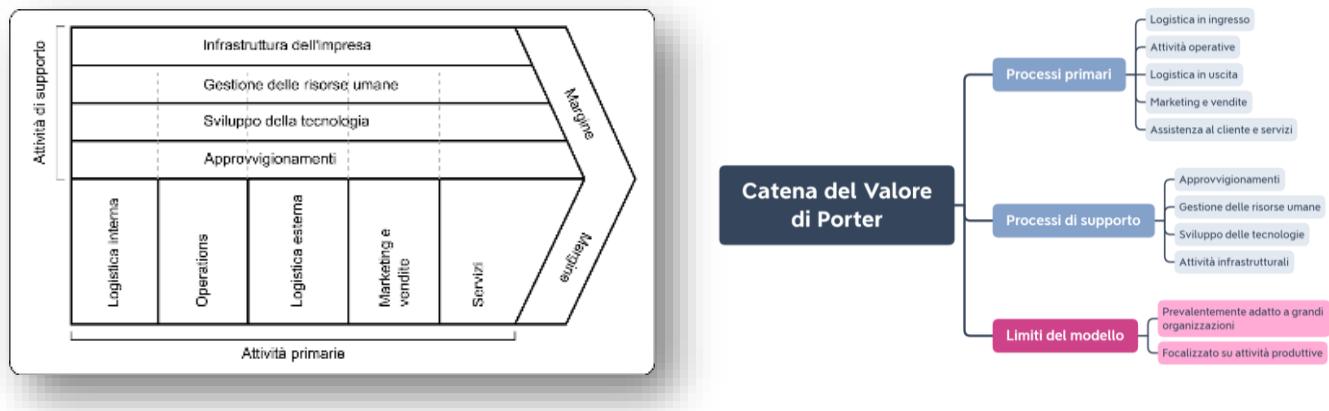
Un'azienda è una struttura che governa processi operativi finalizzati al raggiungimento di uno o più obiettivi.

I sistemi informativi supportano sia i processi di governo che i processi operativi.



## Catena del valore di Porter

Da Wikipedia: La catena del valore è un modello che permette di descrivere la struttura di una organizzazione come un insieme limitato di processi. Questo modello è stato teorizzato da Michael Porter nel 1985 nel suo best seller *Competitive Advantage: Creating and Sustaining Superior Performance*. Secondo questo modello, un'organizzazione è vista come un insieme di 9 processi, di cui 5 primari e 4 di supporto.



### I processi primari

I processi primari sono quelli che direttamente contribuiscono alla creazione dell'output (prodotti e servizi) di un'organizzazione e sono:

- **Logistica in ingresso:** comprende tutte quelle attività di gestione dei flussi di beni materiali verso l'interno dell'organizzazione: flussi che alimentano le attività operative
- **Attività operative:** attività di produzione di beni e/o servizi.
- **Logistica in uscita:** comprende quelle attività di gestione dei flussi di beni materiali verso l'esterno dell'organizzazione: flussi che portano sul mercato i risultati delle attività operative
- **Marketing e vendite:** attività di promozione del prodotto o servizio nei mercati e gestione del processo di vendita.
- **Assistenza al cliente e servizi:** tutte quelle attività post-vendita che sono di supporto al cliente (ad es. l'assistenza tecnica).

### I processi di supporto e approvvigionamento

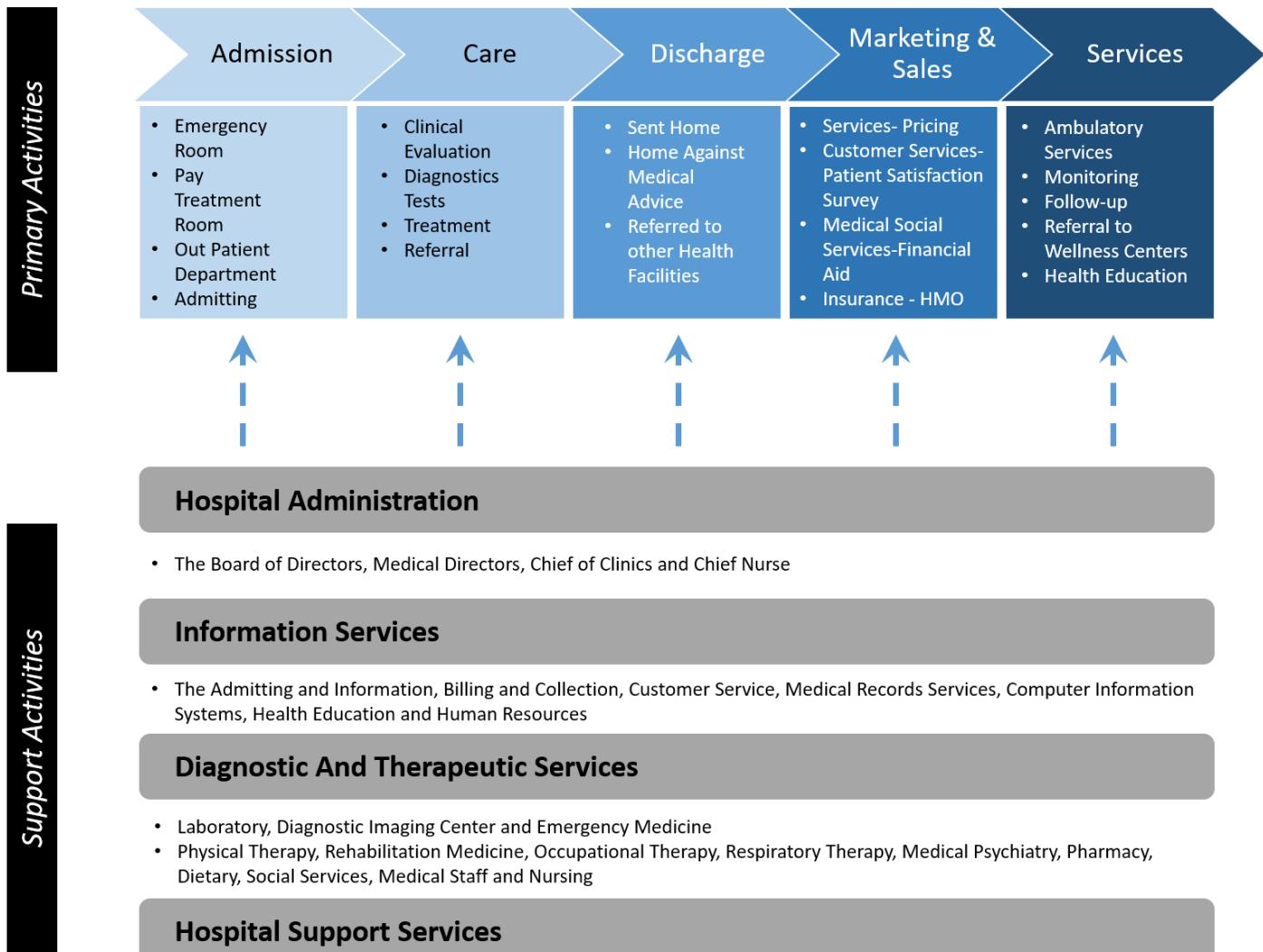
I processi di supporto sono quelli che non contribuiscono direttamente alla creazione dell'output ma che sono necessari perché quest'ultimo sia prodotto e sono:

- **Approvvigionamenti:** l'insieme di tutte quelle attività preposte all'acquisto delle risorse necessarie alla produzione dell'output ed al funzionamento dell'organizzazione.
- **Gestione delle risorse umane:** ricerca, selezione, assunzione, addestramento, formazione, aggiornamento, sviluppo, mobilità, retribuzione, sistemi premianti, negoziazione sindacale e contrattuale, ecc.
- **Sviluppo delle tecnologie:** tutte quelle attività finalizzate al miglioramento del prodotto e dei processi. Queste attività vengono in genere identificate con il processo R&D (Research and Development).
- **Attività infrastrutturali:** tutte le altre attività quali pianificazione, contabilità finanziaria, organizzazione, informatica, affari legali, direzione generale, ecc.

### Limiti del modello

Il modello originale di Porter si adatta prevalentemente a grandi organizzazioni che trattano la produzione di beni. Per le Organizzazioni diverse da quella di produzione di beni è tuttavia possibile utilizzare il modello come un valido spunto per l'analisi dei processi. In tal caso occorre provvedere ad un adattamento del modello stesso all'organizzazione oggetto di studio.

Ad esempio, la rappresentazione seguente, è un adattamento del modello di Porter alla sanità (in particolare la livello ospedaliero della sanità). I processi primari sono attività legate alla cura dei pazienti. I processi di supporto riguardano le risorse umane, tecnologiche, i beni e i servizi.



Fonte: <https://medium.com/@i3consult.com/business-and-digital-transformations-for-a-hospital-to-improve-patient-engagement-a-case-study-f037eeab9fe5>

## Sistemi informativi operativi

I sistemi informativi operativi si occupano di digitalizzare/informatizzare i processi volti all'esecuzione o alla pianificazione di attività.

- Programmazione della fabbricazione e Fabbricazione
- Programmazione della distribuzione e Distribuzione
- Amministrazione
- Gestione del personale.

L'importanza dei sistemi informativi operativi in azienda dipende dall'**intensità informativa** del settore.

Il CIO (Chief Information Officer) ha, tra gli altri, l'onere di valutare l'intensità informativa dei sistemi da implementare.

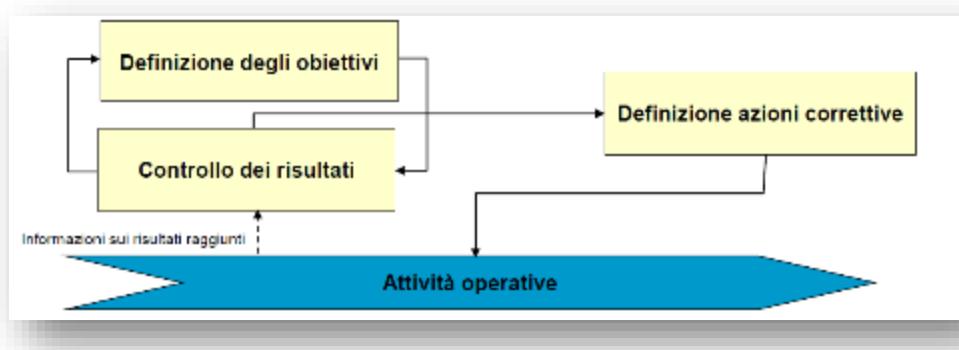
I sistemi informativi sono una tecnologia di produzione nei settori che *vendono* informazioni, sono una tecnologia di processo quando il prodotto è materiale.



## Sistemi informativi direzionali

I sistemi informativi direzionali supportano il processo decisionale, fornendo informazioni e conoscenza al management.

Ciclo di controllo basato sul modello:



Si differenziano dai sistemi informativi operativi in quanto:

- si basano su indici, informazioni aggregate o riassuntive
- utilizzano informazioni elaborate periodicamente, non accedono al flusso continuo di dati
- sono costruiti sopra i sistemi informativi operativi da cui estraggono informazioni.

## Sistemi informativi analitici

I sistemi informativi analitici supportano la comprensione dei fenomeni di business, sono focalizzati su prodotti, clienti e processi, permettendo la profilazione dei clienti (studio delle abitudini e dei comportamenti), mantenendo la storia del prodotto monitorandone l'affidabilità.

Mentre i sistemi informativi direzionali studiano fenomeni *interni* all'azienda, quelli analitici misurano fenomeni *esterni*.

### Sistemi analitici per settori

Settore	Numerosità clienti	Finalità di analisi
Telefonia	Oltre 10 milioni	Profittabilità Comportamento e preferenze
Grandi banche	Oltre 1 milione	Profittabilità
Aziende elettriche e gas	Da 100 mila a 1 milione	Profittabilità Comportamento e preferenze
Pubblica amministrazione	Oltre 10 milioni	Studi di settore Segmentazione dei contribuenti Individuazione di potenziali evasori
Grande distribuzione	Da 100 mila a 1 milione	Comportamento e preferenze

## Mappa dei Sistemi Informativi

La realizzazione di una mappa dei Sistemi Informativi aziendali ne prevede la scomposizione in moduli a fini conoscitivi e di classificazione.

Un **modulo** rappresenta un blocco di funzionalità software che supporta una fase di un processo aziendale, omogeneo per frequenza, per attori e profili di casi d'uso. Un modulo descrive un'esigenza informativa e indica il potenziale impiego di una tecnologia informatica.

La suddivisione in moduli è utilizzata dai grandi produttori di package (SAP, PeopleSoft, ORACLE, ...) per strutturare le proprie offerte.

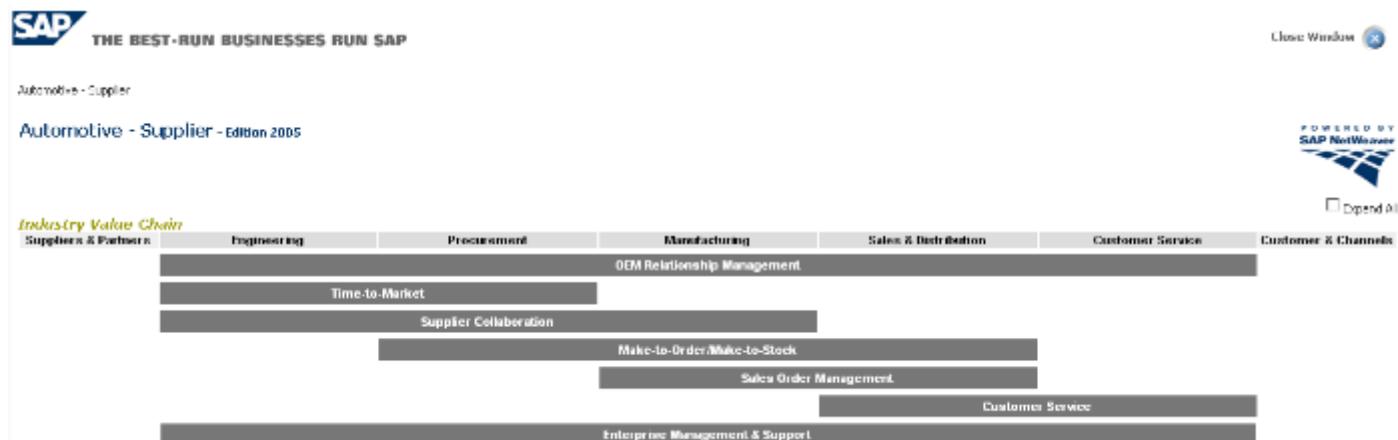
I moduli possono essere:

- Orizzontali: non variano al variare della tipologia aziendale
- Verticali: specifici per settore produttivo.

### Un esempio da SAP

L'intestazione (in grigio, chiaro) rappresenta la “Catena del valore di Porter” per un automotive supplier.

In grigio scuro, sotto, la mappa i moduli applicativi SAP corrispondenti ai diversi processi delle catena del valore.



## Il portafoglio applicativo aziendale

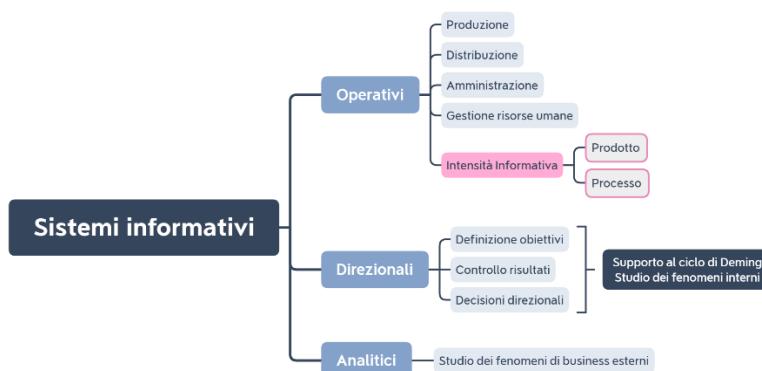
Il portafoglio applicativo in un'azienda rappresenta una possibile mappa dei Sistemi Informativi aziendali. Può essere considerato come una descrizione dettagliata della catena del valore di Porter ottenuta incrociando le *fasi del ciclo di trasformazione* con la *tipologia delle attività* applicabili per la pianificazione ed esecuzione di ciascuna fase del ciclo. Ogni incrocio individua un modulo.

La **pianificazione** ha lo scopo di determinare i piani e controllarne l'avanzamento; le attività di pianificazione si distinguono per l'orizzonte temporale e il livello di dettaglio.

Le **attività di esecuzione** attuano le operazioni pianificate producendo l'output desiderato. I sistemi di supporto all'esecuzione elaborano informazioni in tempo reale originando benefici riconducibili ai risparmi su attività burocratiche o alle migliori prestazioni di tempi e costi dei processi.

### Il portafoglio applicativo (un esempio)

		Progettazione e industrializzazione	Approvvigionamento	Fabbricazione	Distribuzione e vendita
<b>Analisi strategica e ambientale</b>	• Osservatorio tecnologico	• Marketing acquisti			• Ricerche di mercato • Marketing cliente e prodotto
<b>Pianificazione</b>	• Piano dei progetti	• Piano degli acquisti	• Piano della produzione		• Previsioni vendita e piano delle vendite
<b>Gestione dei dati tecnici</b>	• Archivio disegni • Distinta base di progettazione	• Anagrafe dei fornitori • Distinta base di produzione	• Anagrafe impianti • Cicli di lavorazione		• Anagrafe dei clienti • Catalogo dei prodotti
<b>Progettazione operativa</b>	• Pianificazione dei progetti	• Programmazione forniture	• Programmazione stabilimenti		• Programmazione trasporti e consegne
<b>Schedulazione</b>	• Schedulazione dei progetti e dei reparti	• Schedulazione delle consegne e solleciti	• Schedulazione dei reparti		• Schedulazione dei trasporti
<b>Flusso degli ordini</b>	• Schede di lavoro	• Gestione ordini ai fornitori	• Gestione ordini di produzione		• Gestione ordini dei clienti
<b>Flusso dei materiali e delle operazioni</b>	• Gestione dei laboratori	• Ricezione e collaudo • Magazzini materie prime • Conto lavorazione	• Movimentazione delle scorte in fabbrica • Controllo avanzato della produzione		• Spedizioni e trasporto • Magazzino prodotti finiti



## Profilo dei livelli di pianificazione

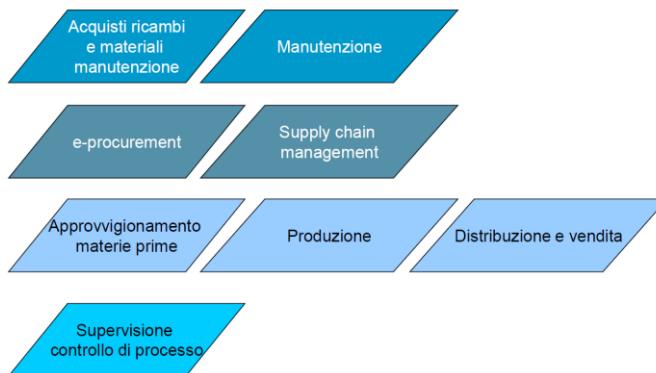
Obiettivo				Flusso informativo	Complessità e volumi
<b>Analisi strategica</b>	<ul style="list-style-type: none"> <li>Previsioni di mercato</li> <li>Monitoraggio dei trend di mercato e dell'innovazione tecnologica</li> </ul>		<ul style="list-style-type: none"> <li>Informazioni esterne</li> <li>Processo molto variabile, destrutturato, spesso influente</li> </ul>		<ul style="list-style-type: none"> <li>Applicazioni ad-hoc</li> <li>Volumi variabili e limitati</li> <li>Dati complessi</li> </ul>
<b>Pianificazione annuale</b>	<ul style="list-style-type: none"> <li>Dimensionamento dei volumi e delle attività nel medio termine</li> <li>Coordinamento delle operazioni</li> <li>Tempificazione dell'innovazione</li> </ul>		<ul style="list-style-type: none"> <li>Informazioni interne e strutturate (obiettivi aggregati di produzione, vendita, acquisto)</li> <li>Flusso informativo interfunzionale (riflette i processi, non i reparti)</li> </ul>		<ul style="list-style-type: none"> <li>Modelli di calcolo complessi in grandi aziende multilocalizzate e/o supply chain estese</li> <li>Volumi generalmente contenuti</li> <li>Dati sintetici</li> </ul>
<b>Pianificazione operativa</b>	<ul style="list-style-type: none"> <li>Calcolo dei programmi di attività e controllo del loro avanzamento</li> <li>Coordinamento delle operazioni: <i>condivisione delle risorse</i> <i>sequenza ottimale delle attività</i> <i>sincronia delle operazioni</i></li> </ul>		<ul style="list-style-type: none"> <li>Informazioni esterne e strutturate</li> <li>Flusso informativo interfunzionale e/o interaziendale</li> <li>Granularità tipica: settimana-giorno, reparto-ordine</li> </ul>		<ul style="list-style-type: none"> <li>Modelli di ottimizzazione</li> <li>Sistemi di supporto ai progetti</li> <li>Dati analitici</li> </ul>

## Profilo delle attività di esecuzione

Obiettivo				Flusso informativo	Complessità e volumi
<b>Flusso degli ordini</b>	<ul style="list-style-type: none"> <li>Elaborazioni transazioni</li> <li>Automazione della manualità e della guida operativa</li> </ul>	<ul style="list-style-type: none"> <li>Flusso informativo interfunzionale e/o interaziendale</li> <li>Input a pianificazione operativa</li> </ul>			<ul style="list-style-type: none"> <li>Grandi volumi di transazioni</li> <li>Basi di dati grandi e complesse</li> <li>Criticità riguardo il tracciamento dell'ordine</li> </ul>
<b>Flusso dei materiali</b>	<ul style="list-style-type: none"> <li>Registrazione eventi</li> <li>Guida alla movimentazione delle scorte</li> </ul>	<ul style="list-style-type: none"> <li>Flusso stratificato su più livelli: interaziendale, interfunzionale, locale</li> <li>Feedback a pianificazione operativa</li> <li>Dati interni (scorte proprie) ed esterni (scorte fornitori e scorte clienti)</li> </ul>			<ul style="list-style-type: none"> <li>Medi volumi di transizioni</li> <li>Basi di dati grandi</li> <li>Criticità riguardo il tracciamento del materiale</li> </ul>
<b>Flusso delle operazioni</b>	<ul style="list-style-type: none"> <li>Registrazione eventi</li> <li>Guida all'esecuzione delle operazioni</li> </ul>	<ul style="list-style-type: none"> <li>Flusso stratificato su più livelli: interaziendale, interfunzionale, locale</li> <li>Feedback a pianificazione operativa</li> <li>Dati interni</li> </ul>			<ul style="list-style-type: none"> <li>Grandi volumi di transazioni</li> <li>Basi di dati grandi e complesse</li> <li>Criticità riguardo la raccolta dei dati in tempo reale</li> </ul>

Il portafoglio applicativo aziendale si differenzia in base alla tipologia dell'azienda.

Es. Aziende chimiche e siderurgiche



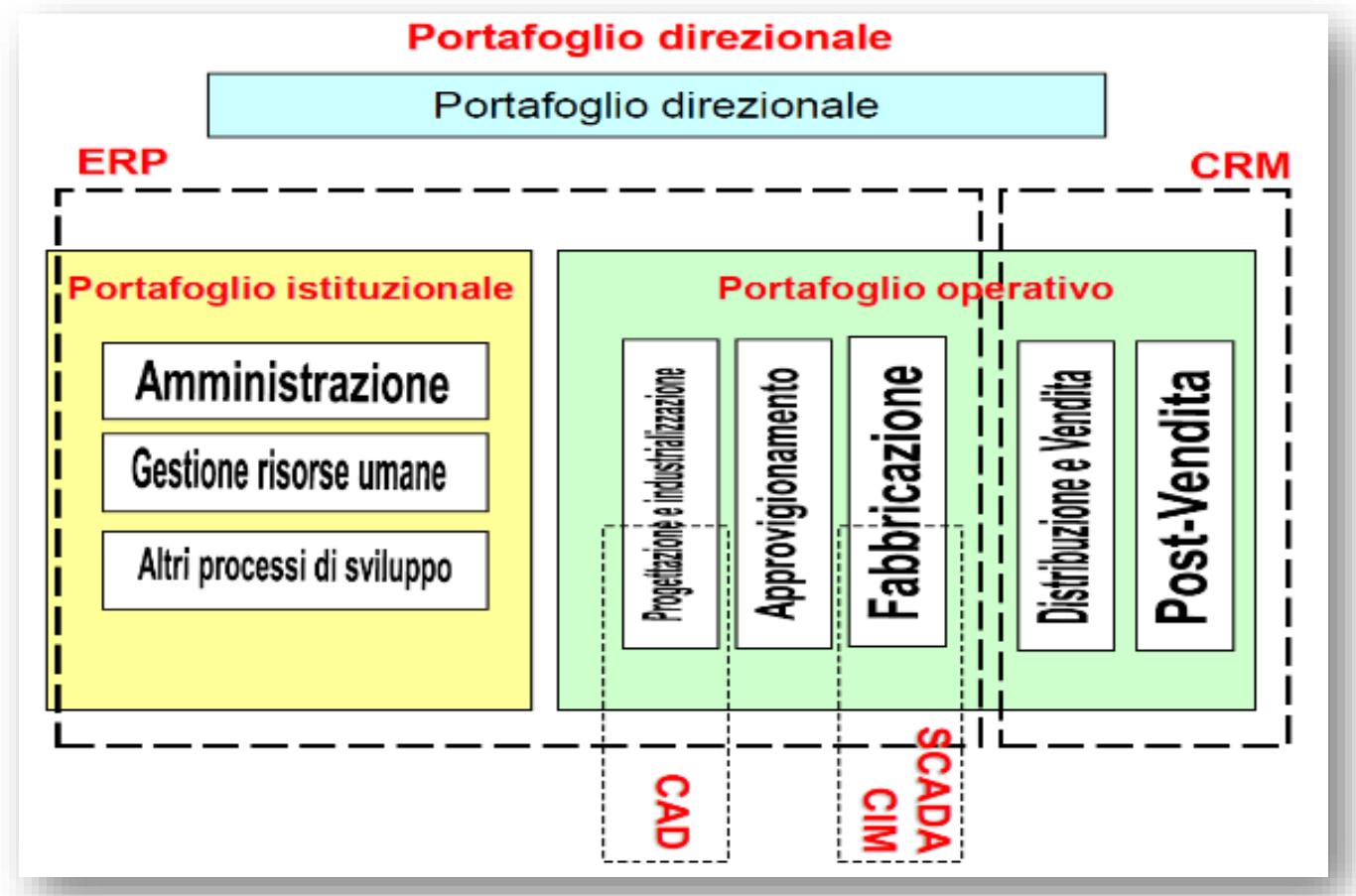
- Prodotto più semplice: meno enfasi sulla progettazione
- Poche materie prime
- Complesso il monitoraggio e manutenzioni impianti
- Notevole investimento sui ricambi degli impianti (e-procurement)
- Necessità di coordinamento nei complessi multi-stabilimento (supply chain management)

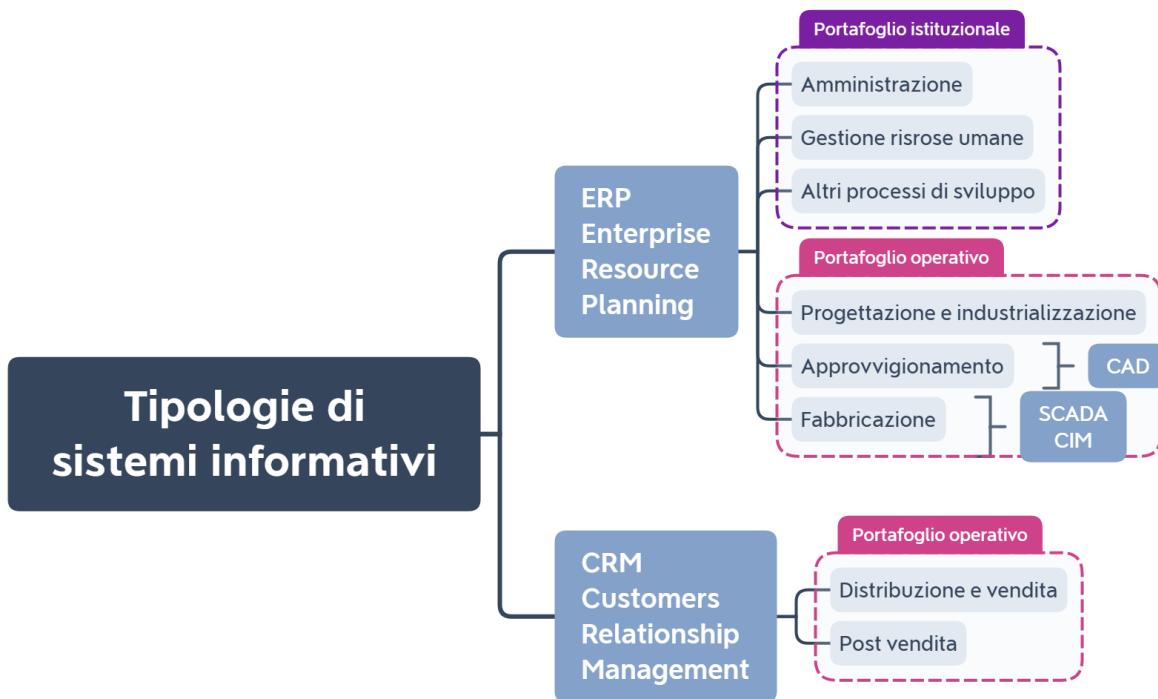
Es. Aziende telefoniche



- Importanza primaria del contatto con il cliente (CRM)
- Problematiche di gestione della rete complesse

Tipologie di sistemi informativi





### Sistemi CIM (Computer Integrated Manufacturing)

Si tratta di sistemi che hanno l'obiettivo di integrare automaticamente i vari settori/livelli di un sistema di produzione (dalla progettazione alla produzione) con finalità di ottimizzazione dei processi e della gestione delle risorse.

Vantaggi:

- Riduzione delle scorte (tramite controllo produzione e pianificazione)
- Riduzione del time to market
- Incremento della qualità del prodotto: analisi e controllo del processo produttivo
- Riduzione dei costi dovuta alla maggiore efficienza della fabbrica.

Le architetture CIM sono strutturate su cinque livelli collegati tra loro, ognuno dei quali esegue specifiche elaborazioni.

Esempi di contenuti applicativi		Esempi di tecnologie
<b>Macchina</b>	Esecuzione del processo fisico	PLC (Programmable Local Control) basati su processori
<b>Cella</b>	Successione dei flussi tra gruppi di macchine	Microprocessori governati da PC o elaborati ad-hoc
<b>Area</b>	Programmazione esecutiva di reparto Movimentazione fisica dei materiali	Reti locali con server di medie dimensioni e posti di lavoro interattivi
<b>Stabilimento</b>	Programmazione della produzione	Server di stabilimento
<b>Azienda</b>	Acquisti di materie prime Sistemi inter-stabilimento	Server con basi di dati aziendali

Il modello CIM prevede l'integrazione di diverse tecnologie interconnesse:

- CAD Computer Aided Design
- CAM Computer Aided Manufacturing
- CAT Computer Aided Test
- CAPP Computer-aided process planning

- MRP Materials Requirements Planning
- Robotica
- AGV Movimentazione automatica dei materiali
- ERP (Enterprise resource planning)
- CNC (computer numerical control) machine tools
- DNC, Direct Numerical Control machine tools
- FMS, flexible manufacturing system
- ASRS, automated storage and retrieval systems
- Automated conveyance systems
- computerized scheduling and production control
- CAQ (Computer-aided quality assurance)
- A business system integrated by a common database

#### *Sistemi SCADA (Supervisory Control and Data Acquisition)*

Il componente dei sistemi CIM preposto al controllo dei sistemi industriali, all'acquisizione e all'analisi dei dati da essi prodotti è denominato **SCADA**.

I sistemi SCADA sono sistemi informatici distribuiti per il monitoraggio e la supervisione di sistemi fisici. Si tratta di una tecnologia in costante evoluzione, di pari passo con i progressi dell'elettronica, dell'informatica e delle reti di telecomunicazioni, principalmente utilizzata in ambito industriale e infrastrutturale.

Il sistemi SCADA sono utilizzati per monitorare e controllare grandi impianti industriali e sistemi meccanici/elettrici distribuiti sul territorio. I principali ambiti applicativi sono:

- Grandi impianti industriali e processi industriali complessi
- Telecomunicazioni (es. ripetitori)
- Sistemi di gestione di acqua, fognature, distribuzione di energia (HERA)
- Sistemi di produzione energia (impianti fotovoltaici ed eolici)
- Raffinerie
- Trasporti (aeroporti, controllo del traffico, ferrovie)
- Centri di ricerca (CERN).

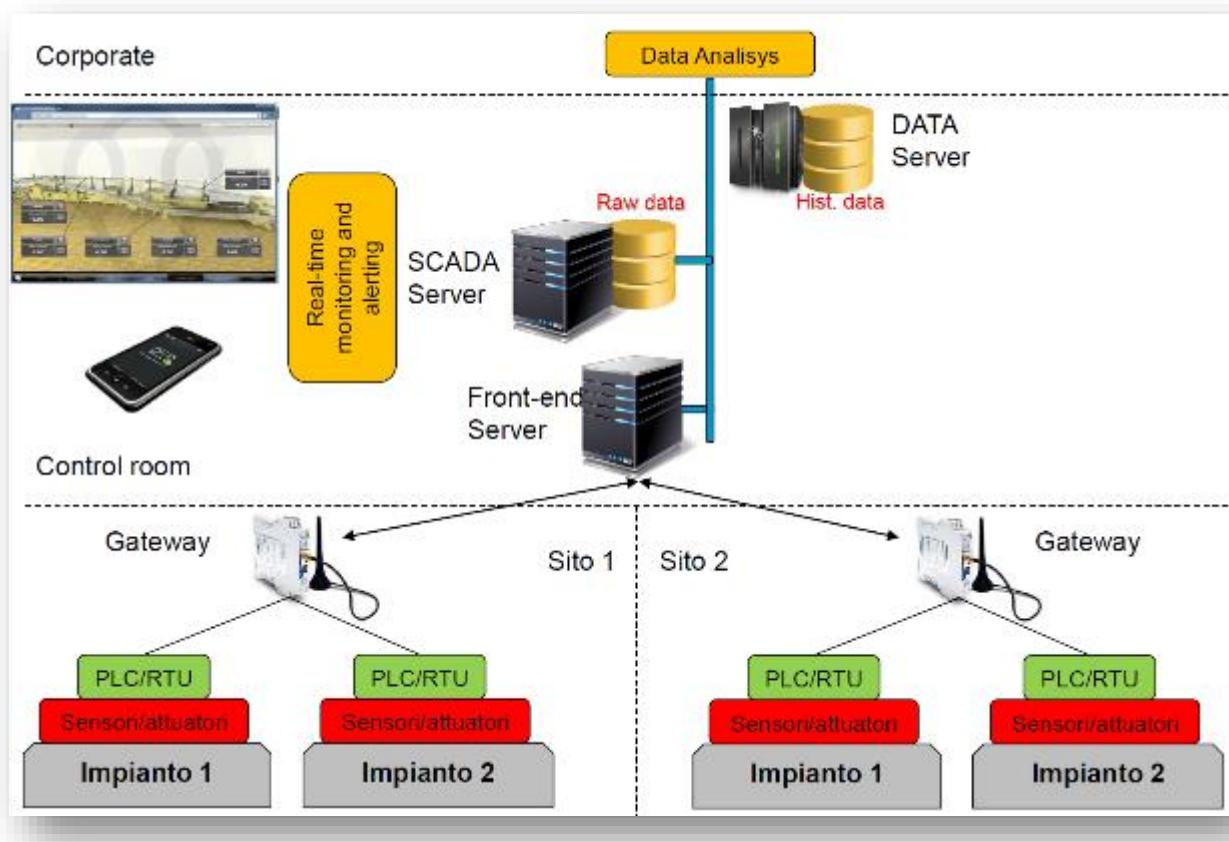
Grazie all'uso di un sistema SCADA è possibile risparmiare tempo e denaro:

- Meno spostamenti per i lavoratori
- Riduzione delle esigenze di personale
- Aumento della produttività maggiore reattività rispetto a situazioni anomale
- Riduzione di costi generali
- Maggiore affidabilità degli impianti.

#### Architettura dei sistemi SCADA

<b>Sensori</b>	Misurazione di pressione, velocità, luminosità, umidità, temperatura, livello, distanza
<b>Attuatori</b>	Valvole, pompe, motori, ...
<b>Controllori</b>	<p><b>PLC</b>: qualunque dispositivo programmabile in grado di risolvere una logica che implementa un algoritmo</p> <p><b>RTU (Remote Terminal Unit)</b>: non elabora né risolve logiche, è un terminale che esegue gli ordini ricevuti da un'entità di classe superiore; è un ripetitore con capacità di comunicazione</p>

<b>Gateway</b>	I gateway raccolgono informazioni dai PLC/RTU e si occupano di trasferirle in tempo reale al sistema centralizzato implementando opportuni protocolli di trasmissione, crittografia, sicurezza (modbus, DNP3).
<b>Front-end server</b>	Ricevono dati dai diversi siti e li traducono in un formato utilizzabile dal server SCADA. Sono logicamente separati dal server SCADA a garanzia di un tempo di acquisizione real-time.
<b>Scada server</b>	Implementano le funzionalità di monitoraggio e gestione degli allarmi; operano in tempo reale tipicamente utilizzando uno stream di dati: <ul style="list-style-type: none"> <li>• Utilizza un database contenente una serie limitata di dati di dettaglio</li> <li>• Può supportare il controllo automatico tramite interfacce grafiche disposte in una sala di controllo o può implementare regole di alerting automatiche</li> </ul>
<b>Data server</b>	Si occupa di salvare una versione storica dei dati (es. campionamento ogni 15 minuti); è la base dati utilizzata a fini di analisi statistiche, per il tuning dei sistemi e le performance, per attività di data mining.



## Funzionalità

Un sistema SCADA moderno può realizzare le seguenti funzioni (fonte wikipedia: <https://it.wikipedia.org/wiki/SCADA>):

- **Acquisizione dati** mediante opportuni driver di comunicazione verso gli apparati in campo (microcontrollori o PLC) a loro volta connessi ai sensori o agli attuatori. I dati scambiati sono normalmente grandezze digitali, analogiche oppure stringhe di testo. Tali dati vengono in certi casi integrati con informazioni aggiuntive gestite dai microcontrollori o dai PLC quali ad esempio la marca temporale oppure l'identificativo di qualità del dato (certo, incerto, misura guasta o fuori misura etc.). La comunicazione può avvenire sia con le classiche interrogazioni sincrone (polling) che asincrone, ovvero basate su notifica - iniziata da parte dell'apparato in

campo e relativa all'avvenuta variazione di una misura o di uno stato classificato come di interesse per il sistema SCADA.

- **Rappresentazione del dato** sullo schermo di un PC, all'interno di un web-browser o sullo schermo di uno smartphone. Il dato può essere rappresentato sia come grandezza che sotto forma di animazione grafica (esempio un serbatoio con il liquido colorato che sale e scende in base alla lettura del livello). Dall'interfaccia grafica è anche possibile inviare comandi al sistema di automazione.
- **Storicizzazione del dato** - I dati di interesse possono essere storicizzati su archivi locali o distribuiti, in varie metodologie: da file di tipo binario a database relazionali, a seconda del tipo di piattaforma impiegata e delle esigenze del progetto. Tali dati poi possono essere visualizzati dall'operatore direttamente dalla piattaforma anche sotto forma di grafici oppure esportati e gestiti su sistemi terzi che ne consentano l'analisi ai fini dell'efficientamento del processo o dell'analisi di situazioni critiche che si sono verificate.
- **Gestione degli allarmi** - L'allarme è una particolare condizione del processo che viene modellizzata dal progettista della piattaforma SCADA e che richiede l'interazione da parte di un operatore. In caso ad esempio di un blocco di una pompa, si potranno avere delle icone lampeggianti, l'emissione di un suono, oppure nei sistemi più complessi l'invio di un'e-mail, SMS, oppure una chiamata telefonica in sintesi vocale all'operatore reperibile - il tutto al fine di rendere edotto l'operatore che è necessaria un'azione umana per risolvere un'eccezione sul sistema non gestibile in autonomia.
- **Interazione con sistemi di livello superiore** - Funzione che è sempre più richiesta e che prevede che lo SCADA possa funzionare come "Gateway" fra il processo e i sistemi di livello superiore (da un controller di dominio per l'autenticazione degli utenti a un sistema MES o ERP ad esempio affinché le scorte di magazzino di prodotto finito e materie prime siano aggiornate automaticamente col procedere della produzione).

Da un punto di vista terminologico si possono distinguere:

- **Sistemi HMI** (Human Machine Interface) quando la fruizione del sistema SCADA avviene da un unico punto (un PC oppure un pannello tattile). Normalmente impiegato in sistemi di automazione locale di processi semplici o comunque confinati in ambienti ristretti. Talvolta i sistemi HMI sono fruibili anche da pannello operatore, ovvero dispositivi hardware e software dedicati che non hanno come base un sistema operativo PC quale Windows, Windows CE o Linux, ma un sistema embedded dedicato esclusivamente a un determinato microcontrollore.
- **Sistemi HMI-SCADA** (Human Machine Interface / Supervisory Control And Data Acquisition) distribuiti che prevedono più punti di acquisizione dati (server) e più punti di utilizzo (ad esempio in un'infrastruttura autostradale si può impiegare un server SCADA dedicato all'acquisizione dati per ogni galleria o casello, e la fruizione del sistema nel suo complesso può avvenire sia dalla centrale di controllo del gestore autostradale che dai terminali mobili degli operatori di manutenzione).

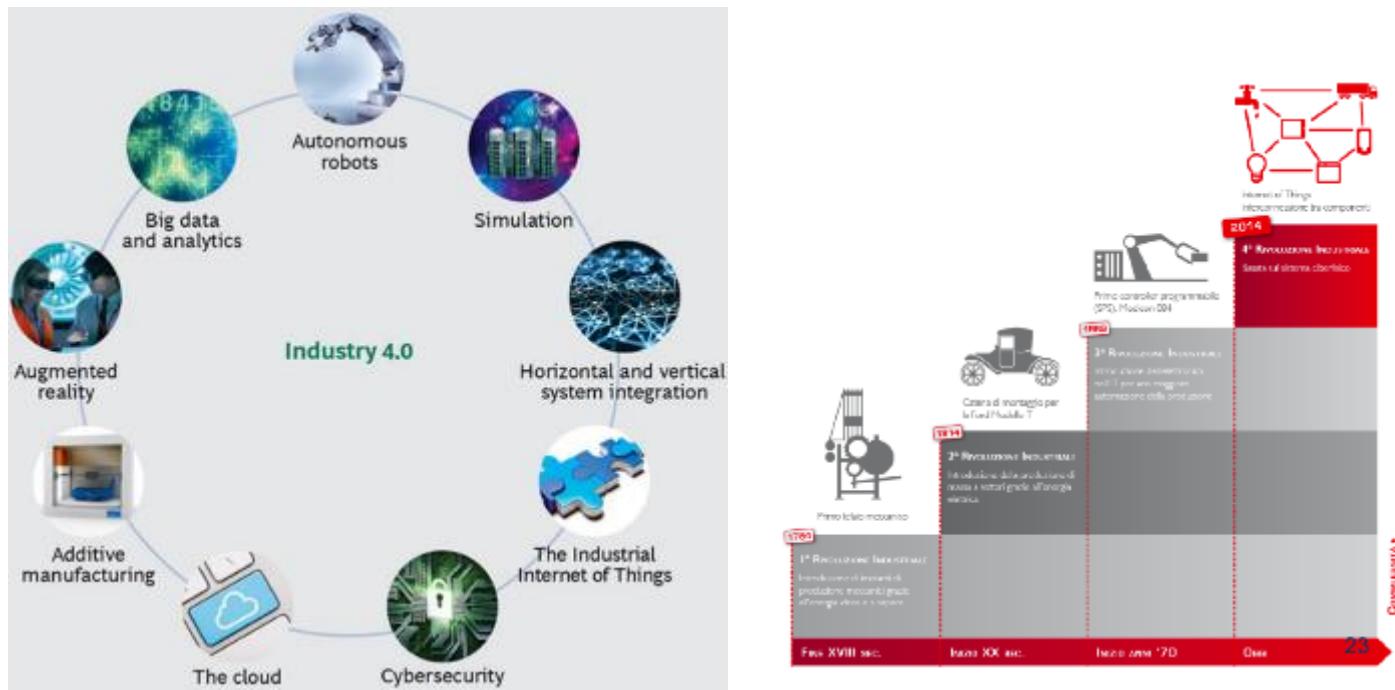
I sistemi SCADA si sono evoluti in funzione di diverse dimensioni o ambiti applicativi:

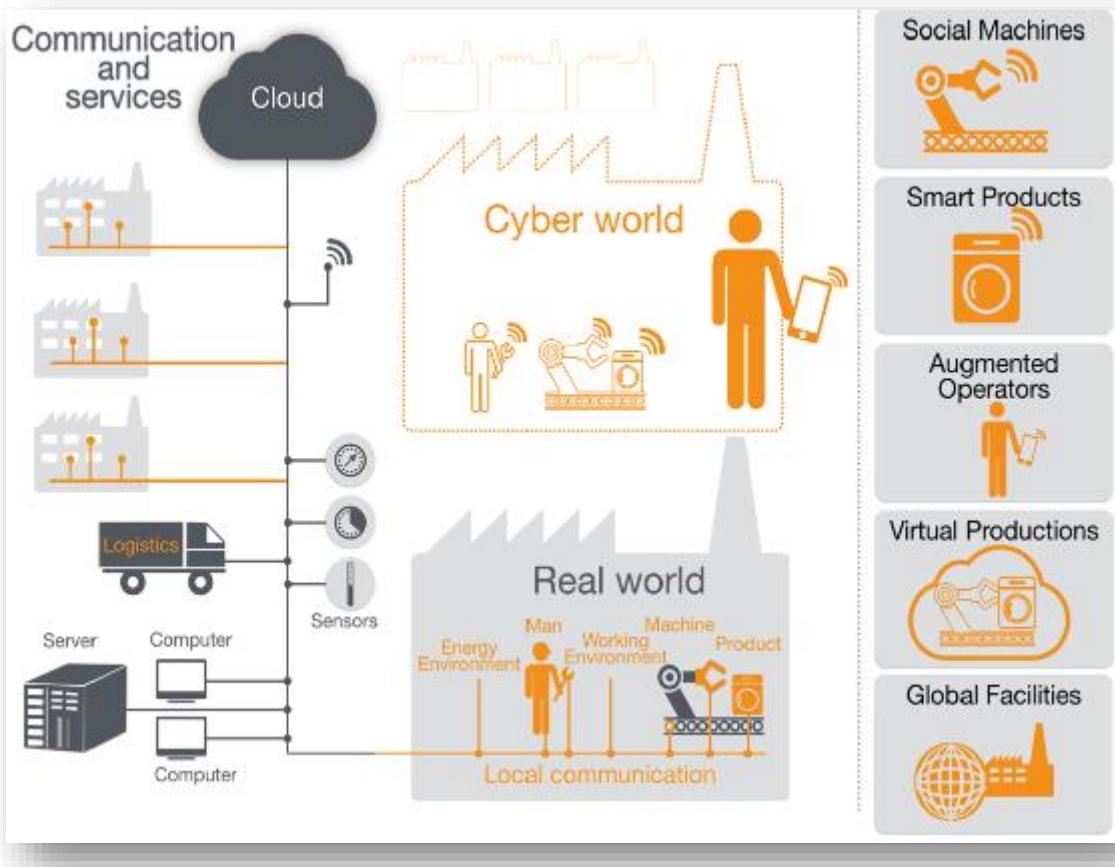
- **Scalabilità**: la crescita del numero di impianti monitorati e di segnali raccolti pone il problema dell'enorme mole di dati da gestire (in tempo reale), memorizzare ed analizzare
- **Capacità di analisi dei dati raccolti**: si aggiungono moduli per la valorizzazione dei dati storici raccolti (data mining, analisi delle performance)
- **Sicurezza**: i sistemi SCADA, pur controllando infrastrutture critiche, sono basati su tecnologie di 5/10 anni in ritardo rispetto allo stato dell'arte dell'IT, rilevando possibili debolezze rispetto ad attacchi (esterni ed interni).

*Nel novembre 2008 è apparso un virus informatico chiamato Stuxnet che prendeva come bersaglio i sistemi SCADA. Il virus ha usato come mezzo trasmissivo una chiave USB e aveva il compito di contaminare il software WinCC sviluppato dall'azienda Siemens. All'inizio del 2011 Stuxnet è uscita sotto una nuova forma chiamata Stars.*

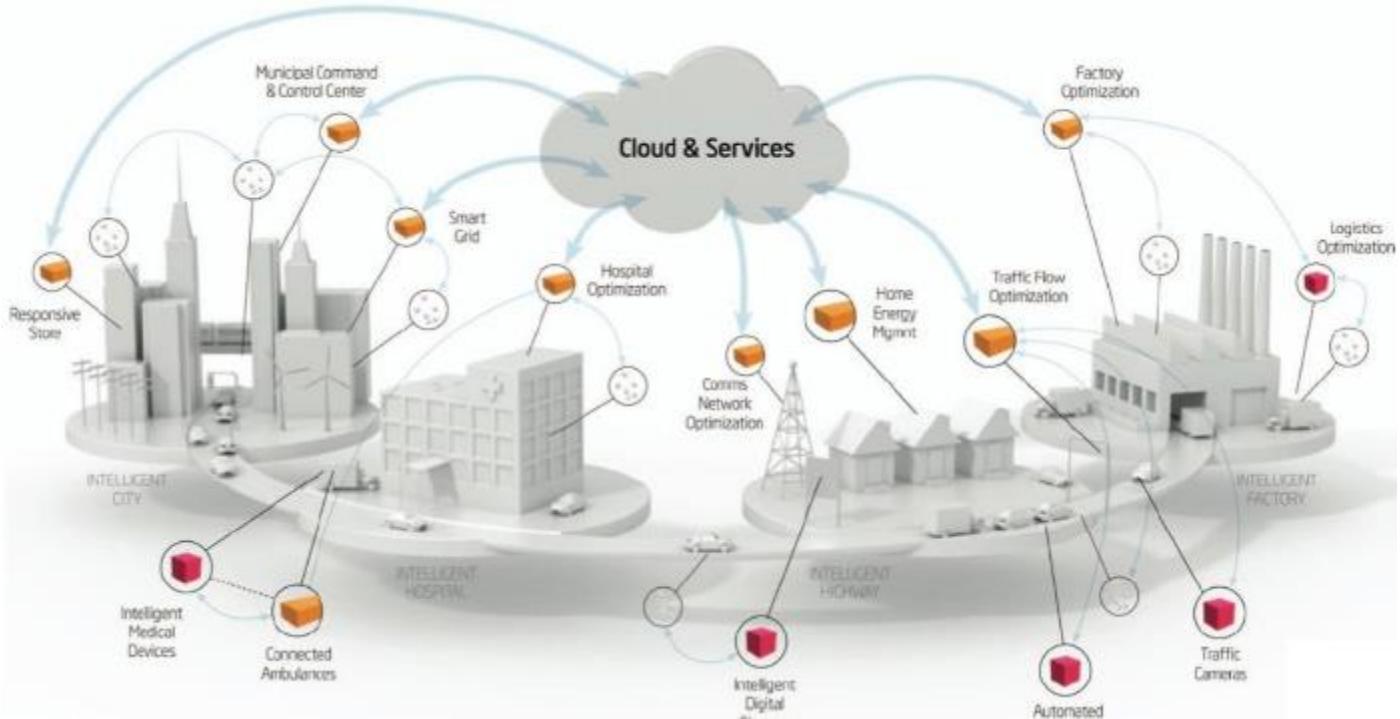
I sistemi SCADA sono sempre di più nel mirino dei criminali informatici, poiché fanno largo uso di protocolli di comunicazione basati su TCP-IP che spesso non hanno alcuna forma di autenticazione, password o crittografia. Il motivo alla base di ciò è che storicamente venivano impiegati come rimpiazzo di reti chiuse punto-punto dove non erano originariamente previsti gateway di interconnessione. È pertanto importante segmentare le reti di processo affinché siano irraggiungibili dall'esterno, ed utilizzare sistemi SCADA moderni e costantemente aggiornati come unico punto d'accesso verso la rete industriale, impedendo che si possano realizzare connessioni dirette agli apparati in campo che bypassano i meccanismi di protezione ed autenticazione presenti sulle piattaforme SCADA.

Il termine **Industria 4.0** indica una tendenza dell'automazione industriale che integra nuove tecnologie produttive per migliorare le condizioni di lavoro e aumentare la produttività e la qualità produttiva degli impianti.





Il termine **Internet of Things** riguarda le connessioni digitali tra oggetti che si rendono riconoscibili e acquisiscono intelligenza grazie al fatto di poter comunicare dati riguardo se stessi e accedere ad informazioni aggregate da parte di altri. Le applicazioni IoT spaziano dalla domotica alla medicina dal settore energy a quello del divertimento.



### Sistemi Enterprise Resource Planning (ERP)

Il termine **Enterprise Resource Planning** è stato coniato agli inizi degli anni '90 da Gartner Group. Un ERP è una suite di moduli applicativi che supportano l'intera gamma dei processi aziendali, può includere moduli orizzontali e verticali.

In linea di massima, le aree funzionali aziendali gestite dai moduli *core* di un ERP sono:

- **Logistica**
  - Material Requirements Planning (MRP)
    - Gestione dei dati prodotti
    - Pianificazione e controllo della produzione
  - Gestione dei materiali
  - Controllo della qualità
  - Controllo degli impianti
  - Vendita e distribuzione
- **Gestione delle risorse umane**
  - Gestione del personale
  - Gestione dell'organigramma
  - Gestione del tempo
  - Gestione delle paghe
  - Sviluppo del personale
- **Gestione finanziaria**
  - Gestione dei capitali
  - Gestione degli investimenti
  - Controllo dei flussi economici

A questi moduli *core* si possono integrare ulteriori moduli aggiuntivi:

- **Product Lifecycle Management (PLM)**

*Supporto alla gestione della documentazione tecnica di prodotto e dei relativi processi produttivi*

- **Supply Chain Management (SCM)**

*Supporto alla pianificazione e al controllo delle attività interaziendali*

- **Customer Relationship Management (CRM)**

*Supporto all'interazione con il cliente*

- **E-procurement**

*Supporto alla realizzazione di mercati elettronici interaziendali in cui le condizioni quadro e i fornitori sono definite dagli uffici acquisti, mentre sono i reparti a seguire gli acquisti veri e propri abbattendo costi e tempi*

### *Paradigmi fondamentali per i sistemi ERP*

#### *Unicità dell'informazione*

Tutte le elaborazioni utilizzano le stesse informazioni. Il valore di ogni informazione è unico all'interno del sistema.

Vantaggi:

- La sincronizzazione dati abilita la sincronizzazione dei processi
- L'assenza di ridondanza semplifica le procedure di aggiornamento
- Tracciabilità degli aggiornamenti
- Affidabilità e consistenza delle informazioni aziendali per analisi e garanzia di conoscenza dell'azienda.

In assenza di unicità possono verificarsi sgradevoli inconvenienti, ad esempio:

- Se l'informazione di affidabilità dei pagamenti di un cliente è registrata nel sistema/modulo contabile e in quello di commerciale, in caso di non unicità dell'informazione il commerciale può continuare ad accettare ordini dal cliente insolvente
- Se le giacenze di magazzino sono presenti, non allineate, nei sistemi di magazzino e di programmazione degli acquisti, il processo di approvvigionamento o quello di produzione vengono alterati
- Se le informazioni sono distribuite e duplicate, il processo di consolidamento può richiedere molto tempo e diverse problematiche per la gestione di conflitti.

#### *Estensione e modularità funzionale*

La modularità dei sistemi ERP permette ad ogni azienda di adottare i soli moduli necessari in determinati momenti storici, secondo diverse strategie di acquisizione:

- **Incrementale:** si acquistano progressivamente i moduli necessari, sostituendo via via sistemi legacy inadeguati
- **One stop shopping:** si acquistano tutti i moduli necessari in un'unica istanza da uno stesso vendor
- **Best of breed:** si utilizzano moduli di diversi vendor in funzione delle valutazioni di qualità da parte dell'azienda (con garanzia da parte dei fornitori di integrazione)

#### *Prescrittività*

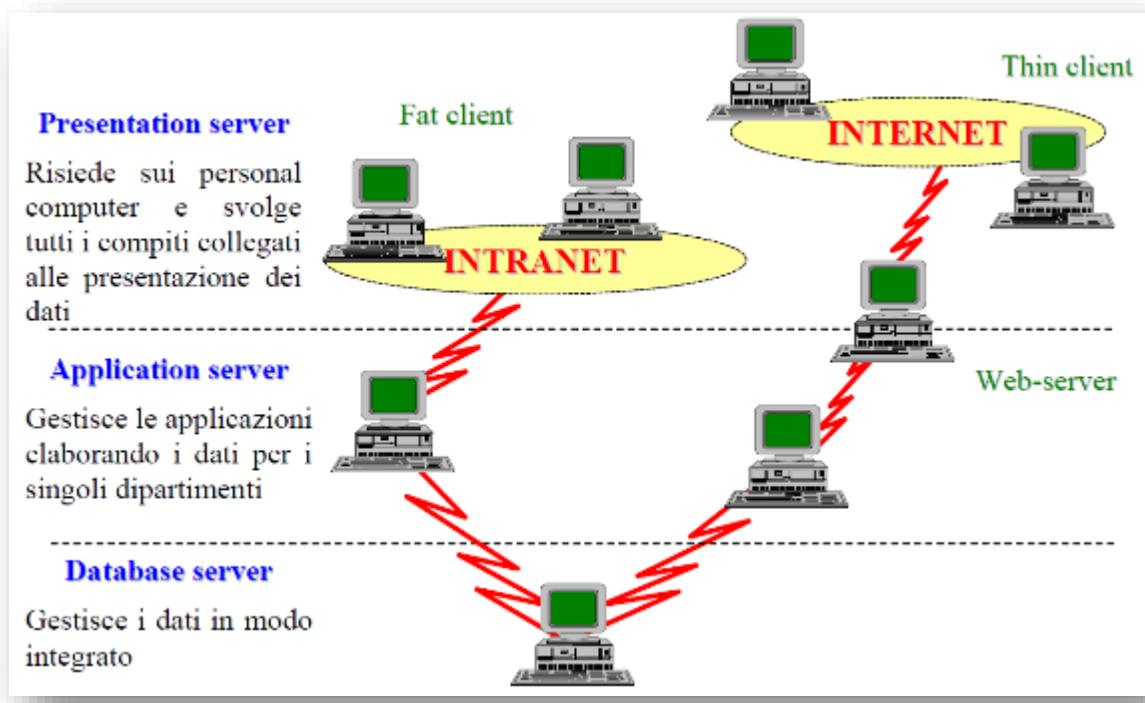
I sistemi ERP incorporano la logica dei processi aziendali. È necessario che i processi aziendali e quelli definiti nell'ERP siano coerenti, solitamente in una dinamica che prevede la modifica dei processi aziendali in funzione di quelli dell'ERP.

Questo può avere un impatto organizzativo elevato, tuttavia:

- I processi negli ERP sono basati su best practices del settore
- Si ha garanzia di correttezza e standardizzazione delle operazioni
- I processi sono razionalizzati favorendo una coerenza tra progetto informatico ERP e progetto di Business Process Reengineering (BPR).

I sistemi ERP sono però personalizzabili per garantire una *possibilità di manovra* allo scopo di ridurre il gap tra standard e specificità aziendali. Il ruolo del *parametrizzatore* è proprio quello di utilizzare le configurazioni del sistema ERP per adattarlo alle esigenze aziendali.

#### *Architettura tipica di un ERP*



#### *ERP nella piccola e media impresa*

Le piccole medie aziende sono caratterizzate da capacità di spesa limitata, complessità aziendale limitata e grande necessità di flessibilità e rapidità di azione. Per far fronte a queste caratteristiche le aziende produttrici di software:

- Realizzano moduli orizzontali semplificati
- Forniscono versioni sottodimensionate dei package standard
- Forniscono una modalità di fruizione Application Server Provider (ASP) che permette di pagare un canone di accesso al posto dell'acquisto.

#### *Sistemi Customer Relationship Management (CRM)*

Questa tipologia di sistemi si occupa di governare in maniera integrata e strutturata il processo di gestione della relazione con la clientela. L'obiettivo di questo processo è costruire relazioni personalizzate di lungo periodo, aumentare la soddisfazione del cliente e il valore dell'impresa per il cliente e, quindi, del cliente per l'impresa.

Il cliente è il centro della strategia commerciale.

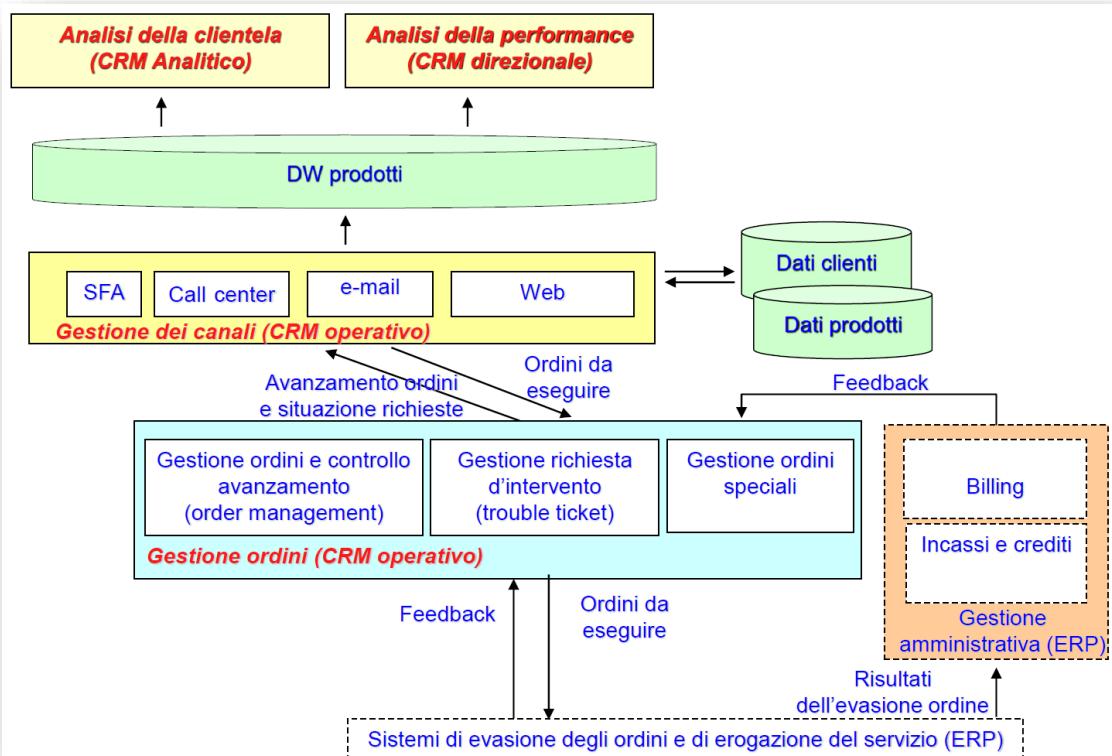
I migliori esempi di approcci CRM sono forniti dalle aziende telefoniche.

I sistemi CRM possono gestire diverse caratteristiche delle strategie aziendali nei confronti della clientela: necessità di relazioni frequenti e continue, necessità di gestire un elevato numero di clienti anche distribuiti geograficamente e che interagiscono tramite diversi canali, in diverse lingue.

Alcuni esempi	Continuità e frequenza della relazione		
	BASSA	ALTA	ALTA
Numerosità della clientela	ALTA	Assicurazioni, utilities, sanità, grande distribuzione, Stato	Banche, compagnie telefoniche, poste
	BASSA	Commodities, Beni di consumo durevole	Trasporti, turismo, beni strumentali, farmaceutica

I sistemi CRM sono composti, principalmente, da tre moduli

- **CRM operativo:** informatizza i canali di contatto con il cliente (presenza, voce, web, posta, ...)
- **CRM analitico:** informatizza l'analisi della clientela al fine di definire le politiche di promozione e contatto; supporta il settore commerciale e marketing nella pianificazione delle proprie attività
- **CRM direzionale:** aiuta il management nella valutazione della performance aziendale verso i clienti; supporta la dirigenza nel valutare l'efficienza del settore marketing e commerciale.



Le caratteristiche di un CRM possono essere riassunte in:

- **Multicanalità:** il cliente sceglie il canale di contatto più conveniente, il servizio deve essere erogato 24h al giorno
- **Completezza e unicità dei dati su prodotti e clienti:** le informazioni sul cliente devono essere condivise dai diversi sistemi di contatto per garantire la multicanalità, questi utilizzeranno una base di dati comune
- **Catene di servizio:** le richieste sottoposte al front-end generano una serie di attività complesse sui sistemi di back-end. Un CRM è più efficiente in funzione di quanto riesce ad interagire con i servizi del Sistema Informativo.

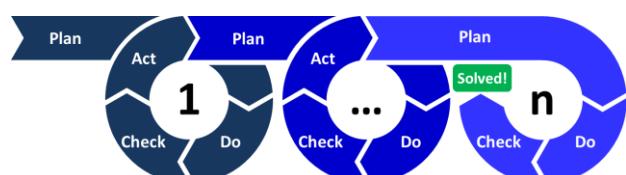
## I canali CRM

<b>Presenza</b>	<ul style="list-style-type: none"> <li>Il cliente interagisce direttamente con l'azienda</li> <li>Il CRM supporta il venditore nell'individuazione, contatto, trattativa e raccolta ordini del cliente</li> <li>Si parla di applicazioni <b>Sales Force Automation</b> (SFA) in quanto principalmente indirizzate ai vendori</li> <li>Adottati fin dagli anni '80</li> </ul>
<b>Voce</b>	<ul style="list-style-type: none"> <li>Il cliente interagisce al telefono con una serie call center e di operatori assistiti dal sistema CRM</li> <li>Le implementazioni informatiche supportano: <ul style="list-style-type: none"> <li>✓ Lo smistamento delle chiamate</li> <li>✓ Risposte e servizi automatici/self service</li> <li>✓ La trasparenza della località dei call center</li> <li>✓ Gli operatori</li> </ul> </li> </ul>
<b>Corrispondenza</b>	<ul style="list-style-type: none"> <li>Il cliente interagisce via mail</li> <li>Gli operatori e/o i sistemi informatici smistano le mail in entrata</li> <li>Si possono implementare sistemi automatici di invio mail per campagne marketing o servizi informativi</li> </ul>
<b>WEB</b>	<ul style="list-style-type: none"> <li>Negli ultimi anni è diventato il principale canale di commercio grazie a: <ul style="list-style-type: none"> <li>✓ Disponibilità dei servizi self service 24 ore su 24</li> <li>✓ Possibilità di personalizzazione per ogni utente</li> <li>✓ Informazioni chiare al cliente circa lo stato degli ordini</li> <li>✓ Registrazione di tutto il processo di navigazione WEB del cliente (per analisi)</li> </ul> </li> <li>I servizi WEB si classificano in: <ul style="list-style-type: none"> <li>✓ <b>Business-to-Customer (B2C)</b>: punto di accesso per gli utenti finali</li> <li>✓ <b>Business-to-Business (B2B)</b>: punti di accesso per aziende partners/rivenditori</li> </ul> </li> </ul>

## Ingegneria dei processi

### Il ciclo dell'innovazione

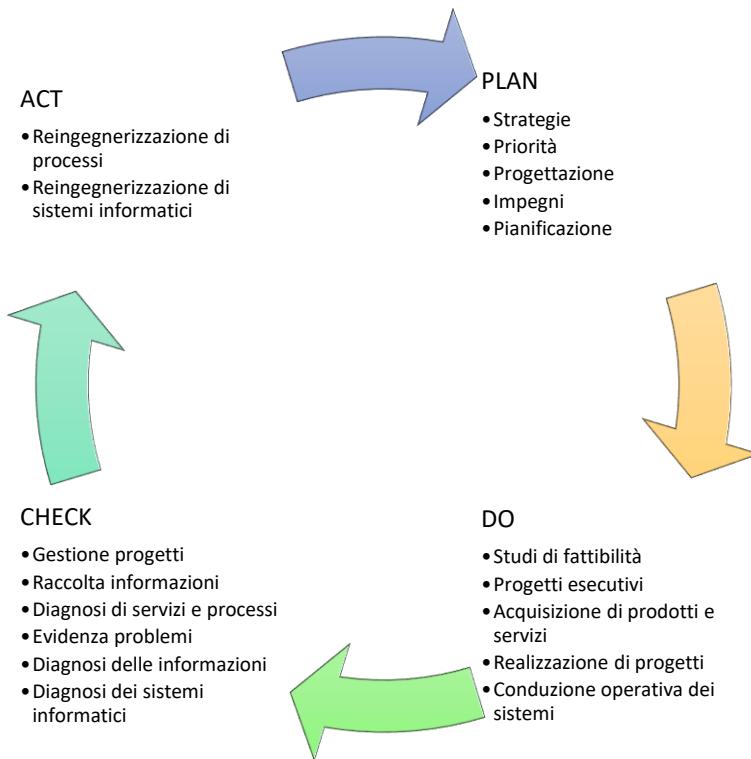
Il **ciclo di Deming** è un metodo di gestione iterativo in quattro fasi utilizzato per il controllo e il miglioramento continuo dei processi e dei prodotti, rappresenta una sintesi di gestione del ciclo dell'innovazione. È diviso in quattro fasi:



- **Plan** (Pianificazione): definizione degli obiettivi e dei processi necessari per ottenere i risultati attesi
- **Do** (Esecuzione): esecuzione delle azioni operative per eseguire un processo, creare un prodotto; in questa fase vengono generati e raccolti i dati
- **Check** (Test, controllo): studio dei dati e analisi dei risultati, misurazione delle azioni effettuate, valutazione dello stato di raggiungimento degli obiettivi, ricerca delle deviazioni
- **Act** (Azione): azioni di consolidamento o miglioramento del processo, definite in funzione dell'analisi effettuata nella fase precedente e coerentemente con le cause di eventuali problemi rilevati.

Il ciclo di Deming è iterativo e continuo.

William Edwards Deming (Sioux City, 14 ottobre 1900 – Washington, D.C., 21 dicembre 1993) è stato un ingegnere, saggista, docente e consulente di gestione aziendale e manager statunitense. A Deming fu ampiamente riconosciuto il merito per gli studi sul miglioramento della produzione negli Stati Uniti d'America durante la seconda guerra mondiale, anche se egli è forse più noto per il suo lavoro in Giappone (Ciclo di Deming).



## Definizione degli obiettivi

Gli obiettivi devono essere formalizzati al fine di garantire il corretto svolgimento dei processi volti al loro raggiungimento. Le aziende redigono **piani di informatizzazione** che devono essere valutati e approvati dai livelli decisionali al fine di ottenere la corretta rilevanza:

- **Piano strategico**: medio termine (3-5) anni, è necessario per evitare di costruire sistemi frammentati, incoerenti, incompatibili dal punto di vista tecnologico; contiene obiettivi strategici, architetture tecnologiche e applicative, progetti di grande rilievo.
- **Piano operativo**: ha validità annuale, definisce in maniera dettagliata gli interventi previsti dal piano strategico da attuare nell'anno in corso.

Gli obiettivi definiti in questi due piani possono essere oggetto di correzioni valutate continuamente.

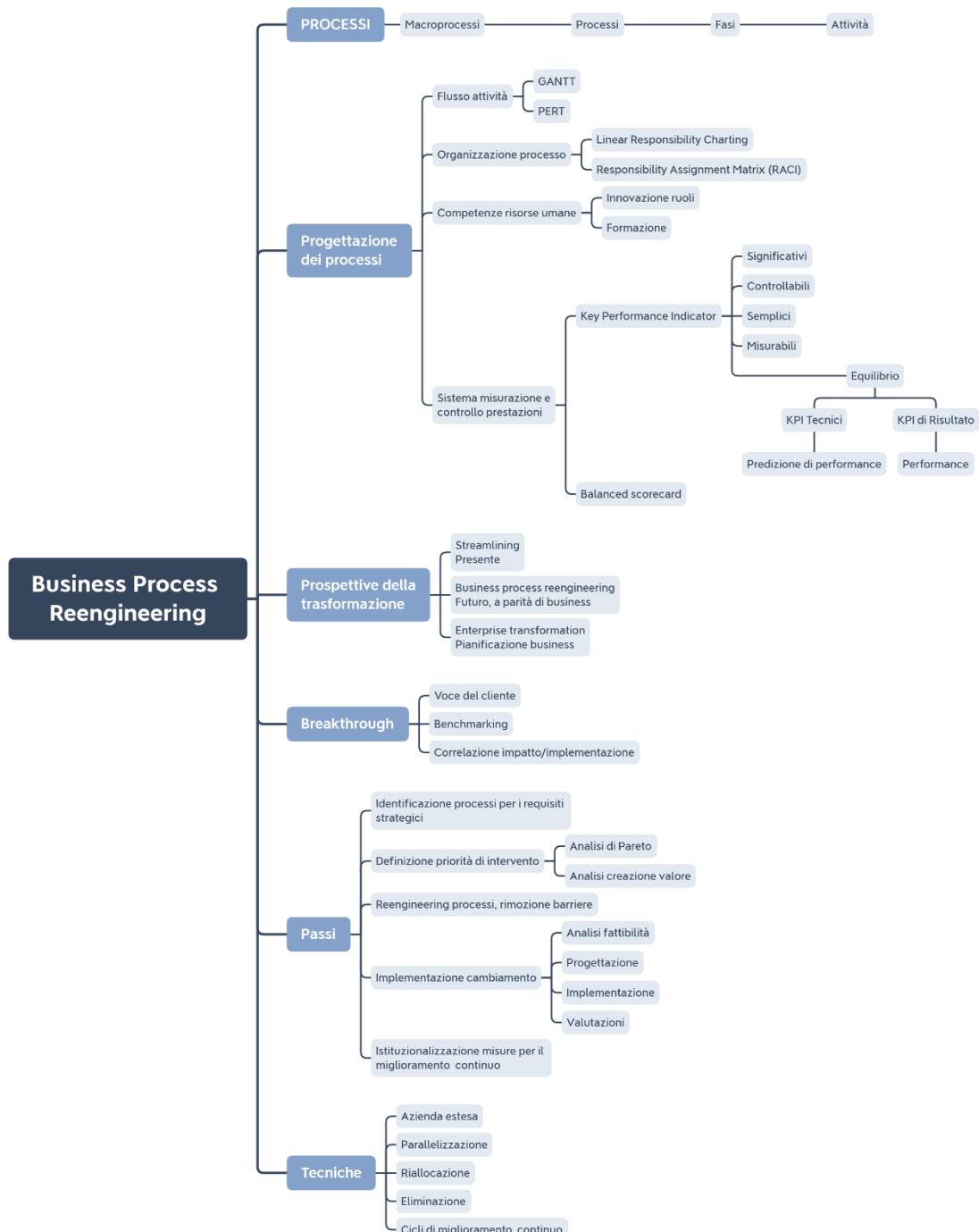
Obiettivi Top-down	Obiettivi bottom-up
Gli obiettivi sono definiti a partire dal piano strategico; spesso contengono forte componenti di innovazione e mutamenti radicali, possono essere recepiti con maggiori difficoltà generando tensioni e rischi organizzativi.	Gli obiettivi vengono sintetizzati dalle valutazioni e indicazioni delle diverse unità operative secondo un approccio più conservativo, riducendo le innovazioni radicali; gli obiettivi sono solitamente facili da raggiungere.

Nella pratica i due approcci vengono utilizzati in maniera complementare cercando di mediare tra le loro caratteristiche speculari.

## Innovazione dei Sistemi Informativi: Business Process Reengineering (BPR)

L'innovazione dei sistemi informativi, oggi, non riguarda solo l'accesso e l'utilizzo delle tecnologie innovative o largamente diffuse, si tratta di mettere in pratica il concetto di Business Process Reengineering già espresso all'inizio degli anni '90 per impulso di Michael Hammer (professore di informatica presso il MIT): **"è ora di smettere di pavimentare i sentieri delle mucche. Anziché incorporare processi obsoleti in silicio e software, dovremmo cancellarli e ricominciare. Dovremmo "reingegnerizzare" il nostro business: utilizzare la potenza della moderna tecnologia informatica per ridisegnare radicalmente i nostri processi aziendali al fine di ottenere miglioramenti straordinari nelle loro prestazioni"**.

Il BPR riguarda un completo ripensamento dei fondamentali processi di un'organizzazione: sebbene semplicemente informatizzare un processo esistente (mantenendolo com'è, utilizzando l'informatica come "supporto") determini già di per sé aumenti dell'efficienza, il BPR suggerisce una completa rivalutazione dei processi, alla luce delle tecnologie disponibili e delle potenzialità offerte dalle innovazioni dei sistemi informatici.



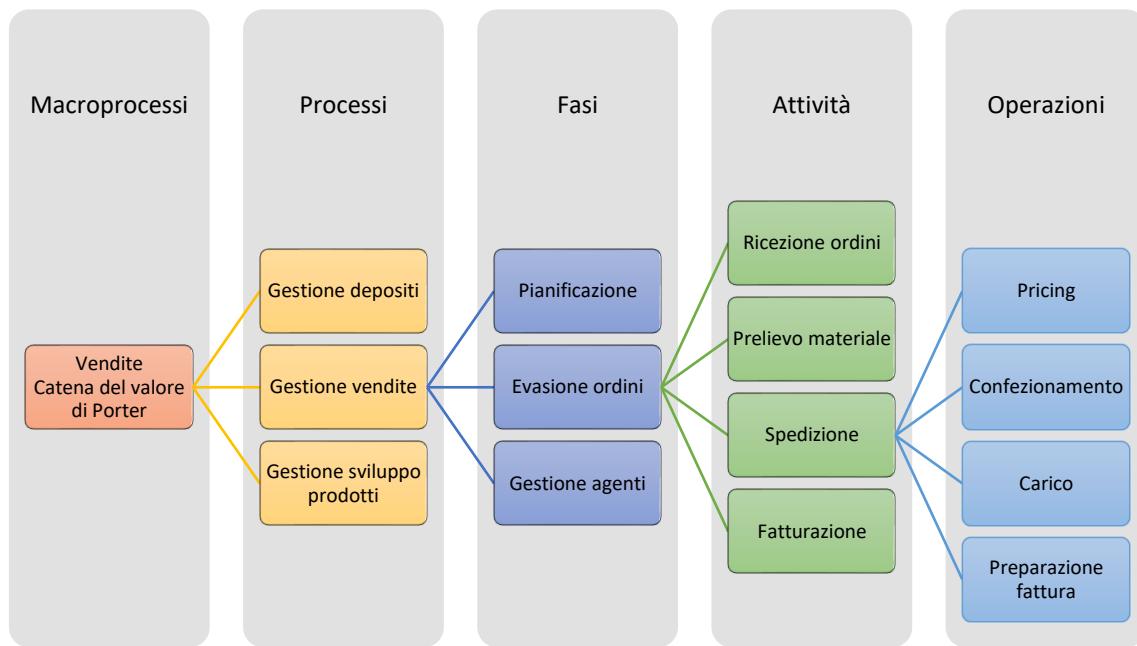
## I processi aziendali

Un **processo** è l'insieme delle attività tra loro interrelate, finalizzate alla realizzazione di un risultato definito e misurabile che contribuisce al raggiungimento della missione aziendale.

Il **cliente** è l'entità che richiede o utilizza un prodotto o un servizio realizzato tramite un processo. Un cliente può essere *interno* oppure *esterno* all'azienda. Ogni output può essere per molti clienti o un cliente può richiedere molteplici output. Il primo compito del **progettista** consiste nella descrizione e classificazione dei processi (dal punto di vista normativo in primis).

I processi possono essere categorizzati per successivi livelli di approfondimento e dettaglio:

- **Macroprocesso:** utile soprattutto nelle fasi iniziali per strutturare un'analisi e individuare le aree di intervento, es. catena del valore di Porter
- **Processo:** descrive ad un livello ragionevolmente dettagliato le operazioni svolte dall'azienda, es: il macroprocesso *sviluppo di prodotti* può essere decomposto in concept, pianificazione, progettazione prototipazione, test, ingegnerizzazione
- **Fase:** è una parte formalizzata e specifica di un processo, un processo è dettagliato in fasi, es. il processo *progettazione* può essere scomposto in sviluppo specifiche, concurrent engineering management, progettazione, documentazione specifiche, sviluppi prototipi, gestione richieste di brevetto.
- **Attività:** livello di massimo dettaglio, ogni fase è scomposta in attività sequenziali per studiare i processi.



La progettazione dei processi deve considerare le seguenti variabili:

- **Flusso delle attività:** sequenza di attività per l'esecuzione del processo, determina la durata, incide sul livello di servizio, determina – assieme alle risorse umane e alla tecnologia – la qualità dell'output; può essere modellato tramite UML
- **Organizzazione del processo:** modella i raggruppamenti delle attività dal punto di vista operativo e di coordinamento e controllo; può essere modellata tramite diagrammi Linear Responsibility Charting (LRC)
- **Competenze delle risorse umane**
- **Sistema di misurazione e controllo delle prestazioni:** sistemi di Key Performance Indicator (KPI) o Balanced Scorecard.

## Linear Responsibility Charting

Una Linear Responsibility Chart descrive la partecipazione dei diversi ruoli di un'organizzazione nell'esecuzione delle attività di un processo di budget. L'acronimo RACI (Responsible, Accountable, Consulted, Informed, in italiano

rimodulata in DEAI: Decisore, Esecutore, Assistente, Informato) rappresenta le diverse responsabilità. LRC viene utilizzata per definire e chiarire ruoli e responsabilità in progetti e processi multi-dipartimentali con funzioni trasversali.

### Esempio 1

#### Responsibility Assignment Matrix - RACI Chart

	Jeff	Michael	Ringo	YOU	Alex	Anita	Bill	Cindy	Felix	Fred	Hans	John	Lidia	Luc	Marc	Paul	Peter	Sue	Ted	Tim
Planning / Schedule	R	A	I	C				C											Q	
Risk Management		I	I	Q						A								R		
Quality Management		R	C							R									A	
Procurement		R		Q					R	R							R		A	
1. Specifications Listing					A			R									R		R	
2. Site Requirements	C	A	R	Q						R								R		
3. Call for Tenders			Q	A	R	C				R							R			
4. Budget Approval			A	Q					R							R			R	
5. Contract Negotiations	A		Q	R	R											R				

\* R – Responsible (works on), A – Accountable, C – Consulted, I – Informed, Q – Quality Reviewer

### Esempio 2

	Filiale di vendita	Direzione commerciale	Direzione distribuzione	Magazzino prodotti	Spedizione	Contabilità clienti
Ricezione ordine	E	D	I	I		A
Evasione degli ordini	I		D	E	I	
Spedizione			I	A	E	I
Fatturazione			I			E

#### Balanced scorecard

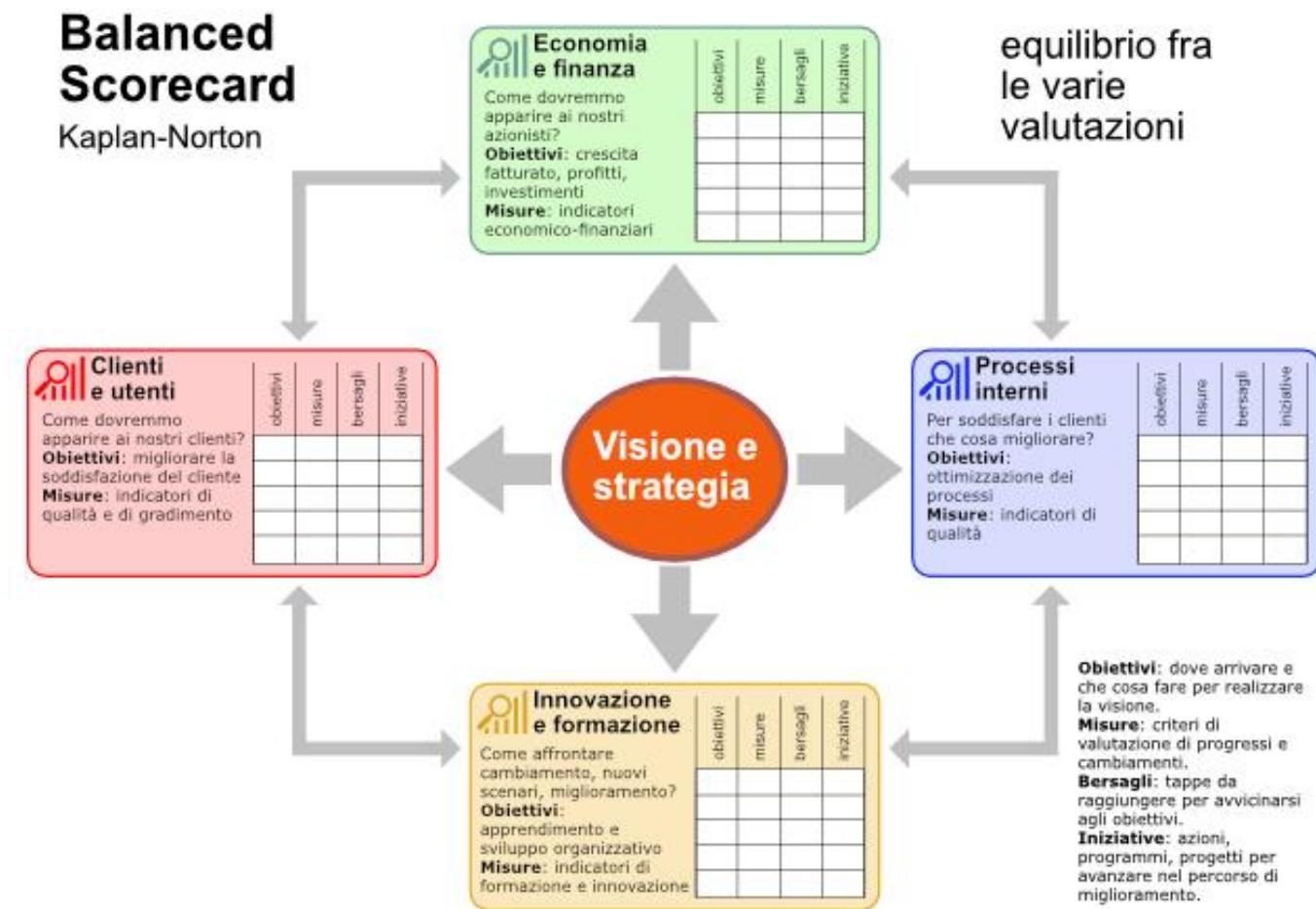
Fonte: <http://www.umbertosantucci.it/atlante/balanced-scorecard/>

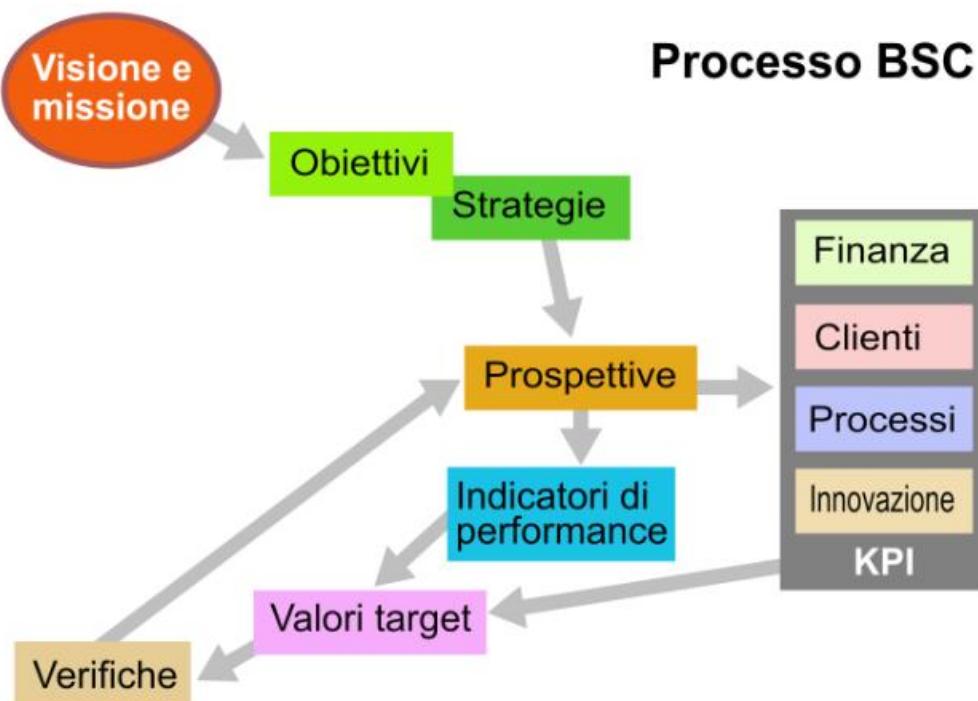
Il metodo della valutazione bilanciata delle prestazioni è stato proposto dagli studiosi americani Robert Kaplan e David Norton nel 1992. Inizialmente veniva usata come strumento di valutazione omogenea di settori diversi di un'organizzazione, poi diventò un vero e proprio metodo di valutazione strategica utile non solo a fotografare la situazione attuale, ma ad influenzare gli sviluppi futuri.

Le aziende spesso mancano di una visione chiara e di una strategia per realizzarla, oppure la strategia non è condivisa, non è allineata con le risorse disponibili, i processi non seguono priorità strategiche, l'organizzazione, la ricerca, la formazione, le politiche del personale non sono coerenti con la strategia. Imprenditori e manager devono ogni giorno risolvere problemi che potevano essere evitati o delegati ad altri, i dipendenti lavorano senza tener conto degli obiettivi aziendali, i reparti non comunicano bene fra loro, i margini di profitto si riducono. Spesso l'imprenditore ha una strategia, ma non sa comunicarla ai collaboratori.

Per ovviare a tutto ciò, il metodo parte dalla definizione della visione (come vogliamo diventare) e la missione (perché esistiamo), e della strategia per realizzare entrambe. La strategia definisce le cose da fare e i sistemi di misurazione di stadi di avanzamento delle attività e dei risultati delle prestazioni.

Per evitare che un settore dell'organizzazione o un aspetto della strategia prenda il sopravvento sugli altri, il metodo propone quattro punti di vista che devono bilanciarsi fra di loro. Ad esempio se si tagliano la formazione e la ricerca si migliora il bilancio annuale, ma a spese delle prospettive future.



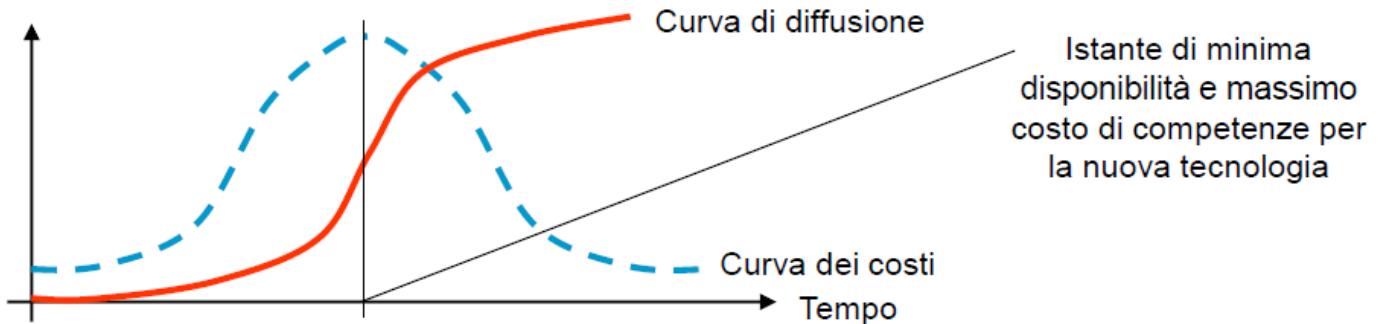


Il processo di applicazione del metodo parte dalla definizione della vision e/o della mission, da cui si definiscono gli obiettivi generali e la strategia per raggiungerli. Per ognuna delle quattro aree si individuano i fattori critici di successo (kpi, key point indicator), gli indicatori di performance (che cosa misurare e come farlo) e i valori target (livelli o quantità da raggiungere entro tempi stabiliti), infine le azioni da intraprendere per realizzare il tutto. Compiuto questo ciclo si fanno le verifiche e si inizia un nuovo ciclo, nello spirito del miglioramento continuo.

Il confronto delle valutazioni nel tempo dà indicazioni precise sui cambiamenti realizzati, al di là delle sensazioni personali, dei timori e delle speranze.

### Le risorse umane

Le risorse umane sono un fattore strategico di fondamentale importanza e determinano la differenza tra risultato massimo teoricamente raggiungibile e risultato effettivo. L'innovazione dei processi e quella informatica richiedono nuovi ruoli, nuove figure professionali il cui reperimento può essere costoso e complesso, la formazione è un ulteriore elemento chiave in grado di supportare un incremento delle performance ma è anch'essa costosa e potrebbe avere un impatto non sempre corrisposto nelle disponibilità del personale dell'organizzazione.



### Key Performance Indicator

La capacità di un processo di soddisfare i requisiti dei clienti si misura tramite i **Key Performance Indicator (KPI)**. I KPI sono indicatori quantificabili dell'efficacia e dell'efficienza di un processo o di un sottoprocesso.

Un KPI è **quantificabile, rilevabile, correlato con l'obiettivo di business interno del processo**.

Un KPI è applicabile ad un processo descritto chiaramente in termini di attività, output, obiettivi.

## Proprietà dei KPI

**Significatività:** i KPI devono essere collegati agli obiettivi strategici.

**Controllabilità:** i KPI devono misurare risultati influenzabili dal processo o azioni che possono essere intraprese; devono focalizzarsi su un determinato periodo di tempo.

**Semplicità:** i KPI devono essere chiari, pochi, se possibile devono poter essere traslati da/su altri sistemi di misurazione.

**Misurabilità:** i dati necessari a calcolare un KPI devono essere disponibili e devono avere un livello appropriato di qualità e accuratezza.

**Equilibrio:** i KPI tecnici devono essere collegati alle attività di un processo; i KPI di risultato devono essere collegati con gli obiettivi e gli output di un processo.

I KPI di risultato misurano la performance in un dato momento; i KPI tecnici misurano attività che influenzano la performance.

## Esercitazione sui KPI

### Case study

Una grande azienda vende prodotti in un grande numero di negozi in giro per il mondo. I negozi vendono un grande numero di prodotti in ogni negozio. L'azienda vuole monitorare come gli *store managers* gestiscono l'esposizione dei prodotti in un panorama temporale di 6 mesi.

Gli esperti di dominio forniscono queste informazioni:

- Gli articoli non esposti non possono essere venduti
- È difficile vendere articoli esposti in quantità minime: una quantità ottimale è di 5 unità per prodotto
- Mantenere in esposizione lo stesso articolo per un'intera stagione riduce le attrattive dei clienti che potrebbero decidere di non tornare successivamente in negozio.

Ogni magazzino è collegato al sistema informativo della azienda e memorizza informazioni relative alle vendite e alle giacenze.

Ogni store è strutturato con due magazzini: il magazzino giacenze vero e proprio e la showroom; i prodotti in giacenza presso la showroom sono quelli esposti in visione ai clienti.

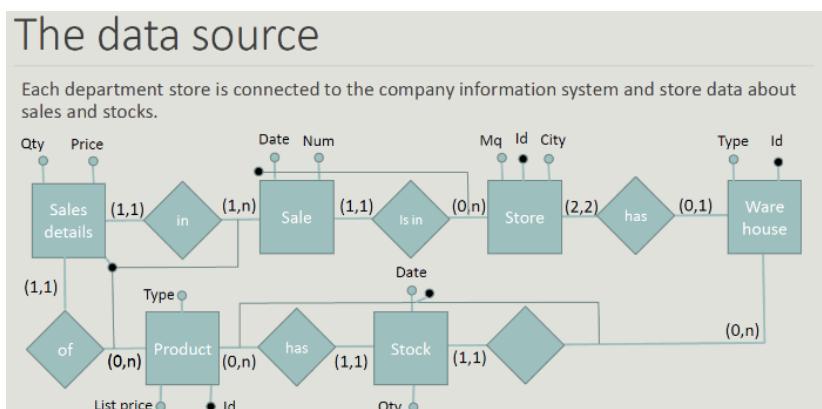
Esiste un magazzino centrale che fornisce le merci a tutti i negozi.

Alla chiusura, le giacenze di ogni prodotto vengono aggiornate per ogni magazzino e negozio e vengono aggiunte le tuple nelle tabella delle giacenze, storizzando le giacenze: `<store_x, warehouse_y, date(), product_z, stock_qty>`.

### Data source

Descrizione sintetica entità:

- I negozi
- Le vendite (scontrini)
  - ✓ Testate
  - ✓ Righe (prodotti venduti)
- I Prodotti
- I magazzini del negozio (ogni negozio ha 2 magazzini ed un magazzino può essere relativo ad un magazzino)
  - ✓ Un magazzino rappresenta l'esposizione (showroom)
  - ✓ Un magazzino rappresenta il magazzino vero e proprio (back shop)
  - ✓ Un magazzino può non essere collegato ad un negozio: si tratta del magazzino centrale



- Le giacenze, di ogni prodotto, in ogni giornata, in ogni magazzino

### Dagli obiettivi alle misure

L’azienda vuole monitorare la gestione dell’esposizione dei prodotti per ogni negozio.

In dettaglio:

- Quanto è lontano ciascun negozio dall’esposizione ottima e perché?
  - ✓ Il negozio ha troppa merce da esporre?
  - ✓ Il negozio ha poca merce da esporre?
- Le politiche di esposizione cambiano nei diversi negozi?
- È possibile identificare managers più o meno efficienti?
- Qual è la rotazione? Quanto viene aggiornata l’esposizione?
- Il numero di articoli è compatibile con le dimensioni del negozio?
- L’esposizione cambia durante la stagione? Durante i saldi?
- Qual è l’impatto dell’esposizione sulle vendite?

### Stakeholders

- **Category manager:** referente per un certo tipo di prodotti, organizza gli acquisti per quel tipo di prodotti
  - ✓ Interessato al layout di prodotto per questa categoria di prodotti
- **Supervisor manager:** referente per una specifica area, decide le politiche per area geografica
  - ✓ Interessato ad ogni singolo negozio
- **Layout manager:** suggerisce il numero di items da esporre e le relative quantità
  - ✓ Layout di ogni negozio, gruppi di negozi, categorie

### KPI

Descrizione	Formula	Range
Percentuale di prodotti esposti rispetto a quelli disponibili in magazzino <i>Obiettivo: arrivare al valore 1 prima dei saldi.</i>	$\frac{ Stock.Product.Type \text{ in ShowRoom} }{ Stock.Product.Type }$	[0,1]
Percentuale tempo di esposizione per articolo (per negozio, per categoria)	$\frac{\sum Stock.Product.Type \times days \text{ in ShowRoom}}{PeriodLength}$	[0,1]
Percentuale articoli venduti rispetto a quelli esposti e al tempo di esposizione (per negozio, per categoria)	$\frac{\sum SalesDetails.Product.Type.Qty}{\sum Stock.Product.Type.Qty \times days \text{ in ShowRoom}}$	[0,1]
Scostamento tra ricavi effettivi e teorici rispetto a prezzo di listino (per negozio, per categoria)	$\frac{\sum SalesDetails.Product.Type.Qty \times Price}{\sum SalesDetails.Product.Type.Qty \times ListPrice}$	>=0
Ritardo tra arrivo in magazzino inizio esposizione (per negozio, per categoria)	$Stock.FirstDay \text{ in ShowRoom} - Stock.FirstDay$	>=0
Durata delle vendite	$SalesDetails.LastDate - SalesDetails.FirstDate$	
Quantità media esposta per item rispetto alla quantità ottima (5 unità) <i>Fragmentation Index</i>	$\frac{\sum Stock.Product.Type.Qty \times days \text{ in ShowRoom}}{\sum 5 \times days \text{ in ShowRoom}}$	>=0
Display Refresh Index (DRindx) <i>Misura l’innovazione di esposizione</i>	$1 - \frac{ ExposedItems_{TimeBin1} \cap ExposedItems_{TimeBin2} }{ ExposedItems_{TimeBin2} }$	
Densità di esposizione	$\frac{\sum Stock.Product.Type.Qty \times days \times space}{Store.Mq \times days}$	[0,1]

## Business Process Reengineering

Da Wikipedia

**Kaizen (改善)** è la composizione di due termini giapponesi, KAI (cambiamento, miglioramento) e ZEN (buono, migliore), e significa cambiare in meglio, miglioramento continuo. È stato coniato da Masaaki Imai nel 1986 per descrivere la filosofia di business che supportava i successi dell'industria nipponica negli anni '80 con particolare riferimento alla Toyota, tanto da rappresentare il sinonimo di Toyotismo.

Nel contesto in cui il termine è stato coniato, Kaizen viene tradotto con "miglioramento continuo" perdendo di originalità rispetto al Ciclo di Deming dal quale deriva ma con il quale non coincide.

Il Kaizen come *pratica economica* è riferito all'efficienza dei fattori produttivi legati alla microeconomia aziendale attraverso lo sviluppo di Sistemi di Gestione finalizzati al contenimento dei costi di produzione.

Il Kaizen come *approccio per i sistemi di gestione per la Qualità* è correlato a concetti quali:

- il Lean manufacturing (produzione snella);
- il Total Quality Management (TQM - Gestione della qualità totale);
- il Just in time (JIT - abbattimento delle scorte);
- il kanban (metodo per la reintegrazione costante delle materie prime e dei semilavorati);
- la Riprogettazione dei processi aziendali;
- lo Statistical process control (controllo statistico dei processi).

Il Kaizen come *strategia comportamentale* si riferisce ad una pratica diretta al miglioramento costante dei processi manifatturieri, Ingegneristici e di business management secondo una logica bottom-up che recentemente ha trovato applicazione nella sanità, Psicoterapia, Coaching, oltre ad altre industrie non manifatturiere quali istituti bancari ed industrie del terziario avanzato.

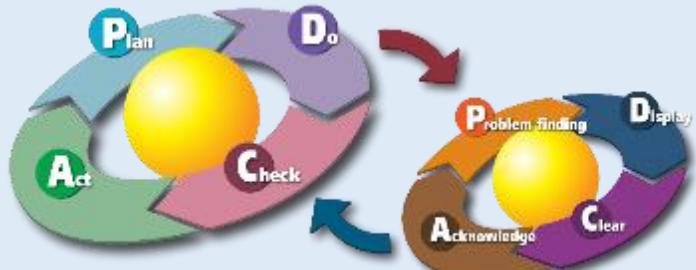
La vision della strategia Kaizen è quella del rinnovamento a piccoli passi, da farsi giorno dopo giorno, con continuità, in radicale contrapposizione con concetti quali innovazione, rivoluzione e conflittualità di matrice squisitamente occidentale. La base del rinnovamento è quella di incoraggiare ogni persona ad apportare ogni giorno piccoli cambiamenti il cui effetto complessivo diventa un processo di selezione e miglioramento dell'intera Organizzazione.

### Il ciclo del Kaizen

L'implementazione di un modello di gestione Kaizen presuppone un'elevata ingegnerizzazione dei processi in fase di progettazione unitamente al massimo controllo sugli stessi sulla falsariga del Ciclo di Deming. La ragione di questo modello risiede nella massima riduzione dei controlli "a valle" (tests distruttivi e non, misurazioni empiriche e verifiche direttamente sui prodotti) in favore dei "controlli concomitanti", dei controlli "a monte" e di un atteggiamento proattivo generale finalizzato all'eccellenza.

In massima sintesi, il ciclo Kaizen può essere definito come:

- Ricerca della **standardizzazione** massima delle operazioni, dei processi e delle attività;
- **Misurazione** dei processi (non quindi degli outputs di processo), delle operazioni e delle attività con riferimento al consumo di risorse ed ai cicli temporali per l'esecuzione (da cui deriva una progressiva razionalizzazione dei processi e non un incremento degli outputs se non in termini di riduzione dei cicli);
- **Valutazione delle misurazioni** e non dei requisiti dei processi e progressivo aggiustamento e miglioramento;
- **Innovazione** (ovvero, detto alla Kuhn, cambio di paradigma) solo quando questo ha esaurito le possibilità di ulteriori sviluppi ovvero perde di efficienza e di conseguenza di legittimazione;
- **Standardizzazione dell'innovazione ed implementazione attraverso un Blitz** (cfr. infra Kaizen Blitz) minimizzando i tempi fra concezione ed applicazione;
- **Ripetizione del ciclo ad infinitum.**



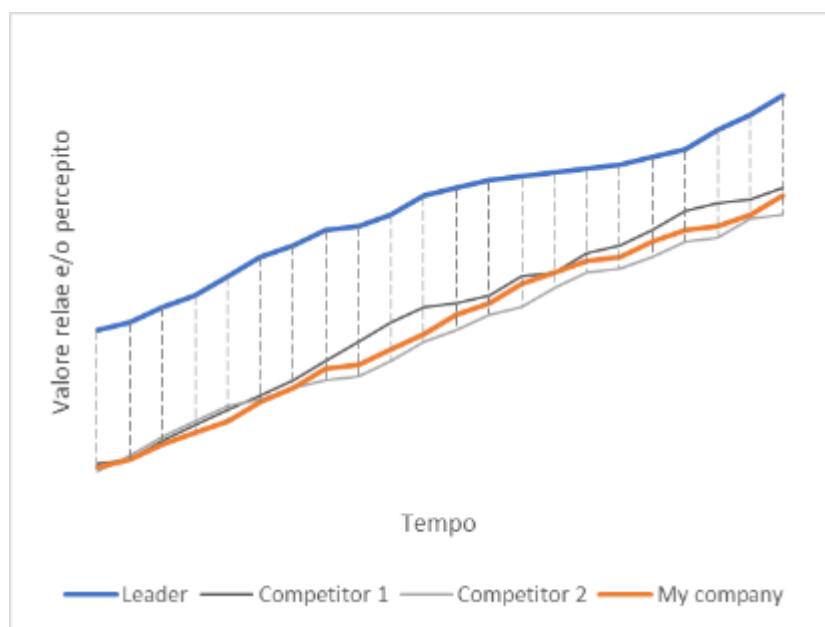
**Il Business Process Reengineering (BPR)** è un intervento organizzativo di profonda revisione dei procedimenti operativi che non risultano più adeguati alle necessità aziendali. Per processo si intende un insieme di attività interconnesse che portano ad un risultato finale identificabile dal cliente, che quindi contribuisce alla formazione di valore per l'azienda.

Sinonimi utilizzati per il BPR sono: business process redesign, business transformation, business process change management.

Il BPR permette di realizzare un *salto di qualità (breakthrough)* necessario per le aziende che agiscono in un determinato mercato: il miglioramento continuo infatti garantisce un incremento del valore aziendale (e/o della percezione del valore aziendale da parte del cliente) ma tale incremento è in qualche modo diffuso e/o omogeneo alle altre aziende competitor o leader nella stessa area di mercato.

Le diverse componenti del valore che possono essere revisionate tramite BPR sono:

- Rapidità dei tempi di ciclo (efficienza)
- Riduzione o eliminazione dei difetti (efficacia)
- Livello del servizio (efficacia)
- Livello di produttività (efficienza).



Varie modalità di reingegnerizzazione sono definite dallo scopo e dall'ampiezza del cambiamento:

### Streamlining

#### pianificazione del presente

Ricerca, attraverso un approccio incrementale sui processi, del miglioramento delle prestazioni (modifica sequenze di lavoro, semplificazione o automazione di attività, ridefinizione input e output, bilanciamento carichi di lavoro, ...).

### Business process reengineering

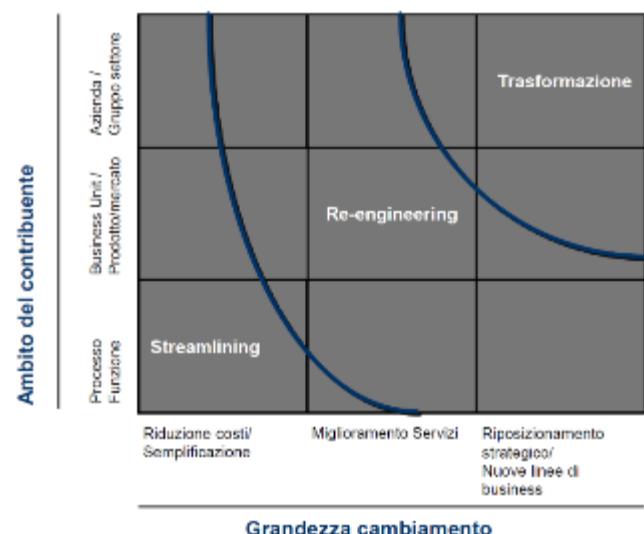
#### pianificazione del futuro, a parità di business

Approccio per la ricerca di cambiamenti radicali nei processi esistenti (eliminazione, fusione, combinazione di processi, ricostruzione da zero, introduzione nuove competenze, implementazione tecnologie impattanti sull'intero processo, riprogettazione del sistema premiante, ...).

### Enterprise transformation

#### pianificazione del business

Approccio radicale per la ricerca di riconfigurazione del business aziendale (eliminazione o sviluppo di combinazioni di prodotti e mercati, sviluppo di alleanze strategiche, revisione del portafoglio di business, riallineamento processi di supporto a nuove combinazioni di processi di business, ...).

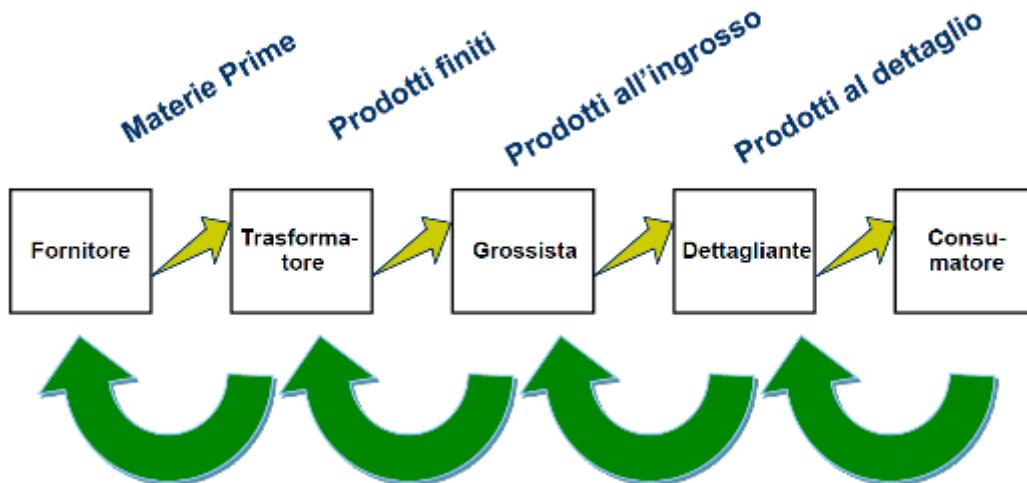


## Breakthrough

Le performance dei processi possono essere misurate in termini di costi, qualità, servizi, tempi. Esistono livelli di performance che determinano miglioramenti significativi e più che proporzionali dei benefici, essi sono chiamati **breakthrough**.

È possibile identificare gli ambiti, le condizioni, le opportunità di breakthrough attraverso due modalità fondamentali:

- La *voce del cliente*: il consumatore rappresenta il cliente finale di un prodotto o di un servizio, nella catena produttiva non è però l'unico cliente; ogni cliente anche intermedio può fornire informazioni utili al miglioramento del business e dei processi di un'azienda ed è di fondamentale importanza per l'azienda porre particolare attenzione alle esigenze espresse da chi acquista materie prime, prodotti finiti o servizi per la distribuzione o per il consumo in quanto esiste il rischio per un'organizzazione legato alla possibilità di offrire ciò che il cliente non richiede con conseguenti ricadute sulla posizione nel mercato, sulla perdita di clienti chiave, sull'incapacità di focalizzare il BPR sulle aree più a valore.



- Il *benchmarking*: l'organizzazione può confrontare le proprie performance con standard esterni definiti ed eccellenze; l'obiettivo del benchmarking è individuare il posizionamento aziendale e ottenere e mantenere livelli di performance adeguati alla missione aziendale attraverso iniziative di miglioramento; inoltre il benchmarking può suggerire soluzioni innovative e accelerare processi di cambiamento con considerevoli livelli di efficacia anche nella comunicazione.

Oltre alle due modalità descritte, è utile considerare che non tutti i processi e le loro applicazioni in differenti domini applicativi o di business sono ugualmente semplici o complessi: fattori esogeni che impattano sulla revisione sono la complessità organizzativa e quella socio-politica; inoltre è importante saper focalizzare gli interventi sui processi che contribuiscono maggiormente alla creazione di valore per gli azionisti e per i clienti: possono esistere processi inefficienti ma legati ad aree a “basso valore” per le quali non è necessario intervenire (o comunque non è necessario farlo con priorità).

La correlazione di due fattori, impatto potenziale e difficoltà di implementazione (o tempo di realizzazione), permette di valutare quali interventi possono essere prioritari rispetto ad altri.

Impatto potenziale	ALTO	Massima priorità	Interventi sfidanti
	BASSO	Interventi conservativi	Minima priorità
		BASSA/VELOCE	ALTA/LENTO
		Difficoltà di implementazione / tempo di realizzazione	

Per migliorare la capacità di identificazione dei processi sui quali focalizzare gli interventi di BPR, può essere utile classificare i processi in funzione dell'output strategico:

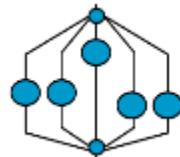
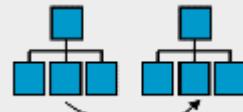
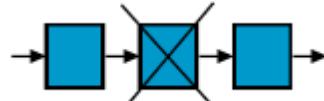
- Processi *di leva*: produzione di ottimizzazione costi e/o benefici
- Processi *chiave*: miglioramento della posizione competitiva
- Processi *di supporto*: miglioramento dell'efficienza
- Processi *di opportunità*: miglioramento dell'efficacia e creazione di opportunità.

### Passi necessari

In sintesi, i passi necessari per un progetto di BPR sono:

1. Identificazione dei processi necessari per soddisfare i requisiti strategici e del cliente
2. Definizione delle priorità di intervento in base all'impatto sul mercato e ai risultati di business
3. Reengineering dei processi per rimuovere le barriere alla performance (effetto su *tutti* gli aspetti dell'organizzazione)
4. Implementazione del cambiamento
  - a. Analisi di fattibilità
  - b. Progettazione
  - c. Implementazione
  - d. Valutazioni
5. Istituzionalizzazione delle misure per il miglioramento continuo.

### Tecniche BPR e Strumenti analitici

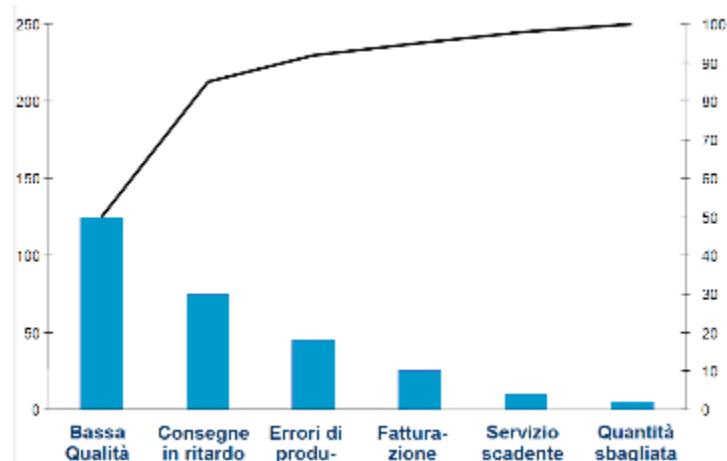
<b>Azienda estesa</b>	Il BPR consiste nell'estensione dei processi all'esterno dei confini dell'impresa	Alleanze con fornitori Condivisione di informazioni Team di processo misti Reti interaziendali ...	
<b>Parallelizzazione</b>	Il BPR consiste nel parallelizzare attività che prima erano svolte in sequenza	Pianificazione di processi simultanei Disegno di concorrenze Analisi conflitti ...	
<b>Riallocazione</b>	Il BPR prevede di riallocare responsabilità, attività e specializzazioni	Certificazione dei fornitori Introduzione ordini da remoto ...	
<b>Eliminazione</b>	Il BPR permette di eliminare attività a basso valore (eventualmente inserendone altre)	Eliminazione solleciti e rilavorazioni Manutenzioni preventive Miglioramento qualità ...	
<b>Cicli di miglioramento continuo</b>	Il BPR attiva anelli di feedback per stimolare il miglioramento continuo	Monitoraggio livelli di servizio Confronto con organizzazioni simili ...	

Una tecnica per il BPR prevede l'**analisi dei tempi di ciclo**, tramite la quale identificare possibili aree di miglioramento dell'efficienza in termini di *tempi morti*.

L'**analisi di Pareto** facilita l'individuazione di aree a *maggior impatto*.

Il diagramma di Pareto è un grafico che rappresenta l'importanza delle differenze causate da un certo fenomeno. Esso contiene al suo interno un grafico a barre e un grafico a linea, dove ogni fattore è rappresentato da barre poste in ordine decrescente e la linea rappresenta invece una distribuzione cumulativa (detta curva di Lorenz).

Il diagramma di Pareto illustra graficamente la regola del “80:20” per cui l’80% degli effetti di un fenomeno è spiegato dal “20%” dei fattori coinvolti.



L'**analisi della creazione del valore** permette di identificare attività che aggiungono valore al cliente attraverso alcune domande chiave:

Questa attività potrebbe essere eliminata se alcune attività precedenti fossero fatte in maniera diversa o correttamente? **SI → Valore aggiunto**

**NO ↓**

Esiste una tecnologia che consente di eliminare questa attività? **SI → Valore aggiunto**

**NO ↓**

È possibile eliminare questa attività senza impattare la qualità del prodotto o del servizio? **SI → Valore aggiunto**

**NO ↓**

Questa attività è stata richiesta dal cliente? Sarebbe disposto a pagare per averla? **SI → Valore aggiunto**

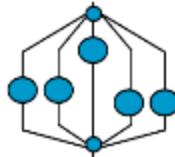
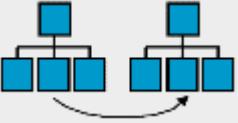
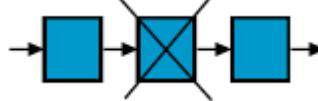
**NO ↓**

**Nessun valore aggiunto**

## Information Technology

Il mondo della tecnologia informatica si è evoluto diffondendo la presenza dei calcolatori praticamente in ogni ambito umano. Anche le organizzazioni modificano i propri processi in funzione delle nuove capacità computazionali ed espressive: la tecnologia informatica passa da avere un ruolo di supporto ad essere parte integrante/constituente dei processi che devono tenere conto delle sue capacità e non considerare l'informatizzazione come un “passo successivo” anche perché la rapidità dell’evoluzione informatica e la varietà di strumenti a disposizione permette di ottenere tecnologie professionali in maniera flessibile (bassi costi, cloud, ...) in contrasto con la tradizionale visione di un sistema informativo rigido e strutturato (sala server, centri elaborazione dati interni alle aziende, ...). Inoltre Internet ha cambiato le carte in gioco e molte soluzioni risultano essere più efficienti ed efficaci se distribuite geograficamente ove più opportuno.

In sintesi: l'**information technology deve essere interpretata come fattore di successo strategico per le prestazioni dei processi di business e le tecnologie devono essere sviluppate contestualmente al modello di impresa per massimizzarne i risultati**.

<b>Azienda estesa</b>	Il BPR consiste nell'estensione dei processi all'esterno dei confini dell'impresa	Internet, intranet, extranet LAN, WAN Electronic Data Interchange (EDI) E-Commerce ...	
<b>Parallelizzazione</b>	Il BPR consiste nel parallelizzare attività che prima erano svolte in sequenza	Workflow Groupware Cooperative processing Networking ...	
<b>Riallocazione</b>	Il BPR prevede di riallocare responsabilità, attività e specializzazioni	Database distribuiti ERP Architetture client/server Soluzioni web ...	
<b>Eliminazione</b>	Il BPR permette di eliminare attività a basso valore (eventualmente inserendone altre)	Automazione Integrazione applicazioni ...	
<b>Cicli di miglioramento continuo</b>	Il BPR attiva anelli di feedback per stimolare il miglioramento continuo	Data warehouse Decision Support System Business Intelligence System ...	

## Ridisegno dei processi: modello in 6 fasi

### 1. CONDIVISIONE OBIETTIVI AZIENDALI

Attività principali	Risultati
<ul style="list-style-type: none"> <li>• Documentazione vision aziendale: strategie e obiettivi</li> <li>• Rilevazione piani di sviluppo ed eventi chiave già pianificati</li> <li>• Rilevazione ruolo e aspettative dell'Azienda, clienti e soggetti coinvolti nel processo</li> </ul>	<ul style="list-style-type: none"> <li>✓ Quadro di riferimento di orientamenti strategici e obiettivi di sviluppo</li> <li>✓ Linee guida per lo sviluppo dell'attività</li> <li>✓ Modello di interrelazione volumi, costi, servizio</li> </ul>

### 2. DEFINIZIONE LINEE GUIDA

Attività principali	Risultati
<ul style="list-style-type: none"> <li>• Selezione processi di gestione</li> <li>• Identificazione skills necessarie e selezione risorse</li> <li>• Strutturazione team e pianificazione attività</li> <li>• Sviluppo e promozione programmi di comunicazione</li> </ul>	<ul style="list-style-type: none"> <li>✓ Quadro processi e priorità</li> <li>✓ Macro target</li> <li>✓ Risorse da coinvolgere e composizione team</li> <li>✓ Struttura operativa del programma</li> <li>✓ Approccio alla comunicazione con programmi e scopi</li> </ul>

### 3. ANALISI DI BASE E BENCHMARK

Attività principali	Risultati
<ul style="list-style-type: none"> <li>• Analisi processi attuali e sotto-processi chiave</li> <li>• Confronto con best practices</li> <li>• Analisi e definizione priorità per processi chiave e sotto-processi</li> <li>• Sviluppo processi interni ed esterni per la definizione dei bisogni e dei requisiti</li> <li>• Definizione priorità relative alle opportunità di miglioramento</li> </ul>	<ul style="list-style-type: none"> <li>✓ Mappatura dei processi</li> <li>✓ Diagrammi di flusso dei processi</li> <li>✓ Benchmark con concorrenti e standard</li> <li>✓ Rischi e opportunità (SWAT)</li> <li>✓ Output processi e posizionamento output su matrice costi/valore</li> <li>✓ Indagini e sessioni di analisi requisiti</li> <li>✓ Lista delle opportunità (andamento domanda, redditività, condivisione servizi)</li> </ul>

Early wins

### 4. SVILUPPO DEI CONCETTI INNOVATIVI

### 5. PROGETTAZIONE DEL CAMBIAMENTO

Attività principali	Risultati
<ul style="list-style-type: none"> <li>• Sviluppo vision per un processo eccellente</li> <li>• Disegno opzioni per nuovi processi</li> <li>• Definizione risorse e tecnologie necessarie</li> </ul>	<ul style="list-style-type: none"> <li>✓ Diagrammi di flusso e prioritizzazione cambiamenti</li> <li>✓ Obiettivi, modifiche, interfacce critiche</li> <li>✓ Richieste di risorse umane e investimenti</li> </ul>

Realizzazione del cambiamento

### 6. PIANIFICAZIONE DELL'IMPLEMENTAZIONE

Attività principali	Risultati
<ul style="list-style-type: none"> <li>• Preparazione e condivisione piano di implementazione</li> <li>• Condivisione variazioni alla struttura organizzativa post-implementazione</li> <li>• Valutazione architettura e piano di migrazione</li> <li>• Allineamento criteri di misurazione delle prestazioni</li> </ul>	<ul style="list-style-type: none"> <li>✓ Programmi</li> <li>✓ Strumenti di misurazione</li> </ul>

La **realizzazione del cambiamento** è un **processo continuo** costituito da una serie di **attività chiave multidimensionali o multidisciplinari** (stati di avanzamento frequenti, flessibilità di impiego delle risorse, incoraggiamento e supporto dei team, controllo e misurazione dei processi, valutazione di azioni di miglioramento, comunicazione, ...) che hanno l'**obiettivo di gestire il processo e monitorare, raggiungere e comunicare i risultati**.

<https://www.inteqgroup.com/blog/6-key-business-process-reengineering-steps>



## Business Process Reengineering: 6 Key Steps + Some Secret Sauce

James Proctor Date: March 21, 2019

### The Six Key Steps of Business Process Reengineering

1. **Define Business Processes.** Map the current state (work activities, workflows, roles and reporting relationships, supporting technology, business rules, etc.).
2. **Analyze Business Processes.** Identify gaps, root causes, strategic disconnects, etc. in the context of improving organizational effectiveness, operational efficiency and in achieving organizational strategic objectives.
3. **Identify and Analyze Improvement Opportunities.** Identify, analyze and validate opportunities to address the gaps and root causes identified during analysis. This step also includes identifying and validating improvement opportunities that are forward facing – often strategic transformational opportunities that are not tethered to current state process.
4. **Design Future State Processes.** Select the improvement opportunities identified above that have the most impact on organizational effectiveness, operational efficiency, and that will achieve organizational strategic objectives. Make sure to select opportunities for which the organization has the budget, time, talent, etc. to implement in the project timeframe. Create a forward-facing future-state map that comprehends the selected opportunities.
5. **Develop Future State Changes.** Frequently overlooked (and a key root cause in failed BPR initiatives), this is where the above opportunities are operationalized before implementation. New workflows and procedures need to be designed and communicated, new/enhanced functionality is developed and tested, etc. Changes and opportunities cannot be implemented until they are operationalized.
6. **Implement Future State Changes.** Classic implementation based on dependencies among changes/opportunities, change management, project management, performance monitoring, etc.

### Business Strategy and BPR – The Secret Sauce

The key point I want to make in this blog is the distinction between business-as-usual improvement opportunities and strategic improvement opportunities. The distinction is essential in selecting the opportunities to move forward into design, development and implementation.

Business-as-usual improvement opportunities are those opportunities that improve, but do not fundamentally change current processes. These are the things that you do (typically incremental changes) to improve operational efficiency and "lean" the process – i.e. reduce waste, errors, cycle time, etc.

To be clear, I am not diminishing the importance of business-as-usual opportunities. Business-as-usual opportunities are ongoing and have significant impact on the bottom line, but they do not move the needle with respect to enabling an organization to achieve strategic objectives.

Strategic improvement opportunities, on the other hand, enable organization to do new (as well as existing) things in new ways. Strategic opportunities enable and support an organization's strategic objectives – new product lines, new approaches to customer service and engagement, speed and quality of fulfillment, etc.

It's my experience (100% of the time) that in a BPR initiative we identify more high-value transformational opportunities for improving business processes (in addition to business-as-usual opportunities) than an organization has the time, budget and resources to implement in the short run. So, how do you shortlist from a group of high-value transformation opportunities? An essential part of shortlisting is to identify which opportunities move the needle the most toward enabling an organization to achieve strategic objectives. BPR opportunities, unless clearly tied to strategic objectives, are untethered, and therefore open to subjective and political shortlisting.

<https://blog.ext.hp.com/t5/BusinessBlog-en/6-steps-for-effective-business-process-redesign/ba-p/7795>

## 6 steps for effective business process redesign

14 December 2016 By Nikhil Kalanje

Business process redesign is all about working out how to improve existing processes. The best case scenario is reducing costs and improving productivity by changing and updating them.

An overhaul of this scale isn't a project to be taken lightly, so it's important to really think about what your aims and designed outcomes are, along with people being realistic about the potential disadvantages. It's human nature to resist change, so your employees might react negatively if you make sweeping changes to their workplace in the wrong way.

We've put together a guide on how to successfully manage your business process redesign:

1. **Set clear goals.** The whole point of the design is to achieve specific objectives. How you action them is important but only after you've established the outcome. This way you can ensure all tasks are working toward the same outcome. For example, if your aim is to create a more secure network then ensure that everything from password reset processes to how people print are considered.
2. **Identify every business process and prioritise them.** It can be a mammoth task to keep track of all of your processes but it's essential to keep a clear and up to date list of them. Without knowing everything you're working with then it's impossible to make genuine change within your organisation. Check out our list of '7 tools to organise your business process workflow' to make this process easier.
3. **Make data capture and processing a routine part of the work day.** If you need to capture data to inform what's important to your business or how best to change it, then do yourself a favour and make it a routine part of the work day. Don't create extra work for yourself when you can simply analyse what's already going on around you. ProcessFlows has a number of options for capturing data throughout the day, such as Optical Character Recognition, Intelligent Character Recognition and Voice Capture.
4. **One workflow.** Instead of separating new activities and changes into a separate stream, integrate them into the workflow. Developing and improving business processes is about streamlining and consolidating all business activities. This also applies to freelancers and contractors. In today's modern workplace it's likely that you'll be working with offices or freelance assets all over the world. Don't treat them as external resources; they should be counted the same as your regular, in-house employees. Start as you mean to go on with a single workflow.
5. **Empower the people who control processes.** This is a very simple one. Give the people who perform processes the power to make decisions regarding them. If there are three levels of approval for a simple, everyday process then ask yourself why. Strip away unnecessary red tape and create a single approval system where possible.
6. **Capture information once and at the source.** Instead of creating an additional process to capture information while a process is ongoing, change the initial process so that data capture is incorporated into it. It's an easy way to refine the amount of time, energy and input required. Not only that, giving employees more autonomy in their roles can lead to big changes, such as high morale, low staff turnover a boost in productivity. As reported by SAGE journals, respondents to a 2000 person student were nearly 2.5x more likely to take a job with autonomy.

Successful business process redesign projects are clear, concise and controlled. They pull in expertise from people all over the business to develop processes that are grounded in reality and not well-meaning but detached corporate theory. Most of it it's essential to get the idea of continuous learning in all employees at every level of the business.

<https://tallyfy.com/business-process-reengineering/>

## Business Process Reengineering (BPR): Definition, Steps, Examples

SONIA PEARSON

Proper execution of Business Process Reengineering can be a game-changer to any business. If properly handled, it can perform miracles on a failing or stagnating company, increasing the profits and driving growth. Business process reengineering, however, is not the easiest concept to grasp. It involves enforcing change in an organization – tearing down something people are used to and creating something new. And that's not an easy task.

### So, What is Business Process Reengineering?

Business process reengineering is the act of recreating a core business process with the goal of improving product output, quality, or reducing costs. Typically, it involves the analysis of company workflows, finding processes that are sub-par or inefficient, and figuring out ways to get rid of them or change them. Business process reengineering became popular in the business world in the 1990s, inspired by an article called Reengineering Work: Don't Automate, Obliterate which was published in the Harvard Business review by Michael Hammer. His position was that too many businesses were using new technologies to automate fundamentally ineffective processes, as opposed to creating something different, something that is built on new technologies. Think, using technology to "upgrade" a horse with lighter horseshoes which make them faster, as opposed to just building a car. In the decades since, BPR has continued to be used by businesses as an alternative to business process management (automating or reusing existing processes), which has largely superseded it in popularity. And with the pace of technological change faster than ever before, BPR is a lot more relevant than ever before.

### Business Process Reengineering Steps

As we've mentioned before, business process reengineering is no easy task. Unlike business process management or improvement, both of which focus on working with existing processes, BPR means changing the said processes fundamentally. This can be extremely time-consuming, expensive and risky. Unless you manage to carry out each of the steps successfully, your attempts at change might fail.

1. **Identity and Communicating the Need for Change.** If you're a small startup, this can be a piece of cake. You realize that your product has a high user drop-off rate, send off a text to your co-founder, and suggest a direction to pivot. For a corporation, however, it can be a lot harder. There will always be individuals who are happy with things as they are, both from the side of management and employees. The first might be afraid that it might be a sunk investment, the later for their job security. So, you'll need to convince them why making the change is essential for the company. If the company is not doing well, this shouldn't be too hard. In some cases, however, the issue is with the company not doing as well as it could be. Meaning, you should do your research. Which processes might not be working? Is your competition doing better than you in some regards? Worse? Once you have all the information, you'll need to come up with a very comprehensive plan, involving leaders from different departments. The management will have to play the role of salespeople: conveying the grand vision of change, showing how it'll affect even the lowest-ranked employee positively.

**Risk of Failure: Not Getting Buy-In From The Company.** If you fail to do this, however, your business process reengineering efforts might be destined to fail long before they even start. Business Process Re-Engineering can seriously impact on everyone in the company, and sometimes this can appear to be a negative change for some. Some employees might, for example, think you'll let them all go if you find a better way to function (which is a real possibility). In such cases, even if the management is on board, the initiative might fail because the employees aren't engaged. Usually, it's possible to get the employees buy-in by motivating them or showing them different views they weren't aware of. Sometimes, however, the lack of employee engagement might be because of a bad workplace culture – something that might need to be dealt with

before starting any BPR initiatives. Getting your employee to commit to change isn't easy. There are a bunch of change management models that help you accomplish this, though. Some of our favourites include the ADKAR Model and Bridge's Transition Model.

2. **Put Together a Team of Experts.** As with any other project, business process reengineering needs a team of highly skilled, motivated people who will carry out the needed steps. In most cases, the team consists of:

- a. Senior Manager. When it comes to making a major change, you need the supervision of someone who can call the shots. If a BPR team doesn't have someone from the senior management, they'll have to get in touch with them for every minor change.
- b. Operational Manager. As a given, you'll need someone who knows the ins-and-outs of the process – and that's where the operational manager comes in. They've worked with the process(es) and can contribute with their vast knowledge.
- c. Reengineering Experts. Finally, you'll need the right engineers. Reengineering processes might need expertise from a number of different fields, anything from IT to manufacturing. While it usually varies case by case, the right change might be anything – hardware, software, workflows, etc.

Risk of Failure: Not Putting The Right Team Together. There are a lot of different ways to mess this one up. If the team consists of individuals with a similar viewpoint and agenda, for example, they might not be able to properly diagnose the problems/solutions. Or, the team might involve too many or too few people. In the first case, the decision making might be slowed down due to conflicting viewpoints. In the later, there might not be enough experts in certain fields to create adequate solutions. It's hard to put all that down as a framework, as it depends on the project itself. There is one thing, however, that benefits every BPR team: having a team full of people who are enthusiastic (and yet unbiased), positive and passionate about making a difference.

3. **Find the Inefficient Processes and Define Key Performance Indicators (KPI).** Once you have the team ready and about to kick-off the initiative, you'll need to define the right KPIs. You don't want to adapt to a new process and THEN realize that you didn't keep some expenses in mind – the idea of BPR is to optimize, not the other way around. While KPIs usually vary depending on what process you're optimizing, the following can be very typical:

- a. Manufacturing
  - i. Cycle Time – The time spent from the beginning to the end of a process
  - ii. Changeover Time – Time needed to switch the line from making one product to the next
  - iii. Defect Rate – Percentage of products manufactured defective
  - iv. Inventory Turnover – How long it takes for the manufacturing line to turn inventory into products
  - v. Planned VS Emergency Maintenance – The ratio of the times planned maintenance and emergency maintenance happen

- b. IT
  - i. Mean Time to Repair – Average time needed to repair the system / software / app after an emergency
  - ii. Support Ticket Closure rate – Number of support tickets closed by the support team divided by the number opened
  - iii. Application Dev. – The time needed to fully develop a new application from scratch
  - iv. Cycle Time – The time needed to get the network back up after a security breach

Once you have the exact KPIs defined, you'll need to go after the individual processes. The easiest way to do this is to do business process mapping. While it can be hard to analyze processes as a concept, it's a lot easier if you have everything written down step by step. This is where the operational manager comes in handy – they make it marginally easier to define and analyze the processes. Usually, there are 2 ways to map out processes:

- a. Process Flowcharts – the most basic way to work with processes is through flowcharts. Grab a pen and paper and write down the processes step by step.
- b. Business Process Management Software – if you're more tech-savvy, using software for process analysis can make everything a lot easier. You can use Tallyfy, for example, to digitize your processes, set deadlines, etc. Simply using such software might end up optimizing the said processes as it allows for easier collaboration between the employees.

Want to get started with BPMS, but not sure how? Our guide to different BPM tools (and their distinct features) is as good of a start as any.

**Risk of Failure: Inability to Properly Analyze Processes.** Or, to put it more succinctly – impatience. It's uncommon for someone to try business process reengineering if their profits are soaring and the projections are looking great. BPR is usually called for when things aren't going all that well and businesses need drastic changes. So, it can be very tempting to hurry things up and skip through the analysis process and start carrying out the changes. The thing is, though, the business needs analysis needs to be done properly, not rushed through to get to the more exciting parts. There are always time and money pressures in the business world, and it's the responsibility of the senior management to resist the temptation and make sure the proper procedure is carried out. Problem areas need to be identified, key goals need to be set and business objectives need to be defined and this takes time. Ideally, each stage requires input from groups from around the business to ensure that a full picture is being formed, with feedback and ideas being taken into consideration from a diverse range of sources. The next step is to identify and prioritize the improvements that are needed and those areas and processes that need to be scrapped. Any business that doesn't take this analysis seriously will be going into those next steps blind and will find that their BPR efforts will fail. Any business that doesn't take this analysis seriously will be going into those next steps blind and will find that their BPR efforts will fail.

- 4. **Reengineer the processes and Compare KPIs.** Finally, once you're done with all the analysis and planning, you can start implementing the solutions and changes on a small scale. Once you get to this point, there's not much to add – what you have to do now is keep putting your theories into practice and seeing how the KPIs hold up. If the KPIs show that the new solution works better, you can start slowly scaling the solution, putting it into action within more and more company processes. If not, you go back to the drawing board and start chalking up new potential solutions.

## Business Process Reengineering Examples

The past decade has been very big on change. With new technology being developed at such a breakneck pace, a lot of companies started carrying out business process reengineering initiatives. There are a lot of both. There are a lot of both successful and catastrophic business process reengineering examples in history, one of the most famous being that of Ford.

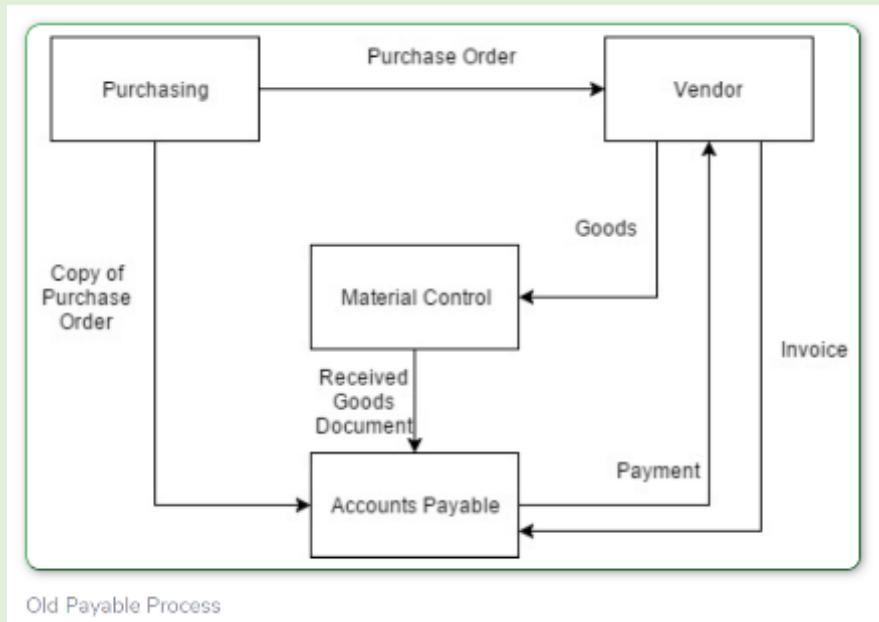
### BPR Examples: Ford Motors

One of the most referenced business process reengineering examples is the case of Ford, an automobile manufacturing company. In the 1980s, the American automobile industry was in a depression, and in an attempt to cut costs, Ford decided to scrutinize some of their departments in an attempt to find inefficient processes. One of their findings was that the accounts payable department was not as efficient as it could be: their accounts payable division consisted of 500 people, as opposed to Mazda's (their partner) 5. While Mazda was a smaller company, Ford estimated that their department was still 5 times bigger than it should have been. Accordingly, Ford management set themselves a quantifiable goal: to reduce the number of clerks working in accounts payable by a couple of hundred employees. Then, they launched a business process reengineering initiative to figure out why was the department so overstaffed. They analyzed the current system, and found out that it worked as follows:

1. When the purchasing department would write a purchase order, they sent a copy to accounts payable.

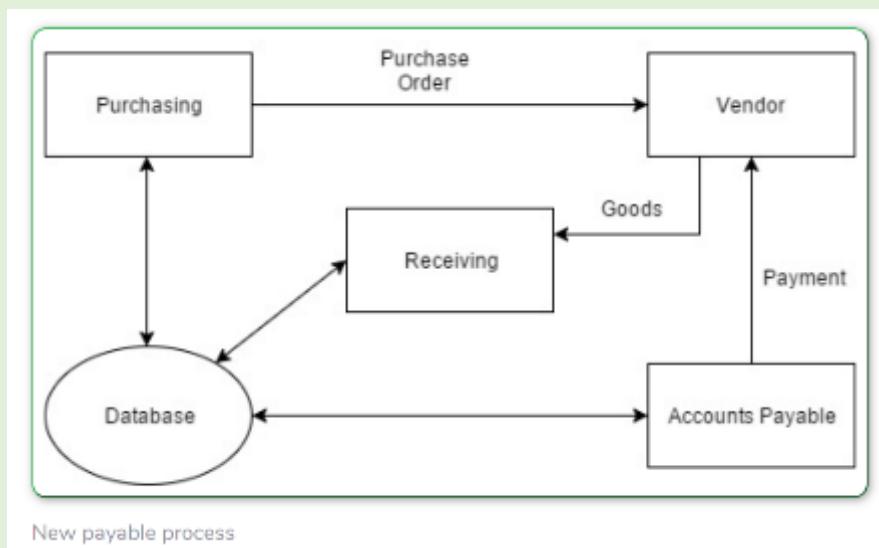
2. Then, the material control would receive the goods, and send a copy of the related document to accounts payable.
3. At the same time, the vendor would send a receipt for the goods to accounts payable.

Then, the clerk at the accounts payable department would have to match the three orders, and if they matched, he or she would issue the payment. This, of course, took a lot of manpower in the department.



So, as is the case with BPR, Ford completely recreated the process digitally.

1. Purchasing issues an order and inputs it into an online database.
2. Material control receives the goods and cross-references with the database to make sure it matches an order.
3. If there's a match, material control accepts the order on the computer.

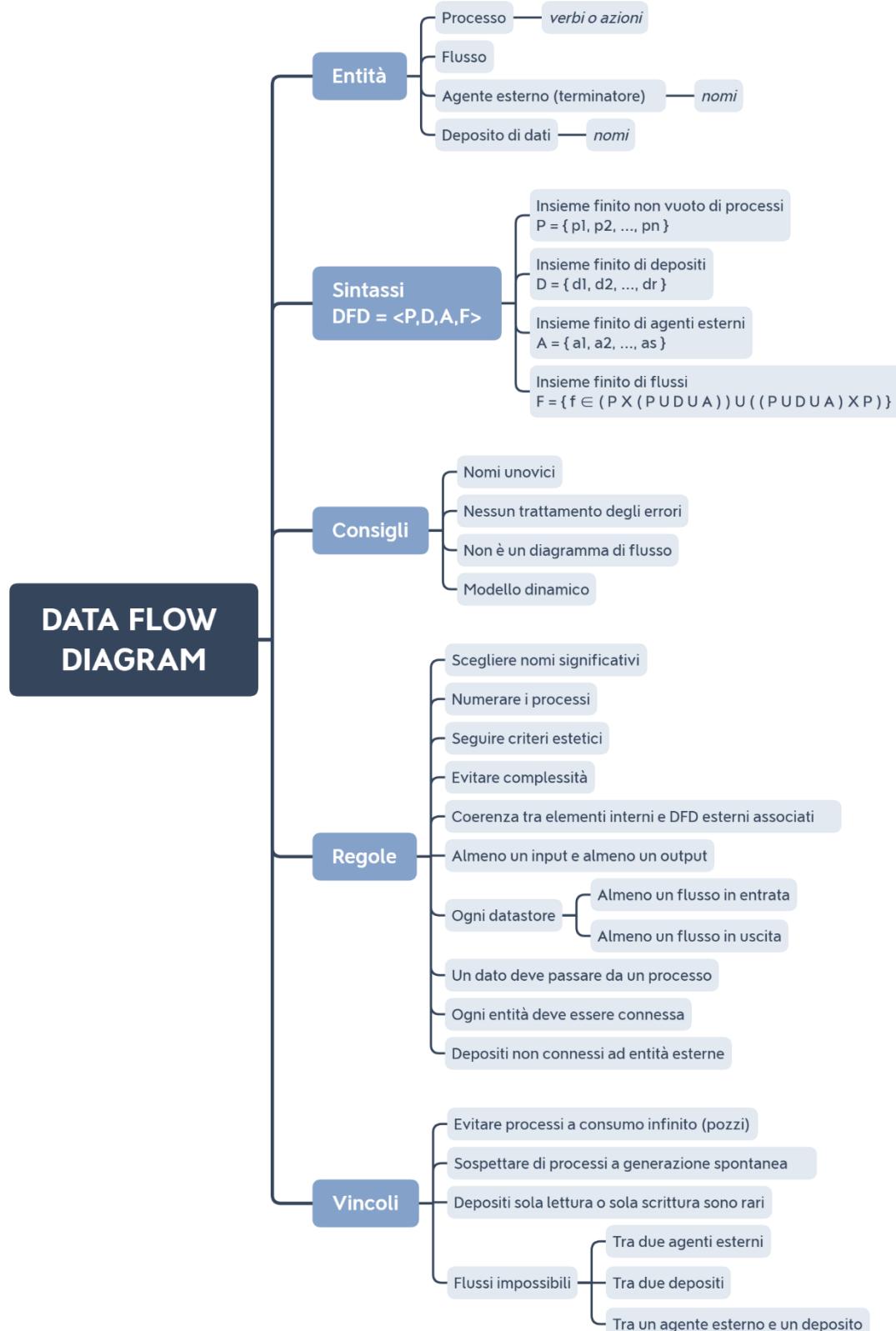


This way, the need for accounts payable clerks to match the orders was completely eliminated.

## Data Flow Diagram (DFD)

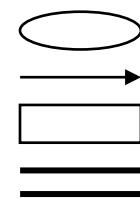
I Data Flow Diagrams hanno origini nella teoria dei grafi, vengono utilizzati fin da prima dell'avvento dei computer per la gestione delle informazioni. Utilizzati per la prima volta nell'ingegneria del software verso la metà degli anni '70.

Non esiste una definizione universale e sono presenti differenti formulazioni operative. Un DFD descrive una rete di processi funzionali interconnessi da depositi di dati. Pone enfasi sulle operazioni effettuate sulle informazioni e sulle dipendenze funzionali tra processi.



## Entità dei DFD

Entità	Ruolo	Pittogramma
<b>Processo (bolla)</b>	Trasforma dati	Ovale
<b>Flusso</b>	Muove dati	Freccia
<b>Agente esterno (terminatori)</b>	Producono e consumano dati	Rettangolo
<b>Deposito di dati</b>	Memorizza informazioni in modo passivo	Doppia linea



## Sintassi

Un DFD è una quadrupla  $\langle P, D, A, F \rangle$  in cui:

- $P = \{p_1, p_2, \dots, p_n\}$  è un insieme finito e non vuoto di processi
- $D = \{d_1, d_2, \dots, d_r\}$  è un insieme finito di depositi
- $A = \{a_1, a_2, \dots, a_s\}$  è un insieme finito di agenti esterni
- $F = \{f \in (P \times (P \cup D \cup A)) \cup ((P \cup D \cup A) \times P)\}$  è un insieme finito di flussi

## Rappresentazione

Un DFD è rappresentato tramite un grafo orientato nel quale:

- ogni nodo appartiene ad uno dei tre insiemi  $P$ ,  $D$  oppure  $A$
- ogni arco rappresenta un flusso di dati.

## Consigli

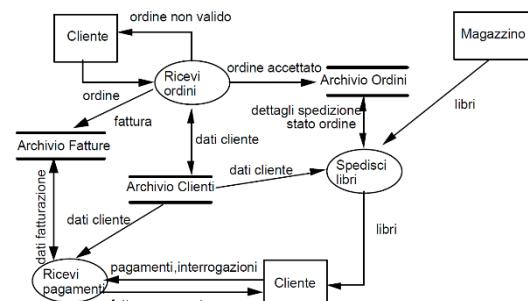
- Utilizzare nomi univoci per identificare le entità (eventualmente ripetendo le entità nella rappresentazione)
- Non rappresentare eccezioni, trattamento errori, delegandoli alle fasi finali dell'analisi
- Un DFD non è un diagramma di flusso, le frecce non indicano un ordinamento negli eventi
- Se si usa anche un modello dinamico (es. OMT), esprimere i flussi di controllo in un DFD comporta una duplicazione progettuale.

## Regole

1. Scegliere nomi significativi per processi (verbi, azioni), flussi, depositi (nomi) e agenti (nomi)
2. Numerare i processi
3. Disegnare i DFD seguendo criteri estetici
4. Evitare DFD estremamente complessi
5. Accertarsi della coerenza interna di un DFD, coerente con quelli ad esso associati
6. Ogni processo dovrebbe avere almeno un input e almeno un output
7. Ogni data store dovrebbe avere almeno un flusso in entrata e almeno un flusso in uscita
8. Un dato memorizzato nel sistema deve passare attraverso un processo
9. Ogni entità esterna deve essere connessa ad un processo
10. I depositi di dati non dovrebbero essere connessi alle entità esterne (che hanno accesso ai dati solo tramite i processi)

## Vincoli

- Evitare processi a consumo infinito (pozzi)
- Sospettare di processi a generazione spontanea
- I depositi a sola lettura o sola scrittura sono rari
- Non possono esistere flussi di dati tra:
  - ✓ Due agenti esterni
  - ✓ Due depositi
  - ✓ Un agente esterno e un deposito



## Avvertenze

In DFD realizzati superficialmente o con trascuratezza potrebbero essere omessi i nomi per flussi, processi, depositi.

I DFD non devono essere confusi con i flowchart.

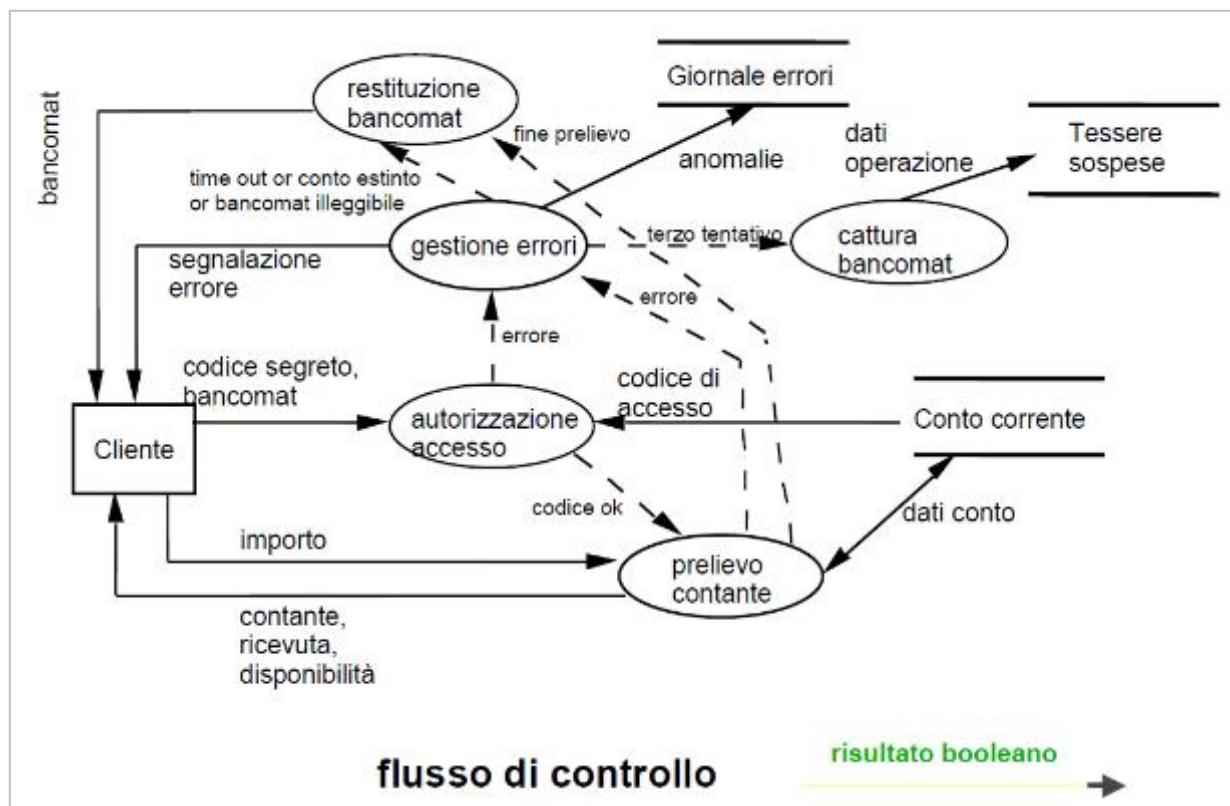
Nei DFD esiste una convenzione per cui un flusso da o verso un deposito può essere non etichettato quando i dati trasferiti corrispondono ad un intero oggetto (record).



Fare attenzione ai *buchi neri*: processi che hanno input ma non hanno output.

Fare attenzioni ai *miracoli*: processi che non hanno input ma hanno output.

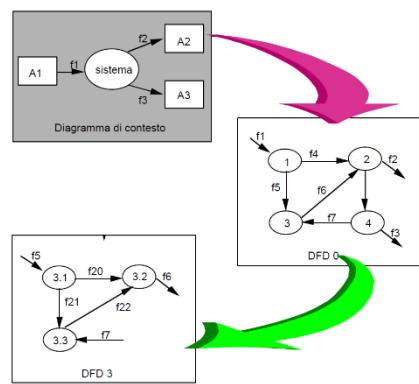
## DFD con flussi di controllo



## DFD multilivello

È possibile e suggerito utilizzare i DFD in maniera *gerarchica* o *a livelli* per garantire un adeguato livello di leggibilità e comprensione: un singolo DFD che modelli un ambiente reale in dettaglio avrebbe dimensioni e complicazioni tali da impedirne l'utilizzo. Solitamente un'applicazione di medie dimensioni richiede da 3 a 6 livelli.

Quando si stratifica un DFD in livelli, i DFD di dettaglio devono ovviamente avere tutti i flussi in entrata e fornire tutti i flussi in uscita descritti nel DFD di livello superiore.



## Consigli pratici

### 1. Evitare DFD molto sbilanciati

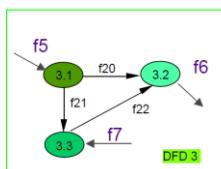
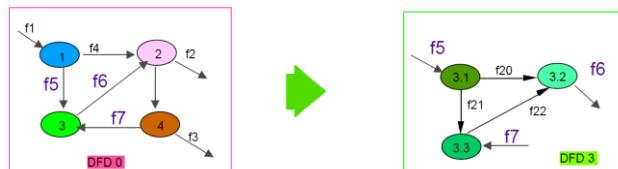
Un DFD che contiene processi atomici non dovrebbe contenere processi di livelli di astrazione più elevati

### 2. Porre attenzione al problema della presentazione del DFD

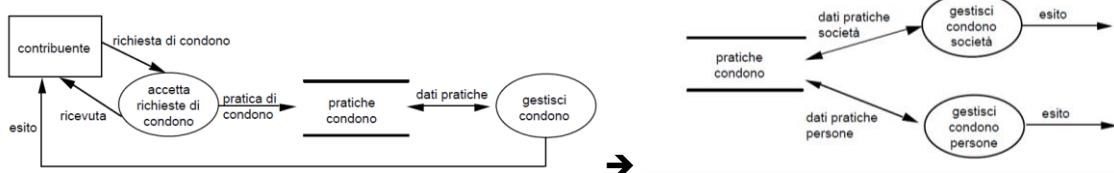
Affiancare documenti cartacei e strumenti di visualizzazione che permettano una navigazione tra le gerarchie

### 3. Verificare la coerenza tra i DFD

I flussi in entrata e in uscita di un DFD di dettaglio devono essere coerenti con il relativo DFD di livello superiore



4. Mostrare un deposito al livello di astrazione più alto, quando se ne scopre la necessità come interfaccia fra due o più processi, quindi riportare il deposito in ogni diagramma di livello inferiore che descrive quei processi



5. Verificare che un agente esterno collegato a un processo in un certo livello, compaia e resti connesso a un discendente di quel processo nel livello gerarchico inferiore.

#### Strategie per la costruzione di DFD

<b>Top-down</b>	Decomposizione di un processo in una serie di sottoprocessi chiaramente identificabili e indipendenti
<b>Bottom-up</b>	Costruzione di connessioni tra collezioni di concetti elementari
<b>Mixed</b>	Raffinamento di un DFD di massima in stadi successivi con utilizzo di tecniche top-down e bottom-up
<b>Outside-in</b>	Propagazione in avanti degli ingressi, a partire dalle interfacce con il sistema, evidenziando i processi coinvolti nei flussi di dati o, in alternativa, propagazione all'indietro delle uscite.

#### Notazione DFDEditor

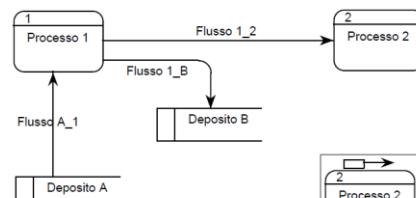


elementi a livello superiore a cui un processo invia dati

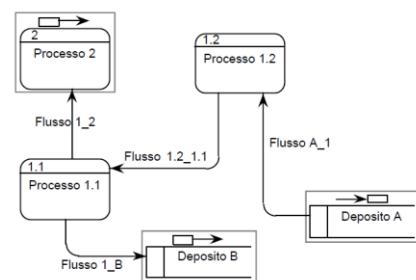


elementi a livello superiore da cui un processo riceve dati

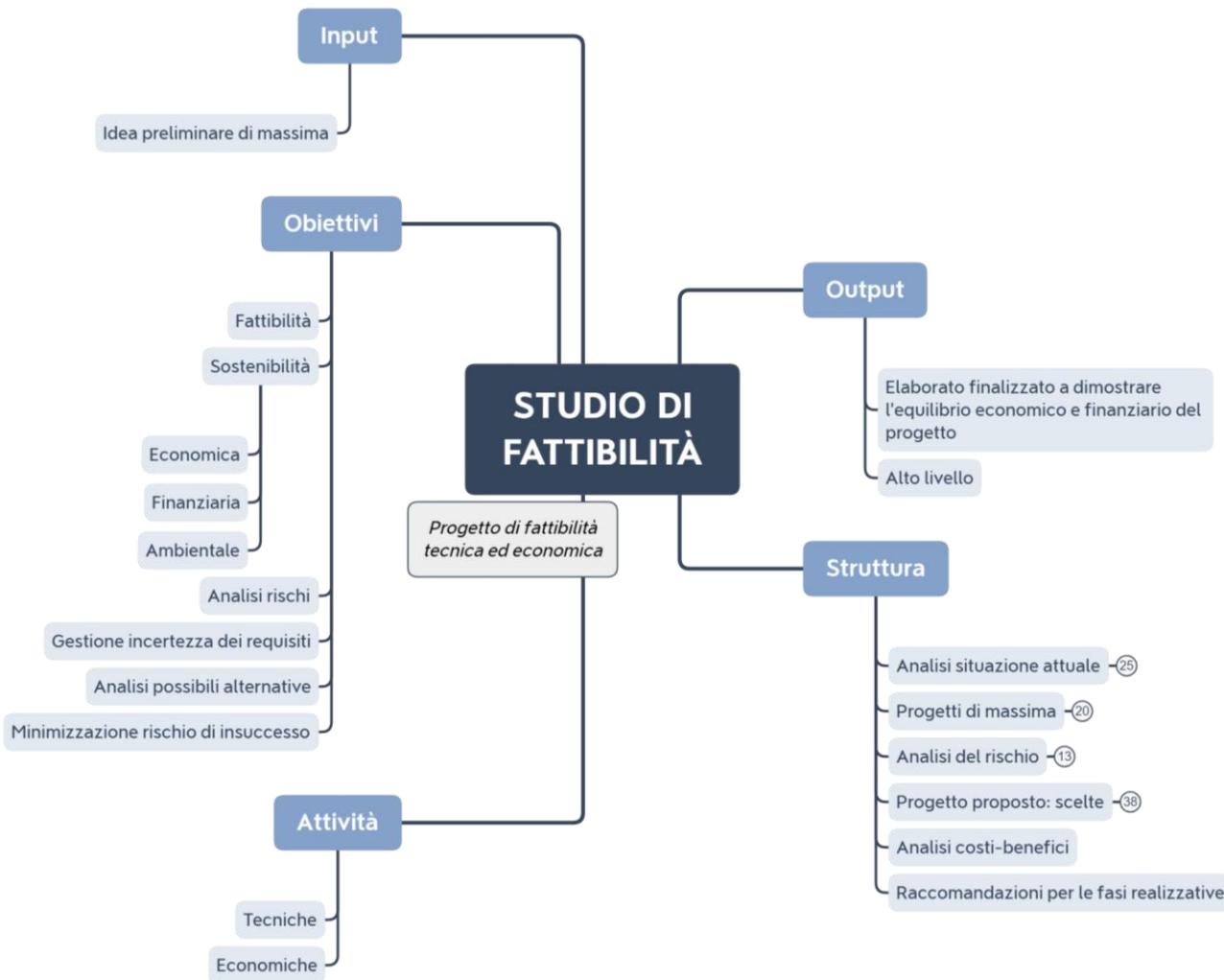
#### Un DFD



#### Esplosione del Processo 1



## Studio di fattibilità



Nell'ambito della progettazione lo studio di fattibilità (o progetto di fattibilità tecnica ed economica) consiste nell'analisi e nella valutazione sistematica delle caratteristiche, dei costi e dei possibili risultati di un progetto sulla base di una preliminare idea di massima. Comprende attività sia di natura tecnica che di natura economica (analisi costi-benefici) relative alla fattibilità e alla sostenibilità economico-finanziaria-ambientale dell'investimento (pubblico o privato). Lo studio di fattibilità si concretizza nella presentazione di un elaborato con grafici e prospetti finalizzato a dimostrare, come indicato dal Codice degli appalti, l'*equilibrio economico e finanziario* del progetto.

Gli studi di fattibilità devono essere realizzati ad elevati livelli di aggregazione, non tutte le componenti del Sistema Informativo devono essere analizzate; è necessario studiare solo le funzionalità principali e i casi d'uso più rilevanti. Lo studio può riguardare solo la parte più importante del Sistema Informativo, relativamente alle principali criticità.

### Motivazioni e obiettivi

È necessario effettuare uno studio di fattibilità per supportare e dare concretezza a progetti pre-identificati che, per fattori quali le dimensioni economiche, la complessità dell'intervento, l'incertezza sui requisiti, la presenza di possibili alternative, richiedono approfondimenti prima di poter avviare la realizzazione.

Lo studio di fattibilità mira a fornire tutti gli elementi necessari per l'avvio di una fase realizzativa minimizzando il rischio di insuccesso di un progetto.

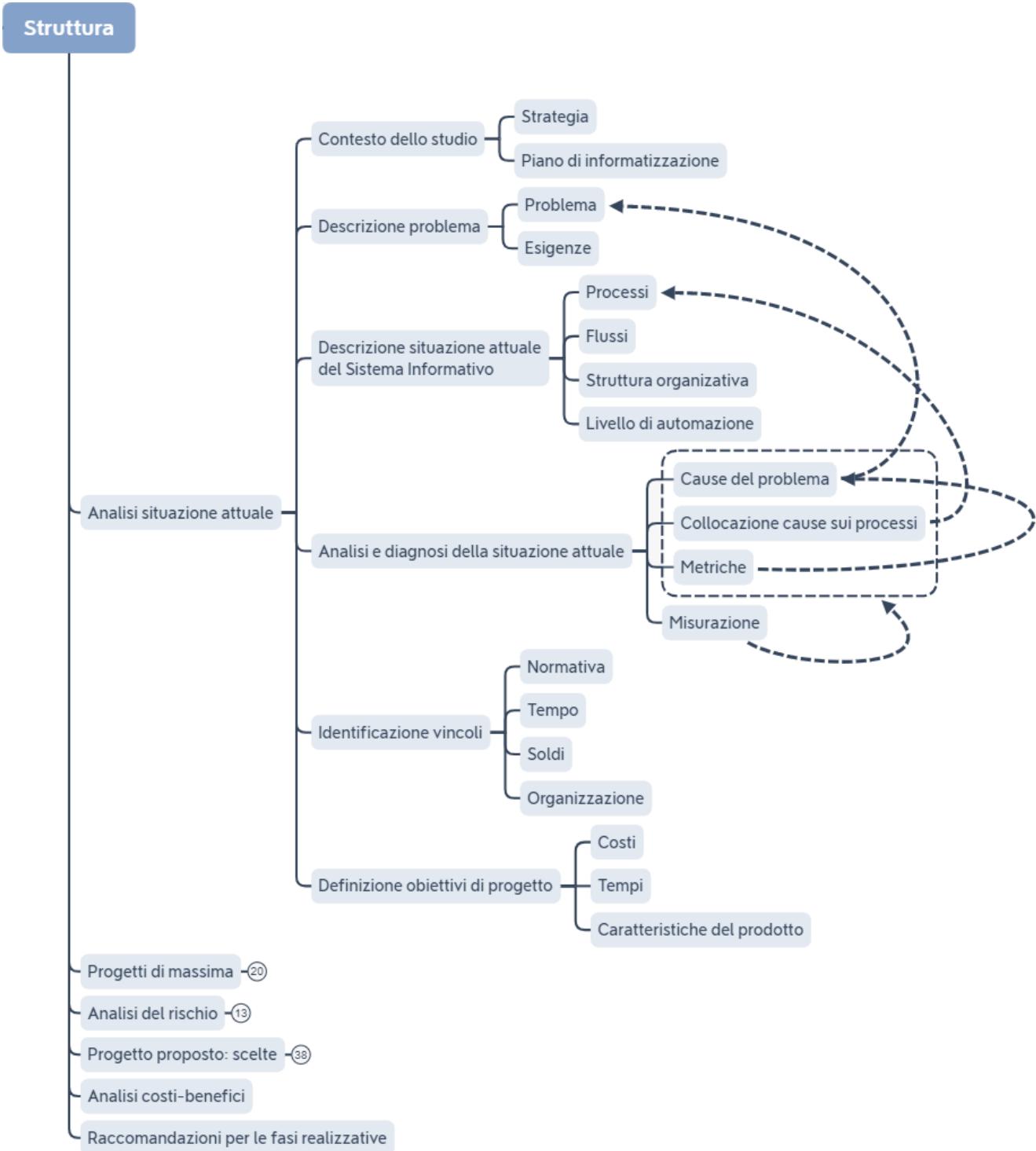
Lo studio di fattibilità presuppone la **valutazione di possibili alternative progettuali** che devono però essere delimitate dai margini decisionali di competenza dello studio stesso al fine di non stravolgere gli obiettivi.

## Struttura di riferimento

1. Analisi della situazione attuale
2. Progetti di massima
3. Analisi dei rischi
4. Progetto proposto
5. Analisi costi-benefici
6. Raccomandazioni per le fasi realizzative.



## Analisi situazione attuale



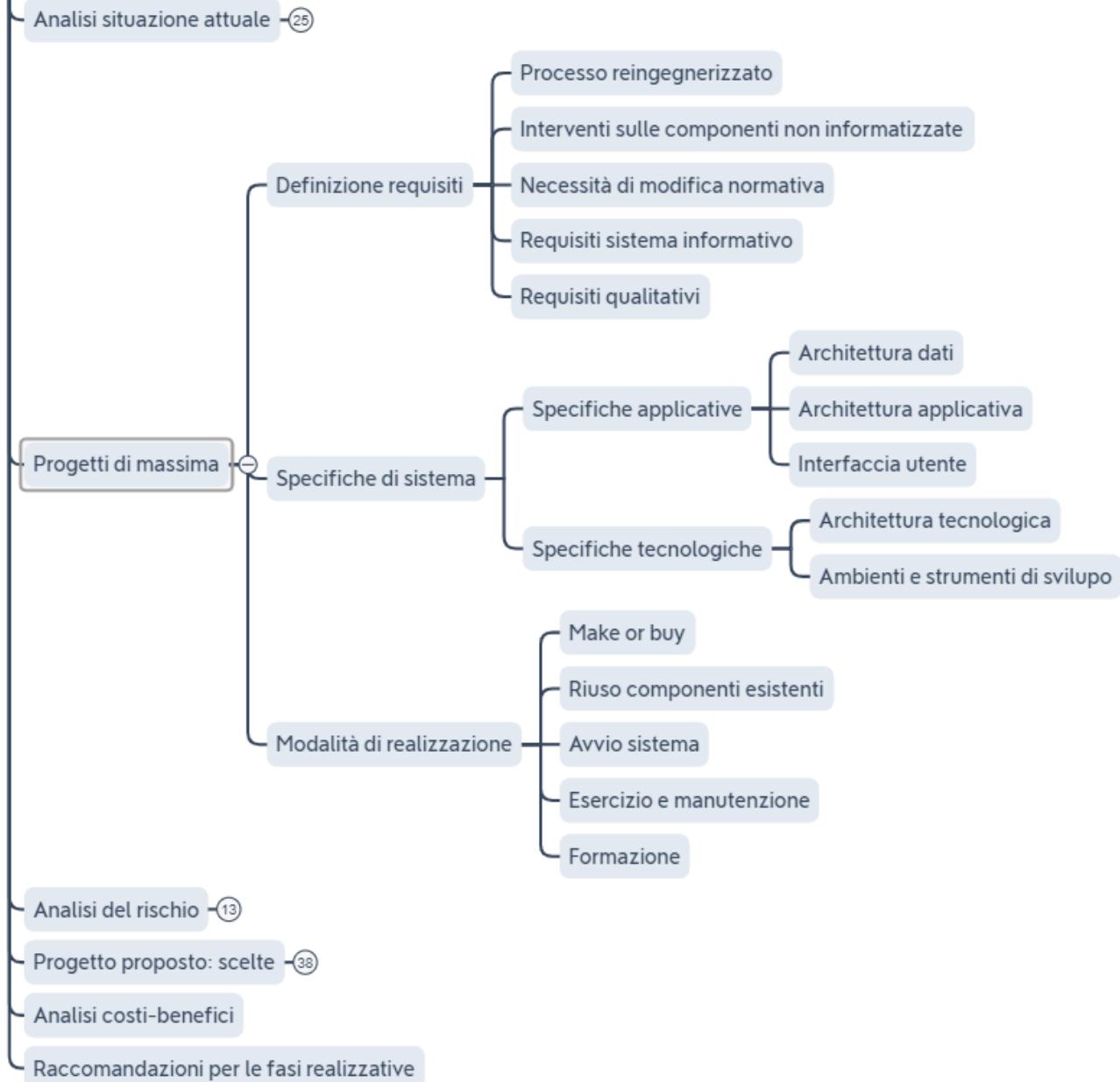
---

L’analisi della situazione attuale descrive le caratteristiche del sistema informativo sul quale deve essere realizzato il progetto utilizzando anche tutte le eventuali informazioni necessarie già presenti in azienda, l’analisi si articola in 6 punti:

- 1. Contesto dello studio:** inquadramento del progetto dal punto di vista della strategia di sviluppo del Sistema Informativo
  - Ripresa della visione strategica in termini di servizi, organizzazione e tecnologia
  - Ripresa dei principali passaggi che hanno portato all’individuazione del progetto
  - Collocazione del progetto all’interno del piano di informatizzazione
- 2. Descrizione della problematica:** presentazione dei problemi e delle opportunità che hanno dato origine al progetto, definizione delle priorità e dei confini
  - Descrizione e rilevanza della problematica
  - Esigenze da soddisfare (rispetto ad utenti interni ed esterni)
- 3. Descrizione della situazione attuale del Sistema Informativo:** individuazione e rappresentazione dei processi e dei sistemi informatici coinvolti nell’area di intervento
  - Individuazione e rappresentazione processi coinvolti
  - Individuazione e rappresentazione flussi informativi
  - Individuazione e rappresentazione struttura organizzativa e utenza coinvolta
  - Individuazione e rappresentazione attuale livello di automazione
- 4. Analisi e diagnosi della situazione attuale:** illustrazione dei risultati, finalizzati all’individuazione e quantificazione degli obiettivi di progetto, dell’attività di esame e valutazione critica dei processi impattati dal progetto
  - Individuazione dei fenomeni che costituiscono le cause del problema
  - Collocazione dei fenomeni sulle diverse componenti del processo di servizio
  - Individuazione di metriche atte a rappresentare fenomeni critici e loro evoluzione
  - Misurazione della situazione attuale
- 5. Identificazione dei vincoli:** identificazione e descrizione condizioni di ambiente non modificabili rispetto alla situazione attuale
  - Quadro normativo di riferimento
  - Vincoli temporali
  - Vincoli economici
  - Vincoli organizzativi
  - ...
- 6. Definizione degli obiettivi di progetto:** quantificazione degli obiettivi in termini di costi, tempi e specifiche caratteristiche del prodotto o servizio erogato, correlazione con le cause che hanno dato origine al progetto.

## Progetti di massima

### Struttura



I **Progetti di massima** sono descrizioni generali del sistema informativo, comprendono:

1. **Definizione dei requisiti** (condizioni che il sistema deve soddisfare)
  - Dettaglio del processo previsto dopo la reingegnerizzazione
  - Interventi previsti sulle componenti non informatizzate del processo
  - Necessità di modifica della normativa
  - Requisiti del sistema informativo da realizzare:
    - Informazioni trattate
    - Funzioni informatizzate
    - Modalità di lavoro
    - Requisiti architettonici

- Requisiti qualitativi

## 2. Specifiche di sistema (descrizione del sistema proposto in termini di proprietà)

- Specifiche applicative

- Architettura dei dati (con valutazione delle alternative)
- Architettura applicativa (con valutazione delle alternative)
- Interfaccia utente

- Specifiche tecnologiche

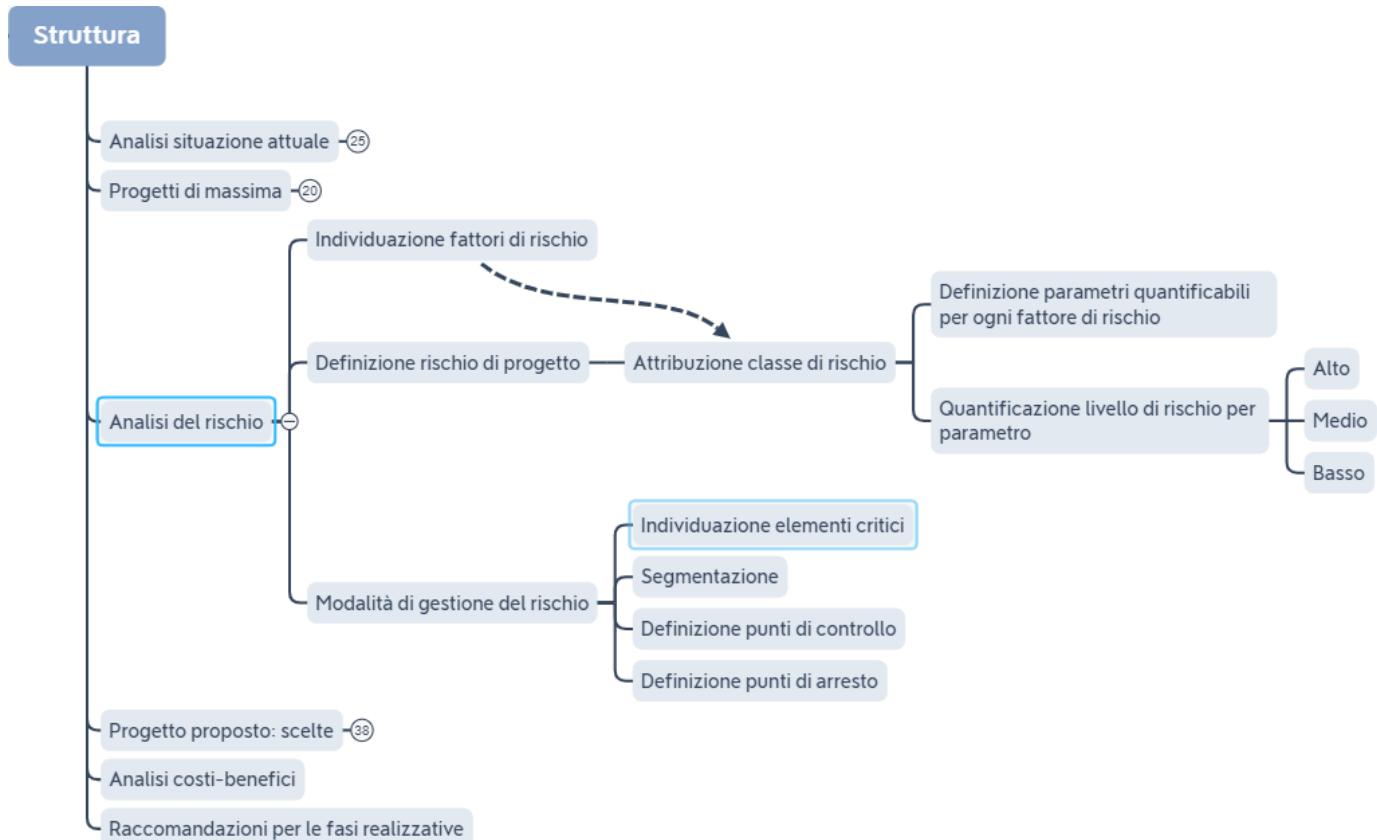
- Architettura tecnologica (con valutazione delle alternative)
- Ambiente e strumenti di sviluppo (con valutazione delle alternative)

## 3. Principali modalità di realizzazione

- Realizzazione o acquisizione (con valutazione delle alternative)
- Riuso di componenti esistenti (con valutazione delle alternative)
- Avvio del sistema
- Esercizio e manutenzione del sistema
- Formazione e assistenza utenti.

Possono essere realizzati e proposti diversi progetti di massima da valutare durante lo studio di fattibilità.

### Analisi del rischio



In un progetto esistono elementi o eventi in grado di pregiudicarne il buon esito in termini di mancata conclusione, difetti nei prodotti realizzati, lievitazione dei costi, allungamento dei tempi, difficoltà di integrazione con altre parti del Sistema Informativo. L'analisi del rischio si occupa di individuare e definire i fattori di rischio e studiare modalità per la loro gestione.

### 1. Individuazione dei fattori di rischio

In questa fase vengono identificati ed elencati i fattori di rischio.

## 2. Definizione del rischio di progetto

L'obiettivo di questa parte dell'analisi del rischio è di attribuire un'appropriata **classe di rischio** delle diverse fasi del progetto. L'attribuzione della classe di rischio prevede tre fasi:

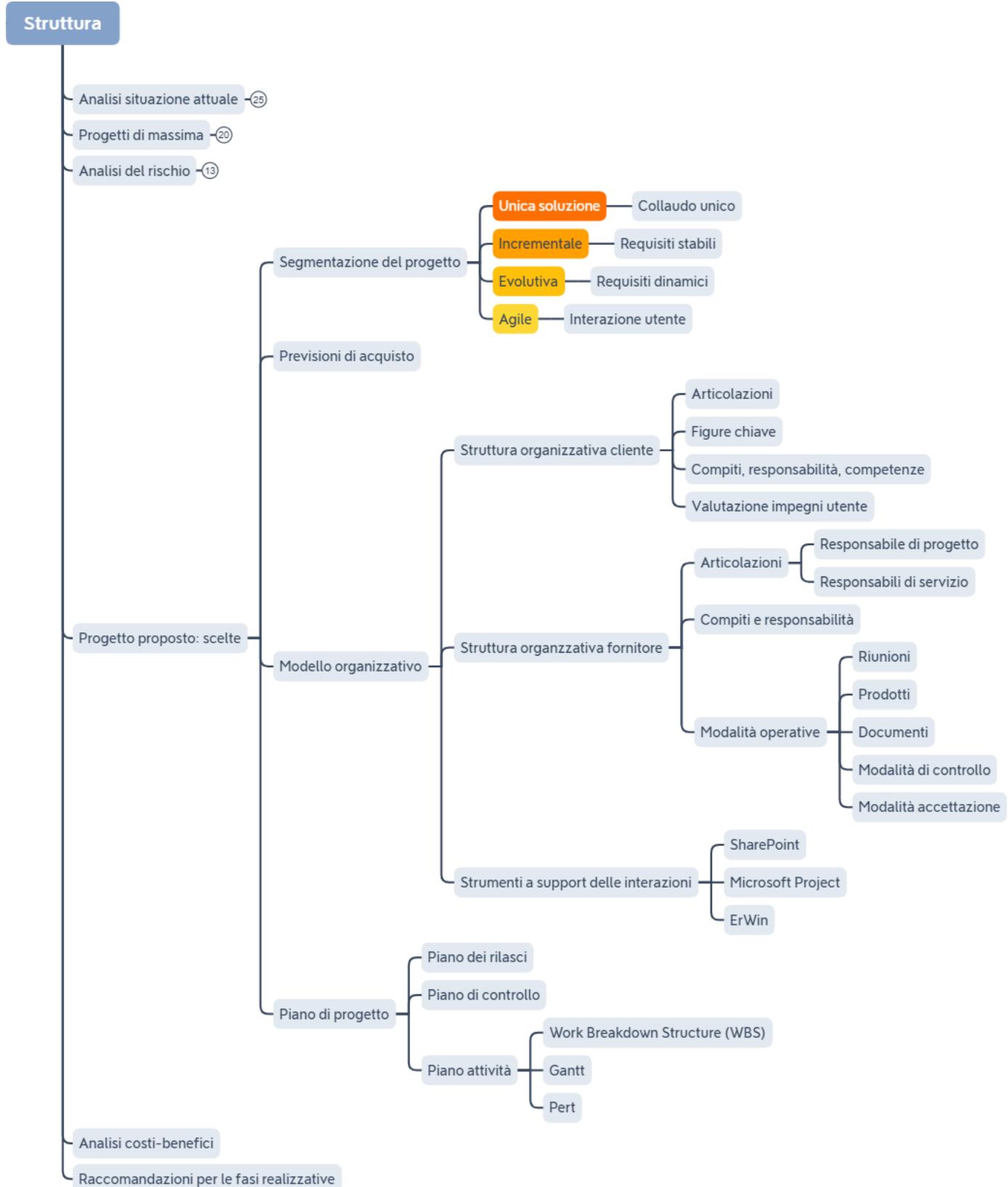
- 1) individuazione di un insieme di fattori di rischio
- 2) definizione di un insieme di parametri quantificabili per ogni fattore
- 3) quantificazione del livello di rischio per ogni parametro (basso/medio/alto).

## 3. Modalità di gestione del rischio

In questa sezione vengono identificati e descritti gli accorgimenti adottabili allo scopo di ridurre i rischi del progetto individuati nelle fasi precedenti:

- **individuazione degli elementi critici**, delle fasi a maggior rischio e quindi soggette a maggiore controllo
- **segmentazione** allo scopo di circoscrivere gli eventuali effetti determinati dal verificarsi di eventi correlati ai rischi inerenti la complessità, la dimensione, l'incertezza
- **definizione di punti di controllo** per la valutazione dello stato del progetto in momenti strategici (in corso d'opera)
- **definizione di punti di arresto** che possono determinare l'interruzione del progetto in caso si verifichino eventi legati a specifici fattori di rischio, allo scopo di evitare la diffusione di ulteriori danni all'azienda.

## Progetto proposto



In base a quanto indicato nel **progetto di massima**, tenendo conto dei risultati dell'**analisi del rischio**, si procede dettagliando le soluzioni scelte, i tempi e i modi di realizzazione.

## 1. Segmentazione del progetto

In funzione delle scelte realizzative fatte, il progetto viene suddiviso in moduli finalizzati alla realizzazione e al completamento di uno specifico sottosistema.

### Realizzazione in soluzione unica

- Implementazione e collaudo in un'unica versione
- attività continuativa

### Realizzazione incrementale

- Implementazione e collaudo in fasi successive
- Ogni fase contiene un sottoinsieme di funzionalità e servizi
- I requisiti sono definiti unicamente prima della realizzazione iniziale

### Realizzazione evolutiva

- Implementazione e collaudo in versioni successive
- Ogni versione contiene tutte o parti delle funzionalità e servizi
- I requisiti possono variare tra versioni successive a seguito di feedback

### Approccio agile

- Fortemente incrementale ed evolutivo
- Interazione con l'utente finalizzato ad ottenere feedback in corso d'opera
- Prevede significative variazioni di progetto e prototipi in corso d'opera

## 2. Riepilogo acquisizioni e realizzazioni previste

In base ai criteri di segmentazione scelti, si focalizza il progetto realizzativo e si sintetizzano le previsioni di acquisto.

## 3. Modello organizzativo

Definisce la struttura organizzativa necessaria per la conduzione e gestione del progetto, nonché le modalità di interazione fornitore-cliente.

### Struttura organizzativa preposta al progetto per il committente

- articolazione della struttura organizzativa
- figure chiave
- compiti, responsabilità e competenze
- valutazione di massima degli impegni richiesti agli utenti, al personale tecnico, ...

### Struttura organizzativa richiesta al fornitore

- articolazione della struttura organizzativa
  - ✓ individuazione del **responsabile di progetto**
  - ✓ eventuale identificazione dei vari responsabili di servizio
- definizione compiti e responsabilità
- modalità operative di interazione tra committente e fornitore (forme di interazione previste):
  - ✓ riunioni
  - ✓ prodotti
  - ✓ documenti
  - ✓ modalità di controllo
  - ✓ modalità di accettazione

### Eventuali strumenti e/o sistemi informatici a supporto delle interazioni

- Share Point
- Microsoft Project
- ErWin

#### 4. Piano di progetto

Il piano di progetto dettaglia le sequenze di attività e le loro dipendenze. È necessario che siano evidenziate le necessità e gli obiettivi per supportare l'adeguamento puntuale delle attività allo scopo di rispettare scadenze intermedie e finali. Il piano di progetto include:

- **Piano dei rilasci**

- ✓ Requisiti globali
- ✓ Specifiche realizzative di fondo
- ✓ Specifiche di dettaglio
- ✓ Produzione di prototipi
- ✓ Installazione finale del sistema

- **Punti di controllo**

- **Piano delle attività**

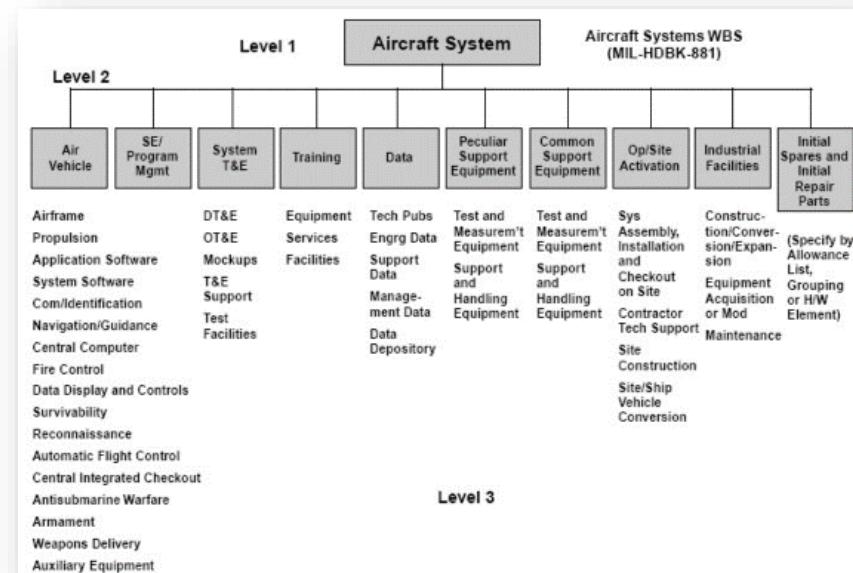
- ✓ Work Breakdown Structure (WBS)
- ✓ Diagramma di Gantt di massima
- ✓ Diagramma di Pert di massima

#### *Work Breakdown Structure (WBS)*

Principalmente tratto da Wikipedia.

Con l'espressione inglese **work breakdown structure (WBS)**, detta anche struttura di scomposizione del lavoro (traduzione letterale) o struttura analitica di progetto, si intende l'elenco di tutte le attività di un progetto. Le WBS sono usate nella pratica del project management e aiutano il project manager nell'organizzazione delle attività di cui è responsabile. Le WBS sono rappresentazioni del progetto in forma grafica e/o descrittiva, suddividono le attività in livelli consentendo un'analisi di dettaglio per la corretta identificazione delle attività elementari la cui esecuzione integrata conduce alla realizzazione dell'intero progetto.

Una WBS evidenzia il collegamento tra *deliverables* e attività pur non descrivendo in alcun modo lo sviluppo temporale del progetto. La WBS è un albero gerarchico orientato al prodotto (o deliverable) che viene suddiviso nel materiale, nel software, nei servizi, nei dati e nelle attrezzature che lo compongono. L'albero viene strutturato in base all'ingegneria di sistema che è sviluppata nella fase iniziale dell'apertura del progetto. La WBS definisce il prodotto, o i prodotti, da sviluppare o da produrre. Essa mette in relazione, con il prodotto finale e fra di loro, gli elementi di lavoro che sono necessari alla sua realizzazione. La WBS può articolarsi in un numero qualsiasi di livelli.



---

Molto spesso i progetti sono composti da migliaia di attività: per facilitare il lavoro di organizzazione delle varie attività esistono delle WBS-tipo che elencano tutte le possibili attività (generiche) per i progetti del rispettivo ambito. L'insieme delle attività può quindi essere confrontata con una checklist.

La WBS è utile per:

- non dimenticare parti del lavoro e di evitare duplicazioni
- fare chiarezza, ordine e dare trasparenza
- facilitare la comunicazione tra gli stakeholders
- definire in maniera chiare, omogenea, inequivocabile il lavoro da eseguire
- operare aggregazioni di dati elementari (tempi, costi, ricavi, ...) sulle diverse parti del progetto.

### **Principi di base della Work Breakdown Structure**

#### Regola del 100%

Uno dei più importanti principi alla base della WBS è noto come Regola del 100%. La Practice Standard for Work Breakdown Structures (Second Edition), edita dal Project Management Institute (PMI), definisce questa regola così:

«La regola del 100%... precisa che la WBS debba includere il 100% del lavoro definito dal progetto e includere **tutto** il necessario - interno, esterno e appaltato - alla realizzazione del progetto, inclusa la gestione del progetto stesso. La regola del 100% è una delle più importanti linee guida per lo sviluppo, la decomposizione e la valutazione della WBS. La regola si applica a tutti i livelli della gerarchia: **la somma del lavoro dei livelli "figli" deve essere uguale al 100% del lavoro rappresentato dal loro "padre"** e **la WBS non dovrebbe includere alcun lavoro al di fuori dai limiti del progetto, ovvero non può includere più del 100% del lavoro**. È importante ricordare che la regola del 100% si applica anche al livello di attività, il lavoro rappresentato dalle attività in ciascun pacchetto di lavoro deve dare, sommato, il 100% del lavoro necessario per completare il pacchetto».

#### Programmazione dei risultati, non delle azioni

Se il progettista della WBS tenta di comprendervi ogni dettaglio relativo alle azioni, probabilmente includerà troppe azioni, o troppo poche. Troppe azioni eccederanno il 100% dei limiti del nodo superiore, mentre troppo poche non arriveranno a quella percentuale. **Il modo migliore di seguire la regola del 100% è di definire gli elementi della WBS in termini di risultati.** Questo assicura che la WBS non dia raccomandazioni eccessive riguardo ai metodi, dando più spazio al pensiero creativo dei partecipanti al progetto. Per i nuovi progetti di sviluppo, la tecnica più comune per assicurare una WBS rivolta ai risultati è di utilizzare una *product breakdown structure (PBS)*. Eventualmente la WBS può risultare dal confronto di PBS e ABS, dove ABS è un *activity breakdown structure*. Progetti software feature driven possono utilizzare una tecnica simile, che impiega una *feature breakdown structure*. Quando un progetto fornisce servizi professionali, una tecnica di uso comune è di comprendere tutti i prodotti rilasciabili per creare una WBS orientata al prodotto. Le WBS che suddividono il lavoro in fasi di progetto (per esempio, fase di progetto preliminare, fase di progetto esecutivo) devono assicurare che le fasi siano chiaramente separate da qualcosa che definisca anche i criteri di delimitazione delle fasi stesse (per esempio, l'approvazione del progetto preliminare o del progetto esecutivo) che comunemente vengono denominati milestone.

#### Elementi reciprocamente esclusivi

In aggiunta alla regola del 100%, è **importante che non ci siano sovrapposizioni nella definizione dei limiti tra due elementi della WBS**. Tale ambiguità potrebbe infatti portare a raddoppiamenti di lavoro e fraintendimenti circa responsabilità e autorità. Allo stesso modo, la sovrapposizione causerebbe probabilmente confusione riguardo alla gestione delle spese di progetto. Se i nomi degli elementi sono ambigui, un dizionario interno alla WBS torna utile per chiarire le distinzioni tra di essi. Il dizionario descrive ogni componente con milestone, prodotti, attività, limiti e talvolta date, risorse, costi, qualità, ecc.

### Livello di dettaglio (granularità) ed elaborazione progressiva

Un quesito fondamentale da porsi durante la progettazione di ogni WBS è quando smettere di dividere il lavoro in elementi più piccoli. Se gli elementi terminali sono troppo ampi, infatti, potrebbe risultare impossibile tenere traccia in modo efficiente delle prestazioni progettuali, se al contrario gli elementi sono troppo piccoli e quindi troppo numerosi, diventerà difficile tenerne traccia, specialmente se il lavoro è pianificato in un futuro piuttosto distante. Un compromesso soddisfacente può essere trovato facendo ricorso alla tecnica dell'elaborazione progressiva, che permette ai dettagli della WBS di essere progressivamente ridefiniti prima che il lavoro inizi su ogni singolo elemento. Una forma di elaborazione progressiva, nei grossi progetti, è nota come rolling wave planning (pianificazione ad aggiornamento costante) che stabilisce una pianificazione su una base temporale regolare per l'elaborazione progressiva. Nella pratica, un limite effettivo della granularità della WBS può considerarsi raggiunto quando non è più possibile definire risultati di progetto pianificati, e i soli dettagli rimanenti sono azioni; a meno che queste azioni non possano essere ridefinite per conformarsi alla "regola del 100%", la WBS non dovrebbe essere ulteriormente suddivisa.

### Schema di codifica

È di uso piuttosto comune numerare gli elementi della WBS in ordine sequenziale per metterne in risalto la struttura gerarchica. Per esempio, 1.3.2 Ruota Posteriore identifica l'oggetto come elemento di livello 3, poiché ci sono tre numeri separati da un punto decimale. Inoltre, lo schema di codifica aiuta a riconoscere gli elementi del WBS in ogni contesto scritto.

<b>WBS livello 1</b>	<b>100</b>	<b>WBS livello 2</b>	<b>100</b>	<b>WBS livello 3</b>	<b>100</b>
1. Bicicletta	100	1.1. Telaio	15	1.1.1. Telaio base	7
				1.1.2. Manubrio	2
				1.1.3. Forcella	3
				1.1.4. Sedile	3
		1.2. Trasmissione	5	1.2.1. Trasmissione	5
		1.3. Ruote	30	1.3.1. Ruota anteriore	13
				1.3.2. Ruota posteriore	17
		1.4. Freni	5	1.4.1. Freni	5
		1.5. Cambio	5	1.5.1. Cambio	5
		1.6. Integrazione	35	1.6.1. Ideazione	3
				1.6.2. Progettazione	5
				1.6.3. Montaggio	10
				1.6.4. Collaudo	17
		1.7. Gestione progetto	5	1.7.1. Gestione progetto	5

La tabella mostra una tecnica di costruzione di una WBS che dimostra quantitativamente la regola del 100%. All'inizio del processo di pianificazione, il project manager ha assegnato 100 punti all'intero progetto, che consiste nella progettazione e realizzazione di una bicicletta personalizzata. Al livello 2 della WBS, i 100 punti totali sono suddivisi in sette elementi comprensivi. Il numero di punti allocato ad ognuno è frutto di un giudizio personale basato sullo sforzo richiesto: non è una stima del tempo occorrente. I tre elementi di dimensioni maggiori del livello 2 sono suddivisi al livello 3, e così via. **L'elemento terminale più grande del terzo livello rappresenta solamente il 17% del totale del progetto.** Questi grandi elementi possono essere ulteriormente suddivisi usando la tecnica di elaborazione progressiva descritta sopra.

Talvolta, lo schema di codifica WBS include un carattere di sottolineatura ("\_") al termine del nome per identificare gli elementi terminali. Si tratta di un sistema utile poiché le attività pianificate ("installare la camera d'aria e lo pneumatico", per esempio) saranno assegnate agli elementi terminali invece che ai nodi superiori.

È raccomandabile che il progetto sia iniziato con un software interattivo, per esempio un foglio elettronico, in grado di sommare automaticamente i punteggi assegnati. Un'altra pratica raccomandata è discutere la stima dei punteggi con gli altri membri del project team, questa tecnica collaborativa aumenta la conoscenza della definizione dei limiti

---

degli elementi e dei presupposti, e per una buona gestione del progetto è necessario raggiungere il consenso circa il livello di granularità.

### Errori e malintesi comuni

Una WBS non è una lista completa di lavori, bensì una classificazione degli scopi del progetto.

Una WBS non è una pianificazione del progetto e non è una lista in ordine cronologico. Infatti, perché una "toolbox" di pianificazione possa essere considerata completa, al processo iterativo di composizione/scomposizione (il ciclo top-down → bottom-up e viceversa) delle attività/produzioni (ABS, Activity Breakdown Structure / PBS Product Breakdown Structure) e a quello di assegnazione delle responsabilità (rispetto alla OBS, Organizational Breakdown Structure) devono seguire:

1. **l'allocazione delle risorse**, consistente nel calcolo dei fabbisogni tempificati relativi alle assegnazioni; esso determina, per ciascuna assegnazione: a) l'inizio, b) la fine, c) la produzione complessiva sulla base della produttività (normalmente espresse in ore/periodo) accertata, d) il costo complessivo e il costo periodico;
2. **la schedulazione delle attività** (o, meglio, delle assegnazioni) basata su quattro elementi, dei quali due sono necessari (il calendario e la durata di ciascuna attività/risorsa/assegnazione) e due sono eventuali (le relazioni di dipendenza e i vincoli temporali specifici).

È sconsigliabile e considerato controproducente pianificare un progetto (per esempio usando un project management software) prima di progettare una WBS appropriata, sarebbe simile al voler pianificare le attività di un cantiere edile prima ancora di completare il progetto dell'edificio. Senza concentrarsi sui risultati del progetto, è molto difficile seguire la regola del 100% a ogni livello della gerarchia della WBS. Non è possibile recuperare una WBS con definizioni improprie senza rifarla dal principio, per cui vale sempre la pena di completarla prima di iniziare qualunque altra pianificazione.

**Una WBS non è una gerarchia organizzativa.** Talvolta, i principianti commettono l'errore di creare una WBS fedele alla struttura organizzativa, mentre è piuttosto comune che la responsabilità sia assegnata a elementi organizzativi, una WBS che ne rispecchi la struttura non è descrittiva del progetto e non è orientata ai risultati. Allo scopo, si veda anche: matrice di assegnazione responsabilità.

**La capacità della memoria a breve termine non dovrebbe influenzare la dimensione della struttura ad albero di una WBS.** Alcune fonti di documentazione suggeriscono che ogni livello sia limitato a 5/9 elementi perché questo è il limite teorico della memoria umana. Tuttavia, è da ritenersi che questo consiglio sia un'applicazione errata delle teorie sui principi cognitivi (si veda: The Magical Number Seven, Plus or Minus Two), poiché gli elementi di una WBS non sono dati interconnessi casualmente. La documentazione definitiva riguardo alle WBS non contengono questa raccomandazione. È molto più importante costruire un raggruppamento logico di risultati pianificati che preoccuparsi dei limiti della memoria umana a breve termine.

**In questo senso può essere utile realizzare una mappa mentale prima della WBS: essendo uno strumento di creatività aiuta a sviscerare alcuni aspetti che potrebbero rimanere reconditi con un approccio esclusivamente logico-razionale.** La realizzazione potrebbe essere affidata al singolo come al gruppo di lavoro; in ogni caso è importante tenere a mente il principio secondo cui "la mappa non è il territorio", per cui... la mappa mentale potrà descrivere in modo molto parziale l'articolazione del discorso e avrà invece soprattutto lo scopo di focalizzare l'attenzione sulla preparazione delle attività. Si noti che, benché la WBS abbia uno scopo più denotativo, anche per essa vale il principio del dualismo mappa-territorio.

Al di là della progressiva elaborazione dei dettagli, gli aggiornamenti della WBS richiedono un controllo formale dei cambiamenti. Questo è un altro motivo per cui una WBS dovrebbe essere orientata alla produzione di risultati (deliverable oriented) e non alle prescrizioni di metodi: i metodi possono cambiare rapidamente, e lo fanno, ma le modifiche nei risultati pianificati richiedono un maggiore grado di formalità. Se i risultati e le azioni si confondono a vicenda, il controllo dei cambiamenti può essere troppo rigido per le azioni e troppo informale per i risultati.

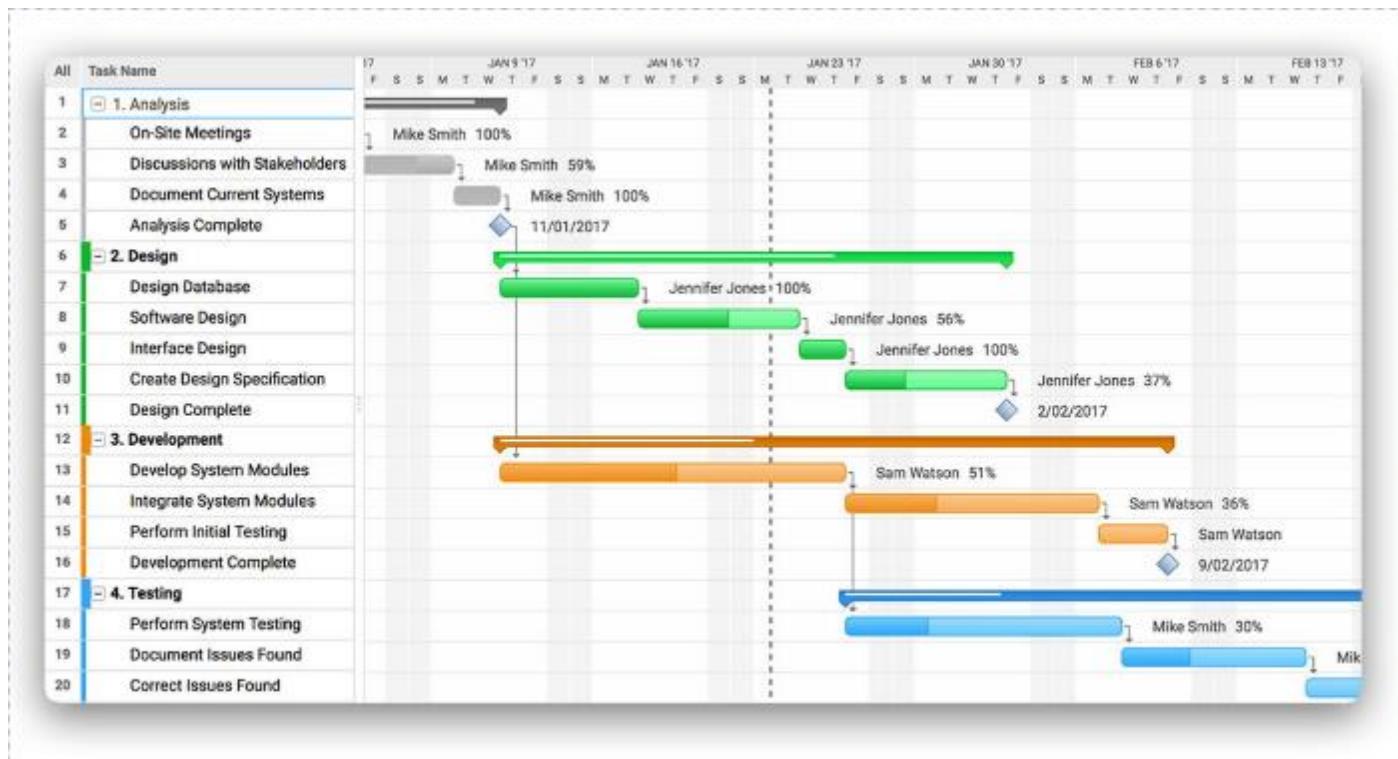
### Diagramma di Gantt

Il diagramma di Gantt è uno strumento di supporto alla gestione dei progetti, così chiamato in ricordo dell'ingegnere statunitense Henry Lawrence Gantt (1861-1919), che si occupava di scienze sociali e che lo ideò nel 1917.

Il diagramma consiste in una tabella nella quale le righe rappresentano le attività previste dalla WBS e le colonne definiscono i tempi necessari per la realizzazione e rappresentano l'arco temporale totale del progetto.

Barre orizzontali di lunghezza variabile rappresentano le sequenze, la durata e l'arco temporale di ogni attività. Queste possono sovrapporsi rappresentando attività parallele.

Durante l'esecuzione del progetto, delle barre secondarie o una differente colorazione mostrano lo stato di avanzamento delle singole attività.



ProjectManager.com's online Gantt chart software.

Il diagramma di Gantt è lo strumento appropriato ed ufficiale per gestire la pianificazione temporale di un progetto; in particolare esso permette di:

- pianificare le tempistiche delle attività di progetto
- verificare la fattibilità temporale del progetto
- permettere a tutti i soggetti coinvolti nel progetto di avere un quadro generale ed integrato delle date di inizio e fine delle attività di progetto
- ufficializzare strategicamente le date desiderate e/o imposte di inizio e fine progetto includendo eventuali milestone intermedie (Master Schedule)
- ufficializzare a livello operativo le date di inizio e fine attese di ogni attività elementare (Gantt di dettaglio)
- controllare, durante l'avanzamento del progetto, gli scostamenti rispetto alle previsioni (anticipo o ritardi)
- verificare continuamente le nuove stime a finire del progetto o di specifiche parti di esso a fronte dei consuntivi, per facilitare eventuali correzioni.

### Diagramma di PERT

Il diagramma di PERT (Program Evaluation and Review Technique) è un metodo statistico di determinazione dei tempi delle attività di progetto. Può essere applicato anche alla stima dei costi. Rispetto alla stima semplice, il metodo permette di determinare valori di stima **ottimale, probabile e pessimistica** (per questo motivo è anche detto stima a

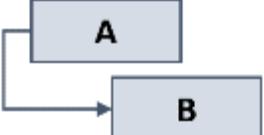
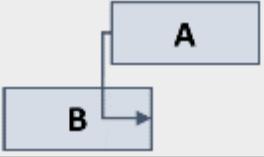
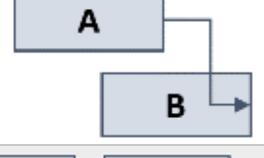
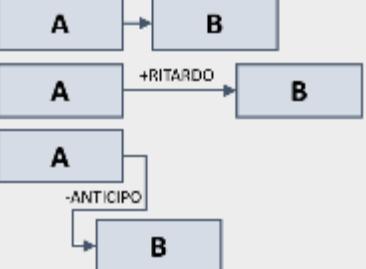
tre valori). Il metodo è stato sviluppato nel 1958 dalla Booz, Allen & Hamilton Inc. per l'ufficio Progetti della Marina degli Stati Uniti con l'obiettivo di ridurre i tempi e i costi per la progettazione e la costruzione dei sottomarini nucleari armati con missili Polaris, coordinando contemporaneamente diverse migliaia di fornitori e subappaltatori.

Il diagramma di PERT è una rappresentazione grafica del progetto che include:

- tutte le attività previste dalla WBS
- le milestones
- i condizionamenti reciproci temporali tra le attività (legami di precedenza).

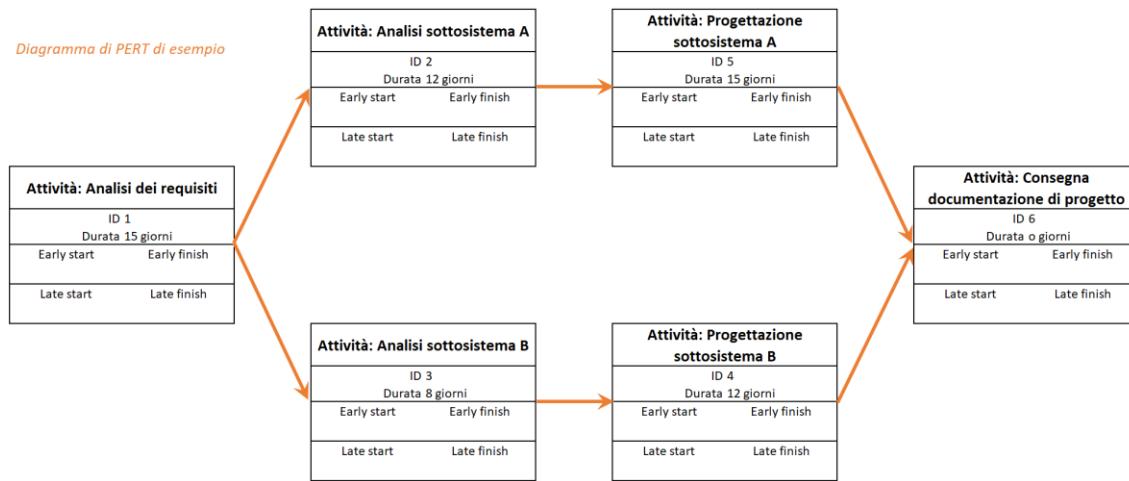
Il primo nodo del grafico rappresenta l'inizio del progetto, l'ultimo rappresenta la fine.

I legami di precedenza solitamente utilizzati sono:

<b>Finish to Start</b>	l'attività B non può iniziare se non è terminata l'attività A	
<b>Start to Start</b>	l'attività B non può iniziare se non è iniziata l'attività A	
<b>Start to Finish</b>	l'attività B non può finire se non è iniziata l'attività A	
<b>Finish to Finish</b>	l'attività B non può finire se non è finita l'attività A	
<b>Ritardi o anticipi</b>	Ad ogni legame di precedenza può essere associato un ritardo o un anticipo	

Conoscendo la durata prevista per ciascuna attività e le relative relazioni di precedenza è possibile definire la tempificazione del progetto e il relativo **percorso critico**: la catena di attività che condiziona la durata del progetto.

## Metodo per l'identificazione del percorso critico:

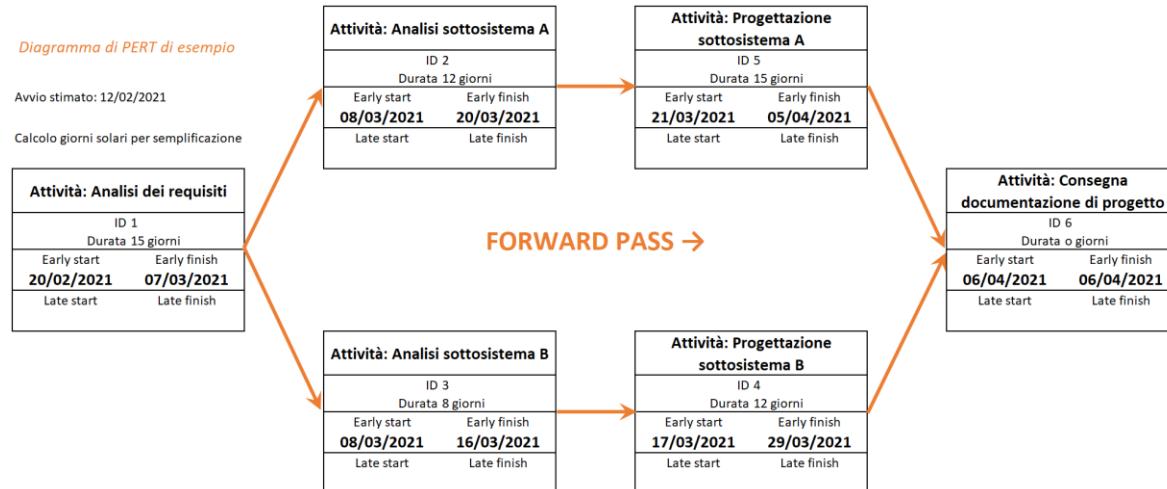


### 1. Forward pass (cammino in avanti)

Partendo dall'inizio del progetto si sommano le durate di ciascuna attività fino a determinare la data di fine progetto. Al termine di questa fase, per ogni attività, saranno determinate le date:

*Early start*: data di inizio al più presto

*Early finish*: data di fine al più presto

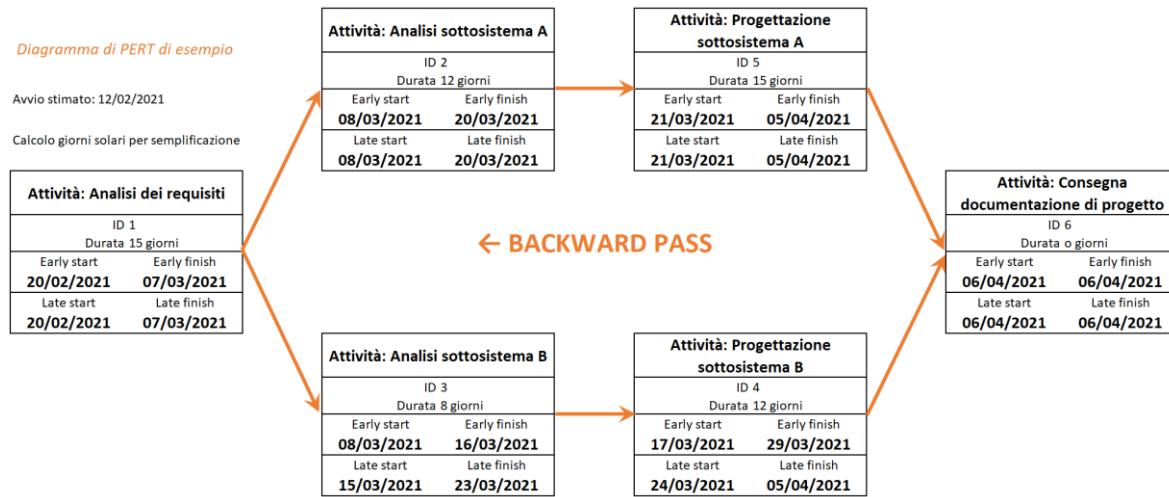


## 2. Backward pass (cammino all'indietro)

Partendo dalla fine del progetto si sottraggono le durate di ciascuna attività. Alla fine di questa fase, per ogni attività saranno determinate le date:

*Late start*: data di inizio al più tardi

*Late finish*: data di fine al più tardi

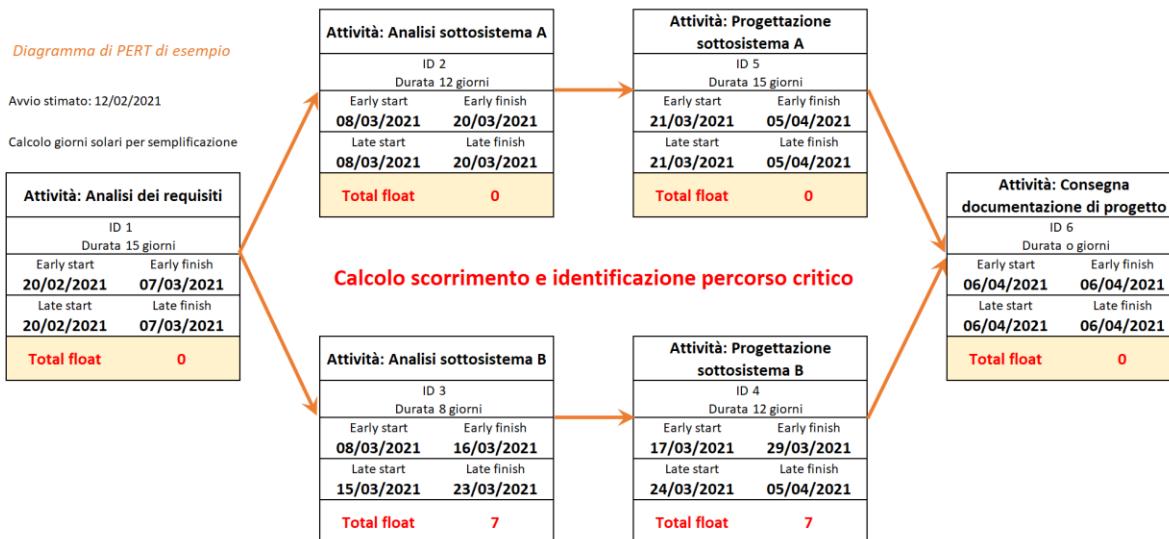


## 3. Calcolo dello scorrimento e identificazione del percorso critico

Una volta determinate le quattro date per ogni attività è possibile calcolare lo scorrimento (Total float) che indica il ritardo massimo che può avere la singola attività senza ritardare la data di completamento del progetto: Total float = Late Start – Early Start

Le attività con Total float uguale a 0 sono critiche.

La catena formata dalle attività critiche è detto **percorso critico**.



Le fasi tipiche di gestione di un PERT includono:

- **Pianificazione e costruzione del modello** (reticolo) di dettaglio sulla base della WBS e considerando la relativa sequenza logica
- **Stime dei tempi e analisi dei percorsi**: ad ogni attività viene attribuita una stima della durata prevista
- **Programmazione operativa**: sulla base dei risultati ottenuti si avvia la definizione delle risorse da impiegare (manodopera, carico impianti, ...).

La realizzazione di un PERT richiede, solitamente, molte revisioni del modello al fine di rientrare nei vincoli di costo e tempi previsti.

## Esercitazione sull'ottimizzazione della gestione dei progetti: il PERT

### Esercizio 1. Organizzazione di un concerto rock

I promotori di un evento rock devono svolgere le attività per organizzare un concerto. Si disegni la rete delle attività del progetto e si calcoli il minimo numero di giorni necessari per organizzare un concerto, evidenziando le attività critiche.

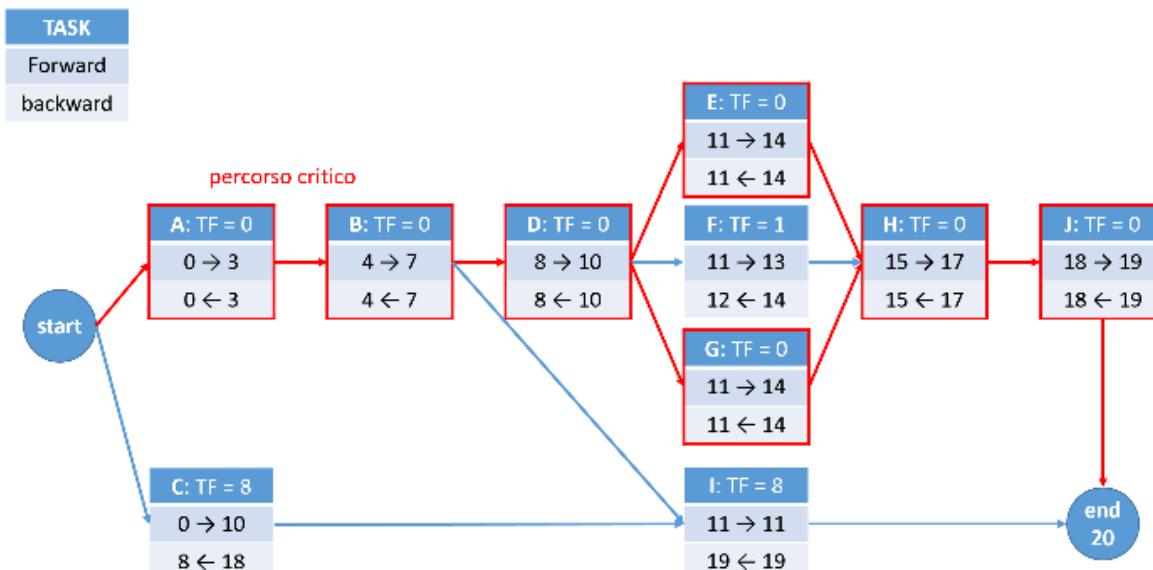
Descrizione	Indice	Predecessori	Durata (gg/uomo)
Individuazione del sito	A	-	3
Assunzione personale	B	A	2
Stampa volantini	C	B	5
Dettagli ultimo minuto	D	G	2
Contatti con agenti di vendita	E	A	3
Affitto apparecchiature	F	A	3
Prove generali	G	H,I	2
Preparazione dei trasporti	H	B	1
Installazione impianto acustico	I	B,F	4
Pubblicità radio e TV locali	J	B	2

### Esercizio 2. Costruzione di una stanza

Ad una ditta edile è stata commissionata la costruzione di una nuova stanza in un edificio esistente. Le attività per portare a termine il progetto sono rappresentate in tabella.

Descrizione	Indice	Predecessori	Durata (gg/uomo)
Scavo delle fondamenta	A	-	4
Costruzione dei pilastri portanti	B	A	4
Ordine e consegna delle finestre	C	-	11
Costruzione delle mura esterne	D	B	3
Posa dell'impianto elettrico	E	D	4
Posatura delle tubature	F	D	3
Montaggio delle grondaie	G	D	4
Posa delle pareti interne	H	E,F,G	3
Posa delle finestre	I	B,C	1
Pittura e rifinitura	J	H	2

Sapendo che il committente vuole che i lavori siano terminati entro quindici giorni lavorativi, il proprietario della ditta edile deve sapere quanti giorni saranno necessari per portare a termine il lavoro. Si calcoli la durata del progetto e le sue attività critiche. Inoltre si calcoli per ogni attività il total float.



Il proprietario della ditta edile sa che terminerà la costruzione della stanza dopo 20 giorni lavorativi. Sfortunatamente i lavori devono terminare entro 15 giorni lavorativi, quindi è necessario riuscire a ridurre la durata dei lavori per rispettare le scadenze. Decide quindi di aumentare il numero di manovali impiegati nella costruzione aumentando di conseguenza il numero di giorni/uomo a disposizione del progetto. Il proprietario riesce a stimare come da tabella i costi aggiuntivi giornalieri, e la massima riduzione possibile per ogni attività. Per esempio, è possibile ridurre l'attività A da 4 a 3 giorni con un costo aggiuntivo di 150 euro, oppure da 4 di 2 giorni pagando 300 euro. In che modo è possibile rispettare la scadenza minimizzando i costi aggiuntivi? Si rappresenti l'andamento dei costi aggiuntivi alla diminuzione della durata del progetto.

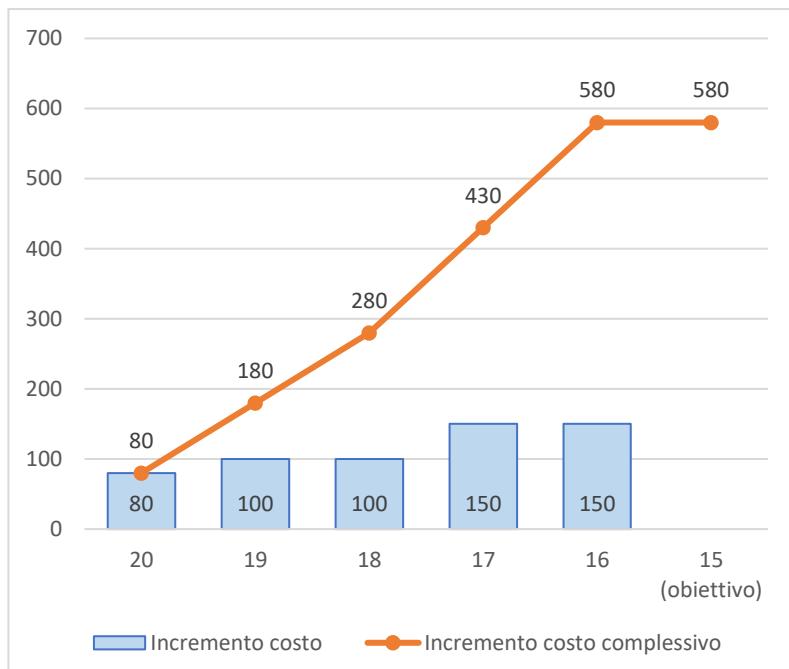
Descrizione	Indice	Percorso critico	Costo giornaliero	Durata (gg/uomo)	Massima riduzione (gg/uu)
<b>Scavo delle fondamenta</b>	A	Si	150	4	2
<b>Costruzione dei pilastri portanti</b>	B	Si	160	4	2
<b>Ordine e consegna delle finestre</b>	C		80	11	4
<b>Costruzione delle mura esterne</b>	D	Si	80	3	1
<b>Posa dell'impianto elettrico</b>	E	Si	160	4	2
<b>Posatura delle tubature</b>	F		150	3	1
<b>Montaggio delle grondaie</b>	G	Si	130	4	2
<b>Posa delle pareti interne</b>	H	Si	100	3	1
<b>Posa delle finestre</b>	I		70	1	0,5
<b>Pittura e rifinitura</b>	J	Si	100	2	1

Si considerano le attività lungo il percorso critico: A, B, D, E, G, H, J.

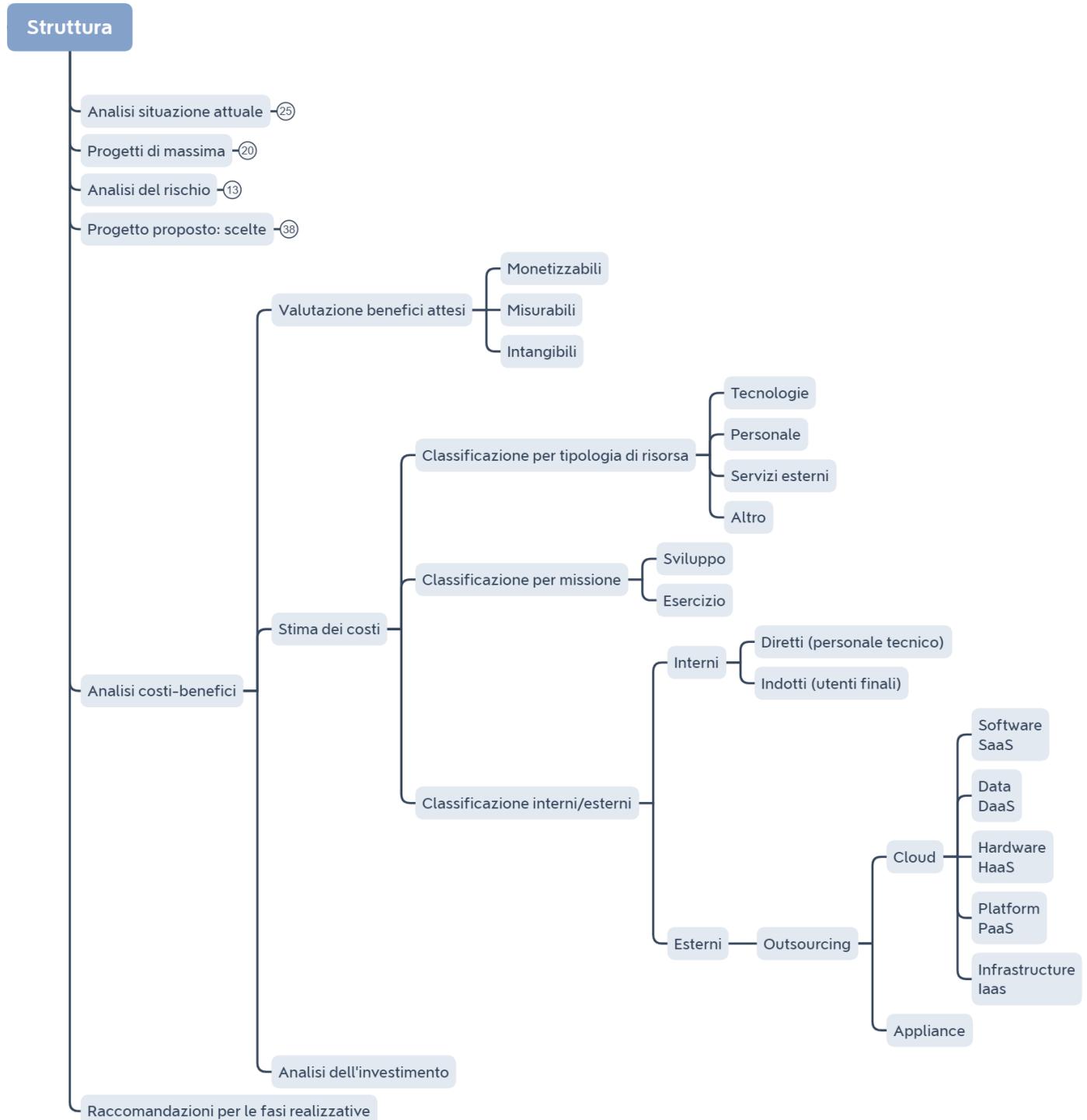
Si sceglie di “spendere” nelle attività per le quali la durata può essere ridotta a fronte di spese minori, pertanto:

1. D: 80 (riduzione massima 1)
2. H: 100 (riduzione massima 1)
3. J: 100 (riduzione massima 1)
4. A: 150 x 2 (riduzione massima 2)

Nota: non si è scelto di spendere per ridurre i tempi dell'attività G (più economica di A) in quanto è eseguita parallelamente a E: per ridurre i tempi sarebbe stato in tal caso necessario spendere  $130 + 160$  per ogni giorno (complessivamente più dispendioso di 150 di A).



## Analisi costi-benefici



## ANALISI COSTI-BENEFICI

Valutazione dei benefici attesi	Stima dei costi	Analisi dell'investimento
---------------------------------	-----------------	---------------------------

La **valutazione dei benefici attesi** fornisce una valutazione limitatamente ai benefici monetizzabili o, comunque, misurabili secondo criteri oggettivi, prevede l'*individuazione e descrizioni dei benefici attesi, l'individuazione ed esplicitazione delle metriche e la correlazione obiettivi-benefici.*

La **stima dei costi** si occupa di individuare le principali voci di costo e le relative metriche utilizzate e di stimare i costi di impianto e di esercizio.

L'**analisi dell'investimento** fornisce una valutazione comparata di costi e benefici, secondo le varie alternative possibili e considerando i tempi di progetto e i relativi tempi di manifestazione di costi e di restituzione di benefici; comprende le giustificazioni economiche dell'investimento e gli elementi per la scelta in caso di più alternative.

### Stima dei costi

I costi legati ad un Sistema Informativo possono essere classificati secondo differenti criteri.

Per tipologia di risorsa	Per missione	Interni / Esterni
<ul style="list-style-type: none"> <li>Costi delle tecnologie</li> <li>Costi del personale dedicato allo sviluppo, alla gestione e manutenzione delle applicazioni, all'assistenza agli utenti e all'amministrazione)</li> <li>Costi dei servizi esterni</li> <li>Altri costi</li> </ul>	<ul style="list-style-type: none"> <li>Costi di sviluppo Costi di progettazione e realizzazione, costi di avviamento</li> <li>Costi di esercizio Necessari per il corretto funzionamento dei sistemi e l'utilizzo delle applicazioni da parte degli utenti</li> </ul> <p><i>I costi di esercizio rappresentano il 70%-80% di tutti i costi informatici di un'azienda (fonte: Gartner Group)</i></p>	<ul style="list-style-type: none"> <li>Costi Interni           <ul style="list-style-type: none"> <li>Costi diretti: personale tecnico impegnato nello sviluppo e nell'esercizio del sistema</li> <li>Costi indotti: legati alle attività svolte dagli utenti finali per attività connesse al progetto (avvio, formazione, ...)</li> </ul> </li> <li>Costi esterni Relativi all'acquisizione di hardware, software e servizi affidati a società esterne</li> </ul>

### Outsourcing

Con il termine **outsourcing** si definisce la modalità tramite la quale un'organizzazione affida ad un soggetto (fornitore) esterno la responsabilità di una o più funzioni o di servizi aziendali (precedentemente svolti internamente) attraverso un accordo contrattuale.

Gli obiettivi o le motivazioni che spingono un'organizzazione ad affidare servizi in outsourcing possono essere:

- Riduzione dei costi di gestione: l'outsourcer ottiene profitto grazie alle sue competenze specialistiche e grazie ad economie di scala
- Benefici finanziari: si affidano servizi ad enti esterni per evitare investimenti
- Maggiore qualità del servizio
- Accesso a tecnologie avanzate
- Scalabilità in funzione della domanda.

Ogni componente di un sistema informativo può essere gestita tramite outsourcing:

- CED
- Server
- Personal computer
- Help desk
- Siti web
- ...

Attenzione: per non rischiare di perdere competenze e il controllo sul know how aziendale relativamente alle attività gestite tramite outsourcing, la progettazione del Sistema Informativo dovrebbe essere fatta sempre all'interno dell'azienda, appaltando esternamente solo la gestione delle problematiche tecnologiche.

### *Cloud computing nei Sistemi Informativi*

In informatica con il termine inglese cloud computing si indica un paradigma di erogazione di servizi offerti on demand da un fornitore ad un cliente finale attraverso la rete Internet (come l'archiviazione, l'elaborazione o la trasmissione dati), a partire da un insieme di risorse preesistenti, configurabili e disponibili in remoto sotto forma di architettura distribuita.

Le risorse non vengono pienamente configurate e messe in opera dal fornitore appositamente per l'utente, ma vengono assegnate, rapidamente e convenientemente, grazie a procedure automatizzate, a partire da un insieme di risorse condivise con altri utenti e lasciando all'utente parte dell'onere della configurazione. Quando l'utente rilascia la risorsa, essa viene similmente riconfigurata nello stato iniziale e rimessa a disposizione nel pool condiviso delle risorse, con altrettanta velocità ed economia per il fornitore.

La correttezza nell'uso del termine è contestata da molti esperti: se Rob van der Meulen e Christy Pettey vedono queste tecnologie come un'evoluzione tecnologica offerta dalla rete Internet, altri, come Richard Stallman, le considerano invece come una parola ingannevole ideata dal marketing per far cadere gli utenti nel tranello dei software offerti come servizio, che spesso li privano del controllo delle loro attività informatiche.

Il cloud computing prevede la presenza di almeno tre soggetti distinti: il fornitore dei servizi (**cloud provider**), l'amministratore del cliente (configura i servizi offerti dal fornitore), il cliente finale che utilizza i servizi configurati dall'amministratore.

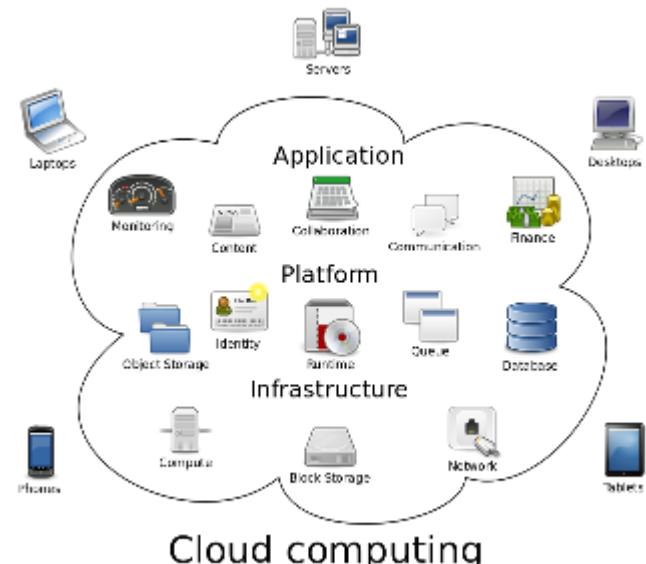
I servizi in cloud sono solitamente realizzati con architetture cluster che offrono alta affidabilità installati presso data center del cloud provider. Il cliente utilizza interface apposite per selezionare i servizi e configurarli. Tali servizi sono spesso associati a macchine virtuali.

Esistono differenti tipologie di cloud computing:

<b>Software as a Service</b>	SaaS	Utilizzo di programmi installati su un server remoto.
<b>Data as a Service</b>	DaaS	Utilizzo di dati messi a disposizione tramite Internet.
<b>Hardware as a Service</b>	HaaS	Utilizzo di risorse di calcolo remoto.
<b>Platform as a Service</b>	PaaS	Utilizzo di piattaforme applicative remote costituite da servizi, programmi, librerie (es. Amazon Web Services o Microsoft Azure)
<b>Infrastructure as a Service</b>	IaaS	Hardware, capacità di rete, memoria, archivi e backup istanziati su richiesta al momento del bisogno.

Il cloud pubblico offre i servizi remoti attraverso internet, tipicamente tramite un fornitore terzo. Offre il massimo livello di efficienza a fronte di una maggiore vulnerabilità agli attacchi. Rappresenta la soluzione migliore se:

- L'obiettivo è testare applicazioni in fase di sviluppo
- Si vuole utilizzare una specifica applicazione SaaS che si ritiene abbia un buon livello di protezione
- È necessario poter scalare rapidamente
- Si vuole realizzare un progetto in collaborazione con altri enti/aziende.



*Di Frank, B. Wilson - CloudNINE, CC BY-SA 3.0,  
<https://commons.wikimedia.org/w/index.php?curid=6080417>*



## VANTAGGI

- **Economici**

- Contenimento costi di startup
- Scalabilità a fronte di picchi di attività
- Ottimizzazione dei costi (risorse computazionali e risorse umane)
- Riduzione investimenti

- **Operativi**

- Riduzione (drastica) tempi di realizzazione e messa in esercizio dei servizi
- Scalabilità
- Rapido ed efficiente provisioni e deprovisioning delle risorse
- Ottimizzazione dei consumi energetici

## SVANTAGGI

- **Normativa privacy**

- La normativa non è nata pensando allo scenario cloud
- Le classiche figure previste non sono adatte al cloud
- I dati sono distribuiti geograficamente

- **Compromissione sicurezza dei dati**

- Isolamento risorse virtuali
- Compromissione interfacce di management
- Cancellazione dei dati
- Gestione identità
- Valutazione di SLA e penali legati alla sicurezza

- **Compromissione sicurezza della rete**

Il termine **appliance** identifica, in ambito informatico, un hardware integrato con il software utile a fornire una certa funzionalità. Questi sistemi non richiedono o non permettono configurazioni, parametrizzazioni o tuning. Non è richiesta, di conseguenza, attività specialistica per la loro messa in funzione. Nell'ambito dei sistemi informativi, alcuni esempi di DBMS realizzati tramite appliance sono: SAP HANA Appliance, Oracle Exadata/Exalogic, IBM Netezza...

A fronte di costi elevati, i vantaggi di questo tipo di soluzione sono:

- Riduzione costi e tempi di setup
- Riduzione costi del personale IT per manutenzione e tuning
- Performance elevate
- Buona scalabilità
- Elementi di base di potenziali cloud private.

### Individuazione dei benefici

<b>Benefici monetizzabili</b>	Si tratta di benefici ai quali è possibile collegare direttamente un valore monetario, ad esempio: <ul style="list-style-type: none"> <li>• Minor costo per riduzione del personale</li> <li>• Minori costi preesistenti</li> <li>• Minori costi per canoni affitto locali e archiviazione</li> <li>• Maggiore tempestività nella riscossione</li> <li>• Aumento dei servizi a pagamento</li> </ul>
<b>Benefici misurabili</b>	Non è possibile associare un valore economico ma è, tuttavia, possibile associare valori misurabili attraverso la definizione di KPI.
<b>Benefici intangibili</b>	Benefici valutabili solo in base a condizioni qualitative, aziendali, sociali o politiche. Ad esempio: miglioramento del servizio, miglioramento dell'immagine aziendale, maggiore motivazione e autonomia del personale, maggiori disponibilità e affidabilità delle informazioni, riduzione supporti cartacei, miglioramento metodologia operativa, efficacia ed efficienza dei processi.

Per poter effettuare analisi complete ed eventualmente confrontare differenti scenari è inoltre necessario definire i tempi di realizzazione di costi e benefici al fine di valutare le soluzioni migliori relativamente alla redditività del progetto.

### Esempio di benefici misurabili nella Pubblica Amministrazione

Macro aree di impatto	Arese di impatto	Indicatore
A. EFFICIENZA	A.1. Benefici monetizzabili	A.1.1. Risparmi nei costi del personale A.1.2. Risparmi costi vivi
	A.2. Empowerment degli operatori	A.2.1. Operatori destinatari di formazione A.2.2. Operatori riallocati ad altre mansioni
	A.3. Miglioramento organizzativo	A.3.1. Transazioni effettuate su nuovi canali (cittadini) A.3.2. Transazioni effettuate su nuovi canali (imprese) A.3.3. Reingegnerizzazione dei processi A.3.4. Documenti digitali autenticati
B. EFFICACIA	B.1. Riduzione del carico amministrativo	B.1.1. Tempo risparmiato dai cittadini B.1.2. Tempo risparmiato dalle imprese B.1.3. Costi risparmiati dai cittadini B.1.4. Costi risparmiati dalle imprese
	B.2. Soddisfazione degli utenti e livello di servizio	B.2.1. Limitazione dei disservizi B.2.2. Ottimizzazione tempi di evasione delle pratiche B.2.3. Servizi online offerti in orari supplementari B.2.4. Usabilità media dei servizi
C. DEMOCRAZIA	C.1. Trasparenza	C.1.1. Procedimenti amministrativi tracciabili via web
	C.2. Partecipazione	C.2.1. Spostamento utenza cittadini su nuovi canali C.2.2. Spostamento utenza imprese su nuovi canali C.2.3. Copertura servizi per i cittadini sul territorio C.2.4. Copertura servizi per le imprese sul territorio C.2.5. Cittadini autenticati/registrati all’uso dei servizi C.2.6. Imprese autenticate/registrate all’uso dei servizi

## Raccomandazioni per le fasi realizzative

### Struttura

Analisi situazione attuale -<sup>(25)</sup>

Progetti di massima -<sup>(20)</sup>

Analisi del rischio -<sup>(13)</sup>

Progetto proposto: scelte -<sup>(38)</sup>

Analisi costi-benefici -<sup>(27)</sup>

Raccomandazioni per le fasi realizzative

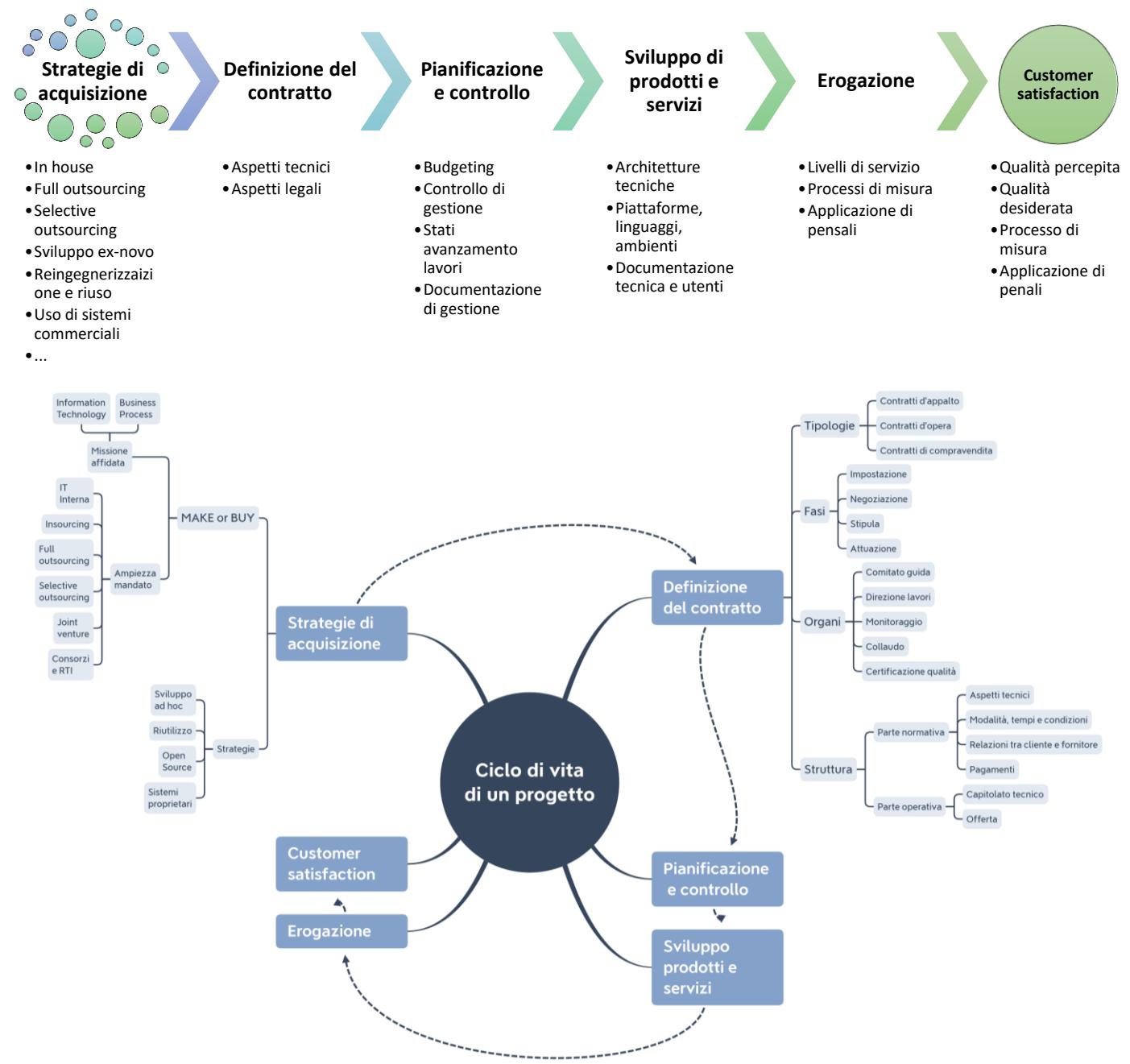


Le raccomandazioni sono il risultato degli approfondimenti effettuati durante lo studio di fattibilità, sono indicazioni da considerare per risolvere o minimizzare le problematiche rilevate.

Questa sezione di articola in:

1. Indicazioni per l'approvvigionamento
  - a. Criteri per la determinazione della tipologia di fornitore
  - b. Criteri di selezione delle offerte
  - c. Indicazioni sulle modalità di approvvigionamento
2. Indicazioni per la gestione del progetto
  - a. Indicazioni piano di qualità e gestione del rischio
  - b. Indicazioni project management
  - c. Esigenze di negoziazione delle varianti
3. Riepilogo degli elementi utili alla stesura del capitolo.

## Il ciclo di vita di un progetto



### Strategie di acquisizione: "make or buy"

#### Wikipedia

Il "make or buy" è la scelta di un'azienda o di un'organizzazione di costruire o di effettuare al proprio interno (make), oppure di acquistare all'esterno (buy), un componente, un prodotto o un servizio necessario alla produzione.

**La scelta si basa sul raffronto dei costi totali da sostenere nei due casi**, tenendo conto inoltre sia delle caratteristiche di reperibilità all'esterno (e quindi di eventuali criticità), sia delle risorse disponibili a questo fine all'interno dell'azienda.

Si tratta di una **scelta strategica fondamentale per la gestione d'azienda**, che definisce il livello di integrazione delle attività (a monte e a valle) e determina anche la struttura dei costi, l'organizzazione e il posizionamento sul mercato. L'opzione **make** (o gerarchia) offre soprattutto il vantaggio di garantire un **controllo diretto sull'attività**, sugli approvvigionamenti e sulla qualità del prodotto/servizio. Consente inoltre di mantenere eventuali segreti industriali.

L'opzione **buy** (o mercato) offre invece il vantaggio di comportare **minori costi fissi** e quindi **minore capitale immobilizzato**, consentendo una **maggior flessibilità della capacità produttiva**.

In generale l'esternalizzazione delle attività aumenta con la maturità del settore industriale, perché aumentano i vantaggi di specializzazione e le economie di scala.

La tendenza globale è quella di mantenere all'interno dell'impresa le attività della gestione caratteristica, basate sulle competenze chiave e quelle con forte potenziale di sviluppo, su cui si fonda il vantaggio competitivo di lungo termine, delegando all'esterno tutte le altre.

Solitamente la condizione delle organizzazioni prevede una scelta tra make e buy per specifici servizi o specifiche aree di sviluppo e sono presenti entrambe le condizioni: quasi nessun fa tutto in casa e quasi nessuno affida tutto all'esterno (anche per l'estrema difficoltà di governo che ne consegue).

### Outsourcing

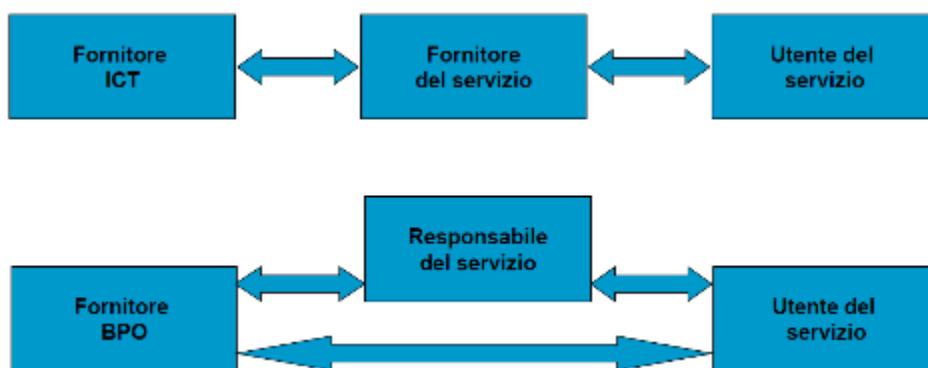
In sintesi, l'opzione di outsourcing (buy) offre una serie di vantaggi: minori costi di gestione, benefici finanziari, aumento della qualità, accesso a tecnologie avanzate, scalabilità.

È possibile classificare l'outsourcing secondo due dimensioni:

- **Missione affidata al fornitore**

- ✓ **Information Technology Outsourcing (ITO)**: sono affidate all'esterno attività di sviluppo, esercizio e manutenzione dei Servizi Informativi e delle risorse informatiche
- ✓ **Business Process Outsourcing**: sono affidati all'esterno i processi operativi dell'organizzazione (solitamente servizi non "core") quali, ad esempio, gestione del personale, contabilità e finanza, assistenza, relazione con gli utenti (CRM), acquisti e forniture (supply chain management), commercio elettronico, ...; in alcuni casi vengono affidati all'esterno servizi "core" attraverso accordi di collaborazione di particolare rilevanza fiduciaria (partnership).

#### Scenario tradizionale vs BPO



Esempio: servizi BPO di Accenture

<http://www.accenture.com/us-en/Pages/service-bpo-overview-summary.aspx>

➤ Cross-Industry BPO Services

- Engineering BPO
- Finance & Accounting BPO
- HR BPO
- Learning BPO
- Procurement BPO

- Supply Chain BPO
- Industry-Specific Services
  - Credit Services BPO
  - Health Administration BPO
  - Insurance BPO
  - Network BPO
  - Utilities BPO
- Bundled Outsourcing Services

- **Aampiezza del mandato conferito al fornitore**

- Funzione IT interna**

La funzione IT è assegnata ad una struttura interna dell'organizzazione, essa fornisce ed implementa nuovi servizi e architetture mediante progetti interni, può acquistare applicazioni, infrastrutture, hardware, software.

- Insourcing**

La funzione IT è delegata ad una società di servizi separata dall'organizzazione, tale società è di proprietà dell'organizzazione oppure esiste un rapporto (più o meno formalizzato): i servizi e le implementazioni sono regolati da un contratto (informale con un centro di ricavi oppure con definizione di tariffe).

Esempio: Consip (<http://www.consip.it/azienda/chi-siamo>)

Nel caso di BPO si parla di Captive company: società “prigioniera”, fondata da altre aziende allo scopo di eseguire operazioni per conto dell'azienda madre (es. FGA Capital gestisce il processo di finanziamento per la società FGA (Fiat Group Automobiles).

- Full outsourcing**

La funzione IT è delegata ad un unico fornitore esterno sulla base di un contratto di servizi, l'organizzazione vuole instaurare una partnership strategica con l'outsourcer. Rappresenta il classico modello di outsourcing, il contratto copre la maggior parte delle esigenze IT dell'organizzazione e ha una durata medio-lunga (5/10 anni).

- Selective outsourcing**

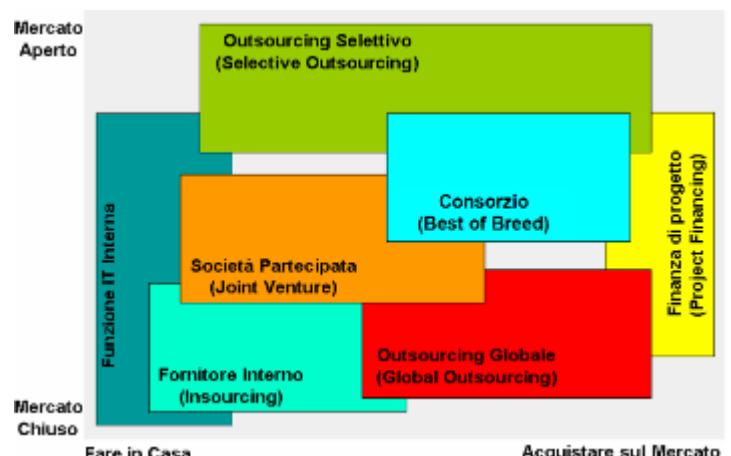
La funzione IT è delegata a diversi fornitori esterni: fornisce ed implementa nuovi servizi architetturali sulla base di diversi contratti di fornitura di durata medio-breve (3/5 anni). Può riguardare data center (facility), reti di telecomunicazioni, desktop e sistemi distribuiti, applicazioni e procedure. L'organizzazione in questo caso attua un approccio tattico per creare un ambiente competitivo con impatti su costi, capacità e innovazione a fronte di una maggiore complessità gestionale.

- Joint Venture**

La funzione IT è delegata ad una società di servizi separata e indipendente dall'organizzazione a cui fornisce servizi, in partecipazione con un fornitore. Le joint venture sono diffuse per gli enti locali.

- Consorzi e RTI**

La funzione IT è delegata ad un consorzio (stabile o temporaneo) costituito da un insieme di fornitori esterni: fornisce e implementa servizi a fronte di un contratto, prevede la costituzione di una partnership strategica con il consorzio, elevata complessità gestionale. Più spesso di tratta di Raggruppamenti Temporanei di Imprese (RTI) costituite per specifici progetti.



I servizi che possono essere affidati in outsourcing sono:

- **Application management:** manutenzione e conduzione del patrimonio applicativo software
- **Application service provision**
- **Servizi di consulenza:** direzione lavori, monitoraggio, consulenza e formazione
- **Desktop management:** postazioni di lavoro, assistenza, controllo e manutenzione hardware e software
- **Network:** connettività e gestione delle apparecchiature di rete
- **Facility management:** infrastrutture hardware (presso il committente o presso il fornitore), spesso include servizi di disaster recovery e business continuity
- **System integration**
- **Help desk, CRM:** infrastrutture e/o servizio

Outsourcing: pro e contro

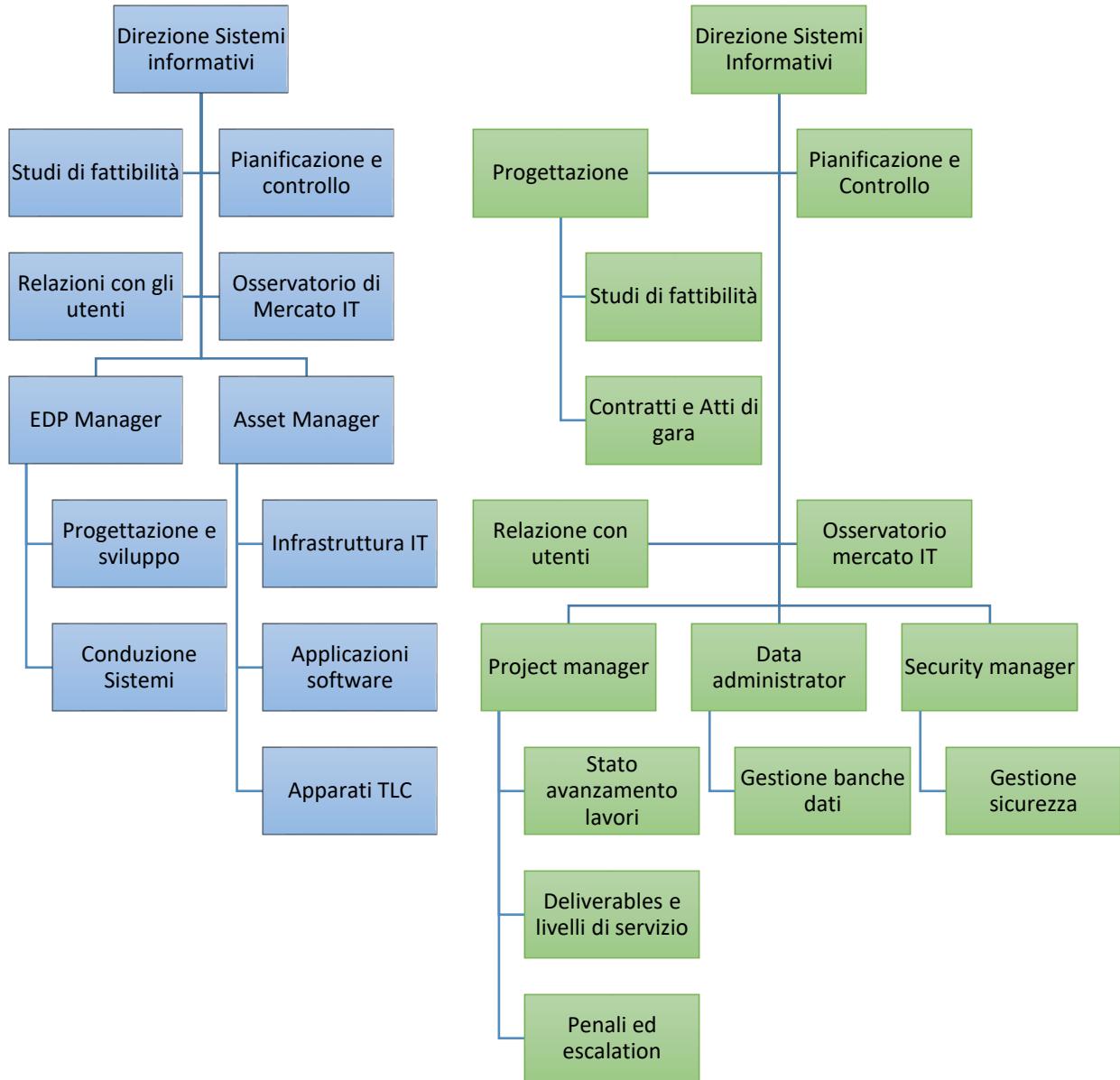
	PRO	CONTRO
<b>In generale</b>	<ul style="list-style-type: none"> <li>• Attenzione al core-business</li> <li>• Assenza di persone specializzate</li> <li>• Riduzione dei tempi, adeguamento ai rapidi cambiamenti tecnologici</li> <li>• Maggiore flessibilità nell'offerta dei servizi</li> <li>• Riduzione dei costi</li> </ul>	<ul style="list-style-type: none"> <li>• Perdita di controllo, rischi</li> <li>• Riduzione del potere negoziale a medio termine</li> <li>• Demotivazione del personale IT interno</li> </ul>
<b>Outsourcing globale</b>	<ul style="list-style-type: none"> <li>• Interfaccia unica</li> <li>• Unitarietà e integrazione delle componenti</li> <li>• Riduzione costi e tempi di acquisizione</li> <li>• Semplificazione della gestione contrattuale</li> </ul>	<ul style="list-style-type: none"> <li>• Limitata ottimizzazione</li> <li>• Perdita di controllo</li> <li>• Riduzione del potere negoziale e lock-in</li> <li>• Demotivazione del personale IT interno</li> <li>• Rischio di insuccesso globale</li> <li>• Complessità del singolo contratto</li> </ul>
<b>Outsourcing selettivo</b>	<ul style="list-style-type: none"> <li>• Clima di competizione tra fornitori</li> <li>• Controllo del committente su coordinamento e integrazione</li> <li>• Riduzione dei singoli tempi di acquisizione</li> <li>• Possibile semplificazione nella gestione dei singoli contratti</li> <li>• Minore rischio di insuccesso globale (rischio distribuito)</li> </ul>	<ul style="list-style-type: none"> <li>• Aumento della complessità nella gestione di molti contratti</li> <li>• Complessità nell'identificazione delle responsabilità (soprattutto in caso di problematiche... <i>scarica barile</i>)</li> <li>• Difficoltà di integrazione</li> </ul>

### Impatto organizzativo

Dal punto di vista dell'organizzazione le interazioni cliente-fornitore sono significativamente più complesse di quanto possa essere descritto e definito dal contratto che definisce un “contorno” all'interno del quale vengono gestiti risorse, servizi e rapporti.

L'esternalizzazione dei servizi informatici non prevede la soppressione della funzione IT interna, al contrario può determinare un incremento della fiducia in termini di responsabilità attribuite da parte della Direzione Aziendale nei confronti dell'IT interno visto come intermediario.

## Organizzazioni nei casi estremi: MAKE vs BUY



Un impatto organizzativo della scelta make or buy ricade anche sull’evoluzione delle competenze professionali del team IT: nel caso di outsourcing la funzione IT assume un ruolo più che altro di governo, pertanto diminuiscono i ruoli operativi e tecnici poco specializzati (operatori, programmatore) mentre aumentano i ruoli manageriali e i tecnici molto specializzati (capi progetto, analisti, sistemisti). L’outsourcing richiede meno personale interno ma con elevate competenze.

## Compiti e responsabilità della funzione IT

Progettazione	<ul style="list-style-type: none"> <li>➤ Studi di fattibilità e rappresentazione dei requisiti</li> <li>➤ Stima investimenti e analisi costi/benefici</li> <li>➤ Contratti e atti di gara</li> </ul>
Pianificazione e Controllo	<ul style="list-style-type: none"> <li>➤ Pianificazione informatica coerente alla missione</li> <li>➤ Definizione delle priorità dei progetti</li> <li>➤ Verifica del raggiungimento degli obiettivi</li> </ul>
Relazione con gli utenti	<ul style="list-style-type: none"> <li>➤ Acquisizione requisiti e bisogni reali</li> <li>➤ Verifica soddisfazione degli utenti</li> </ul>

Osservatorio mercato IT	➤ Controllo soluzioni proposte dai fornitori ➤ Contenimento del rischio di perdita di controllo
Gestione del progetto	➤ Gestione rapporti con i fornitori ➤ Budget e controllo di gestione ➤ Verifica stato avanzamento lavori ➤ Accettazione/collaudo dei prodotti ➤ Misurazione dei livelli di servizio ➤ Segnalazione tempestiva di rilievi e non conformità ➤ Proposta di azioni correttive o preventive ➤ Monitoraggio
Gestione delle banche dati (Data Administration)	➤ Sorveglianza qualità dei dati
Gestione della sicurezza (Security Management)	➤ Sorveglianza applicazione politiche di sicurezza ➤ Verifica rispetto della normativa vigente

### Fattori chiave e strategie di realizzazione di software applicativo

I **fattori chiave** per il successo dei progetti sono:

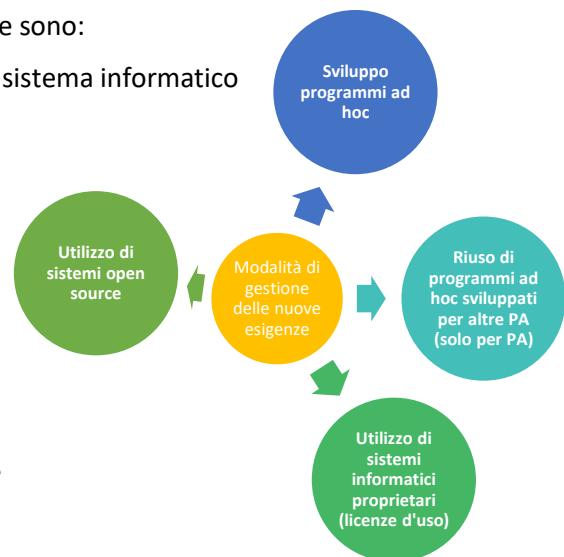
- La scelta della strategia di sourcing
  - ✓ Approccio progressivo
  - ✓ Adattamento alle mutazioni delle condizioni interne e/o esterne
- La selezione del fornitore
  - ✓ Ricerca delle garanzie necessarie a contenere i rischi
- Definizione del contratto
  - ✓ Responsabilità reciproche cliente-fornitore
  - ✓ Modelli di applicazione tariffe
  - ✓ Pariteticità, correttezza, funzionalità, flessibilità
- Governo del contratto.

Le cause che portano alla necessità di realizzare applicazioni software sono:

1. Nuove esigenze di automazione per processi non coperti dal sistema informatico
2. Adeguamento delle applicazioni esistenti
  - a. Manutenzione correttiva (MAC)
  - b. Manutenzione evolutiva (MEV)

Le nuove esigenze possono essere strategicamente gestite:

- **Sviluppando programmi ad hoc**
  - ✓ Funzioni peculiari o sistemi commerciali esistenti non adeguati allo scopo
  - ✓ Maggiore possibilità di personalizzazione
  - ✗ Maggiore incertezza su tempi e costi di realizzazione
  - ✗ Necessità di competenze interne specifiche
  - ✗ È considerato retaggio degli anni '80
- **Riutilizzando programmi sviluppati ad hoc per Pubbliche Amministrazioni (obbligo per le PA)**
  - ✓ Utile se le funzioni da informatizzare sono simili a quelle già realizzate
- **Utilizzando di sistemi open source**
  - ✓ Basso costo iniziale e maggior controllo sul costo complessivo d'esercizio (TCO)



- ✓ Maggiore indipendenza dai fornitori
- ✓ Possibile incremento della trasparenza (grazie all'accesso ai sorgenti)
- ✓ Maggiore possibilità di personalizzazione (ma a quale costo?)
- Minore compatibilità con standard commerciali/standard de facto
- Disponibilità limitata del supporto
- Instabilità del mercato e potenziale mancanza di una evoluzione del sistema
- **Utilizzando di sistemi proprietari tramite licenze d'uso**
  - ✓ Maggiore garanzia in termini di affidabilità e performance rispetto a open source
  - ✓ Talvolta vincolato per compatibilità con altre soluzioni presenti in azienda
  - ✓ Richiede valutazione del reale TCO (Total Cost Ownership) considerando anche i rischi legati alla mancanza di supporto adeguato, manutenzione correttiva ed evolutiva.

## Definizione del contratto

Wikipedia: Un contratto è un istituto giuridico che vincola due o più parti tra di loro. Il procedimento che porta alla sua stipula è detto contrattazione. Codice Civile, art. 1321, Nozione di contratto: Il contratto è l'accordo di due o più parti per costituire, regolare o estinguere tra loro un rapporto giuridico patrimoniale.

Sono previsti diversi tipi di contratto:

- **Contratti d'appalto** (art. 1655 CC). Contratto che prevede che una parte assuma, con organizzazione dei mezzi necessari e gestione a proprio rischio, il compimento di un'opera o di un servizio a fronte di un corrispettivo in danaro.
- **Contratti d'opera** (art. 2222 CC). Quando una persona si obbliga a compiere, verso un corrispettivo, un'opera o un servizio, con lavoro prevalentemente proprio e senza vincolo di subordinazione nei confronti del committente, si applicano le norme di questo capo, salvo che il rapporto abbia una disciplina particolare nel libro IV.
- **Contratti di compravendita** (art 1472 CC). Nella vendita che ha per oggetto una cosa futura, l'acquisto della proprietà si verifica non appena la cosa viene ad esistenza. Se oggetto della vendita sono gli alberi o i frutti di un fondo, la proprietà si acquista quando gli alberi sono tagliati o i frutti sono separati. Qualora le parti non abbiano voluto concludere un contratto aleatorio, la vendita è nulla, se la cosa non viene ad esistenza.

In ambito ICT possono esistere diverse tipologie di contratti specifici:

- forniture di apparecchiature ICT o di sistemi ICT completi (chiavi in mano)
- locazione di apparecchiature ICT o di sistemi ICT completi
- licenze d'uso di programmi software
- sviluppo di software
- outsourcing di servizi ICT.

Da un punto di vista manageriale un contratto ha l'obiettivo di soddisfare entrambe le parti coinvolte (ed eventualmente terzi coinvolti o interessati). Se i risultati non vengono raggiunti, committente e fornitore hanno fallito. Per questi motivi un contratto deve definire, in modo cooperativo tra le parti, le prestazioni (contenuti, costi, qualità responsabilità). Deve inoltre essere considerato che un contratto non può prevedere qualsiasi scenario o evento, pertanto deve specificare gli obblighi e gli impegni ma anche le procedure e le regole di relazione. L'efficacia (successo) di un contratto dipende da diversi fattori tra i quali hanno particolare rilevanza le **competenze tecniche e legislative** dei suoi estensori e le **competenze tecniche e manageriali** del personale preposto al governo durante il ciclo di vita del progetto. Un contratto si articola in generale secondo quattro fasi: impostazione (definizione oggetto e strategia di acquisizione), **negoziazione** (definizione del contratto) nel settore pubblico attraverso gare, **stipula, attuazione** (e governo). Le categorie di organi solitamente coinvolti nel governo di un contatto sono: comitato guida, direzione lavori, monitoraggio, collaudo, certificazione di qualità.

## Soggetti coinvolti

I soggetti solitamente coinvolti durante lo svolgimento di un contratto sono:

Per il committente	Per il fornitore
<ul style="list-style-type: none"> <li>➤ <b>dirigenti, definizione scelte strategiche</b> <ul style="list-style-type: none"> <li>○ responsabili missione istituzionale e politiche attuative</li> <li>○ responsabili strategie di acquisto</li> <li>○ responsabili sistemi informativi automatizzati</li> <li>○ responsabili utenti dei sistemi informativi automatizzati</li> </ul> </li> <li>➤ <b>personale funzione acquisti</b> <ul style="list-style-type: none"> <li>○ gruppi di lavoro per realizzazione atti di gara</li> <li>○ commissioni di gara</li> <li>○ direttore lavori <i>Project manager</i></li> <li>○ responsabile controllo di gestione</li> <li>○ commissioni di collaudo</li> </ul> </li> <li>➤ <b>personale funzione legale</b> <ul style="list-style-type: none"> <li>○ gruppi di lavoro per realizzazione atti di gara</li> <li>○ commissioni di gara</li> </ul> </li> <li>➤ <b>personale funzione sistemi informativi</b> <ul style="list-style-type: none"> <li>○ gruppi di lavoro per studi di fattibilità e atti di gara</li> <li>○ commissioni di gara</li> <li>○ direttore lavori</li> <li>○ gruppi di monitoraggio</li> <li>○ commissioni di collaudo</li> </ul> </li> <li>➤ <b>personale utente dei servizi informativi</b> <ul style="list-style-type: none"> <li>○ gruppi di lavoro per studi di fattibilità e atti di gara</li> <li>○ commissioni di gara</li> <li>○ commissioni di collaudo</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>➤ <b>dirigenti, definizione scelte strategiche</b> <ul style="list-style-type: none"> <li>○ responsabili marketing</li> <li>○ responsabili commerciali</li> <li>○ responsabili legali</li> <li>○ responsabili assicurazione e controllo qualità</li> <li>○ responsabili erogazione servizi</li> </ul> </li> <li>➤ <b>personale funzione commerciale</b> <ul style="list-style-type: none"> <li>○ gruppi di lavoro realizzazione offerte</li> <li>○ responsabile marketing settore di mercato P.A.</li> <li>○ responsabile cliente e/o contratto (Account manager)</li> </ul> </li> <li>➤ <b>personale funzione legale</b> <ul style="list-style-type: none"> <li>○ gruppi di lavoro realizzazione offerte</li> </ul> </li> <li>➤ <b>personale funzione erogatrice dei servizi ICT</b> <ul style="list-style-type: none"> <li>○ gruppi di lavoro per studi di fattibilità e offerte</li> <li>○ responsabile di progetto <i>Project manager</i></li> <li>○ responsabile controllo di gestione del progetto <i>Project controller</i></li> <li>○ responsabile controllo qualità e collaudi interni <i>Quality controller</i></li> <li>○ responsabile erogazione specifici servizi ICT</li> </ul> </li> <li>➤ <b>personale della funzione assicurazione qualità</b> <ul style="list-style-type: none"> <li>○ gruppi di lavoro per studi di fattibilità e offerte</li> <li>○ responsabile assicurazione e controllo qualità <i>Quality manager</i></li> <li>○ responsabile analisi soddisfazione utente <i>Customer satisfaction</i></li> </ul> </li> </ul>

## Struttura di un contratto

Un contratto è strutturato in due parti:

- **Parte normativa:** *corpo del contratto* che definisce e corrella aspetti tecnici, modalità tempi e condizioni, relazioni tra cliente e fornitore, corrispettivi (pagamenti).
- **Parte operativa:** *capitolato tecnico* e *offerta*. Il capitolato tecnico è predisposto dal cliente (insieme al bando, in caso di gara) e fornisce al fornitore le informazioni di dettaglio utili per preparare l'offerta. L'offerta è redatta dal fornitore in risposta a quanto richiesto nel capitolato tecnico e ha lo scopo di dimostrare al cliente la capacità e le modalità per soddisfare quanto richiesto nel capitolato tecnico. L'offerta tecnica può contenere, allegati, il piano di progetto e il piano di qualità.

È tuttavia facile sovrapporre le due parti e, talvolta, generare delle incoerenze tra esse.

In caso di gara, *corpo del contratto* e *capitolato tecnico* fanno parte della documentazione predisposta dal committente mentre l'*offerta* è la risposta (coerente) del fornitore.

Esempio: *contratto di manutenzione ed evoluzione dei sistemi informativi*

Di seguito sono rappresentati l'indice e alcune sezioni di un contratto sviluppato da CONSIP (società pubblica che fornisce consulenza tecnologica, organizzativa e di progetto agli uffici del Ministero dell'Economia).

Il contratto prevede una serie di servizi per aree:

- Sviluppo e manutenzione evolutiva (MEV) di software ad hoc
- Gestione applicativa

- Manutenzione adeguativa e correttiva (MAC)
- Supporto specialistico

In molti casi si utilizza una struttura in due parti: **parte generale** (comune a tutti i contratti) include gli aspetti standard e **parte speciale** con gli aspetti peculiari caso per caso.

Lo schema del contratto CONSIP di esempio prevede una parte generale che contiene:

- Aumento e diminuzione (standard 20%, “sesto quinto”)
- Modalità di esecuzione: luogo e *convivenza*, impiego di risorse specializzate
- Rispetto della normativa sui rapporti di lavoro (sicurezza, igiene, previdenza infortuni, contratti collettivi) ed eventuali penalizzazioni
- Obblighi di riservatezza, risoluzione
- Utilizzo software e hardware (l’impresa deve essere autorizzata)
- Danni, responsabilità civile e assicurazione
- Oneri fiscali e spese contrattuali
- Cauzione (10% del valore contrattuale, ridotto se l’azienda è certificata per la qualità, aumentato in caso di ribasso significativo); il costo di una polizza per la cauzione è pari allo 0,5-1% del valore assicurato
- Recesso (committente con preavviso, fornitore no) e recesso per giusta causa
- Divieto di cessione del contratto e del credito
- Trasparenza dei prezzi: assenza di intermediazione e rispetto per la concorrenza
- Subappalto: previsto a priori e con vincoli e responsabilità che restano sul fornitore e si ripetono sul subappaltatore
- Foro competente esclusivo
- Trattamento dati personali
- Condizioni particolari di risoluzione (antimafia, verifica autocertificazioni, sanzioni interdittive).

La parte speciale contiene i seguenti articoli.

## 1. Oggetto, luogo della prestazione e responsabile del procedimento

Servizi di:

- ✓ Sviluppo e manutenzione evolutiva di software ad hoc
- ✓ Gestione applicativa
- ✓ Manutenzione adeguativa e correttiva (MAC)
- ✓ Supporto specialistico

Dettagli:

- ✓ Riferimento al capitolato e all’offerta tecnica
- ✓ Trasferimento di know-how
- ✓ Il tutto in misura almeno pari al 10% dell’importo contrattuale

## 2. Durata e affiancamento

60 mesi

- ✓ Di cui gli ultimi 12 solo per garanzia
- ✓ I primi due: affiancamento del fornitore uscente
- ✓ Ultimi due: anche trasferimento know-how al committente o a terzi

## 3. Obblighi e adempimenti a carico dell’impresa

- ✓ Oneri e rischi a carico dell’impresa inclusi viaggi e missioni se necessari
- ✓ Esecuzione a regola d’arte e con rispetto di regole tecniche e norme di sicurezza
- ✓ Rispetto delle indicazioni del committente
- ✓ Rispetto dei requisiti di accessibilità Web

- ✓ Disponibilità a verifiche da parte del committente
- ✓ Risoluzione e danni

#### **4. Proprietà del software sviluppato e dei prodotti in genere**

- ✓ L'Amministrazione acquisisce la proprietà del software e della documentazione
- ✓ Possibilità per l'Amministrazione di acquistare licenze e pacchetti utilizzati dal fornitore
- ✓ Possibilità di riuso per altre amministrazioni alle medesime condizioni
- ✓ Possibilità di utilizzare componenti open source

#### **5. Dimensioni massime dei singoli servizi**

- ✓ Sviluppo e manutenzione e evolutiva
  - 60.2000 punti funzione (59.800 nuovi, 4.000 eliminati, pesati al 10%)
- ✓ Gestione applicativa
  - 38.320 giorni persona
- ✓ Manutenzione adeguativa
  - 1.840 giorni persona
- ✓ Manutenzione correttiva
  - 210.900 punti funzione
- ✓ Supporto specialistico
  - 6.520 giorni

Con possibilità di travaso.

#### **6. Piano della qualità**

Il fornitore deve predisporre:

- ✓ Piano della Qualità generale
- ✓ Piani della Qualità per i vari obiettivi

Il committente verifica e può richiedere modifiche.

Il Piano della qualità descrive un servizio in termini di **struttura organizzativa, responsabilità e risorse impiegate, procedure e procedimenti, definizione di “chi”, “cosa”, “come” e “quando”** assicura la qualità del servizio (controllo di processo, descrizione del metodo di lavoro) e garantisce (cerca di garantire) il fornitore che lo utilizza, il cliente che fruisce servizi supportati nel SQ.

##### **Piano della qualità di esempio**

1. **Scopo** del piano della qualità (Contiene le finalità del Piano della Qualità)
2. **Documenti applicabili e di riferimento** (Contiene l'elenco sia di tutti i documenti contrattuali applicabili e sia di tutti i documenti che costituiscono un riferimento per quanto esposto nel presente Piano della Qualità)
3. **Glossario** (Contiene tutte le abbreviazioni, gli acronimi, le definizioni che sono utilizzate all'interno del Piano della Qualità)
4. **Organizzazione** della fornitura (Contiene l'organigramma del gruppo di lavoro impegnato sul contratto senza l'indicazione dei nominativi (con l'identificazione del responsabile unico delle attività , dei coordinatori delle attività del servizio di gestione applicativa ove previsto nel capitolato, dei responsabili della gestione configurazione) e le relazioni con le altre organizzazioni coinvolte nella fornitura. A ciascun ruolo indicato nell'organigramma, deve essere associata una precisa responsabilità, in modo che ciascun componente del gruppo di lavoro abbia ben chiari i ruoli, i compiti, le responsabilità ed i poteri nell'ambito del contratto. Utilizzare una matrice, denominata “matrice delle responsabilità”, per sintetizzare le responsabilità assegnate)
5. **Ciclo di vita del software applicativo** (Describe il ciclo di vita del software applicativo, le fasi in cui è suddiviso, i criteri di uscita delle fasi, e l'insieme della documentazione da produrre. Qualora si utilizzino diversi cicli di vita, suddividere il paragrafo in sotto paragrafi relativi ai diversi cicli di vita previsti)
6. **Ciclo di erogazione dei servizi** (Contiene la definizione del ciclo di erogazione di ciascun servizio contrattuale, la descrizione dei processi coinvolti nel ciclo e l'insieme della documentazione da produrre)
7. **Metodi, tecniche e strumenti**
  - 7.1. **Progettazione** del software applicativo (Describe le metodologie, le tecniche e gli strumenti che si intendono adottare per la progettazione, la realizzazione ed il test del software applicativo)
  - 7.2. **Scrittura e documentazione** del software applicativo (Riporta o referenzia gli standard che si intendono adottare per la stesura del codice sorgente e per la stesura dei commenti nel codice sorgente)

- 
- 7.3. Progettazione ed esecuzione dei **test** (Riporta o referenzia le linee guida ed i principi ispiratori per la progettazione ed esecuzione delle sessioni di test sia per i nuovi sviluppi che per le mev, ivi compresi i test di conformità ai requisiti di accessibilità stabiliti dal decreto del Ministro per l'innovazione e le tecnologie dell'8 luglio 2005)
  - 7.4. **Erogazione dei servizi** (Describe le metodologie, le tecniche e gli strumenti che si intendono adottare per l'erogazione dei servizi)
  - 7.5. **Standard documentali** (Contiene gli standard della redazione della documentazione)
  - 8. Requisiti di qualità
    - 8.1. Identificazione dei **requisiti di qualità** (Contiene la chiara e non ambigua identificazione degli indicatori di qualità. Per questo è necessario definire:
      - gli **attributi di qualità** (caratteristiche e sotto-caratteristiche nella terminologia ISO 9126) relativi a ciascun prodotto ed i livelli di servizio relativi a ciascun servizio;
      - gli **indicatori** con cui misurare gli attributi ed i livelli identificati;
      - i **valori limite** ritenuti accettabili con cui confrontare le misure degli attributi di qualità e dei livelli di servizio effettuate sulla base di indicatori definiti)
    - 8.2. **Procedura per la valutazione della qualità** (Riporta o referenzia la procedura per la valutazione della qualità dei prodotti e/o servizi. La procedura deve esplicitare:
      - **modalità di misura**;
      - **modalità di calcolo** e di aggregazione delle misure (per il computo di indicatori derivati);
      - **frequenza delle misure**;
      - **periodi temporali** di riferimento;
      - le **regole con cui si perviene ai giudizi** di Approvazione Incondizionata / Approvazione con Riserva / Non Approvazione di un prodotto e/o un servizio considerando i risultati delle misure relative ai singoli attributi di qualità associati al prodotto e/o livelli di servizio associati al servizio
  - 9. **Registrazioni della qualità** (Riporta l'elenco di tutte le registrazioni della qualità, sia quelle previste dal sistema qualità adottato, sia specificatamente previste per l'attuazione del contratto, necessarie a supportare le attività di gestione del contratto e di assicurazione della qualità)
  - 10. **Verifiche ispettive** (Definisce o referenzia le modalità con cui effettuare le visite ispettive interne sulle attività della fornitura)
  - 11. **Riesami, verifiche e validazioni** (Contiene l'elenco dei controlli da effettuare (riesami, test, verifiche e validazioni, valutazioni, ecc) per le attività della fornitura, e le modalità di esecuzione dei controlli comprensive sia degli strumenti da utilizzare e sia della modulistica di rendicontazione dei risultati)
  - 12. **Segnalazione di problemi ed azioni correttive** (Riporta o referenzia le specifiche procedure previste per la gestione di problemi quali malfunzionamenti e non conformità. La descrizione deve comprendere la casistica, la modulistica di supporto prevista, i ruoli e le responsabilità delle risorse coinvolte)
  - 13. **Controllo della configurazione** del software (Contiene la descrizione dei criteri, delle procedure e degli strumenti adottati per il controllo (immissione, salvaguardia e catalogazione) e la consultazione delle versioni degli elementi software)
  - 14. **Controllo dei sub-fornitori** (Delinea le procedure e gli accorgimenti da adottare per il controllo dei sub-fornitori)
  - 15. **Raccolta e salvaguardia dei documenti** (Contiene la descrizione della procedura per la gestione, conservazione e salvaguardia della documentazione di progetto, nonché il periodo di mantenimento previsto della documentazione. Inoltre riporta o referenzia le modalità di identificazione, archiviazione, protezione, reperibilità delle registrazioni della qualità ed il periodo previsto di mantenimento delle registrazioni)
  - 16. **Formazione ed addestramento** (Contiene la descrizione delle attività di formazione inerenti al contratto. Tali attività riguardano sia gli eventuali aggiornamenti tecnici a cui sottoporre le risorse del fornitore che lavorano per l'espletamento del contratto, sia l'addestramento degli utenti all'uso dei prodotti/servizi contrattualmente previsti)
  - 17. **Gestione del prodotto** fornito dal cliente (Describe le modalità di gestione dei prodotti e degli strumenti forniti dall'Amministrazione/Consip)
  - 18. **Gestione dei rischi** (Contiene la metodologia e le modalità operative di identificazione e controllo dei rischi)
  - 19. **Analisi dei dati per il miglioramento** (Describe le modalità di rilevazione, analisi e rendicontazione dei dati per le attività legate al miglioramento interno)

**Indicatori di qualità (esempio)**

Indicatori di Qualità	Azione contrattuale		
	Rilievo	Penale	IP
IQ01 – Slittamento nell'esecuzione dell'obiettivo		•	
IQ02 – Slittamento della consegna di un prodotto dell'obiettivo	•		
IQ03 – Rilievi sull'obiettivo		•	•
IQ04 – Test negativi in collaudo	•		
IQ05 – Difettosità in collaudo	•		
IQ06 – Giorni di sospensione del collaudo	•		
IQ07 – Slittamento della risoluzione dei malfunzionamenti in collaudo	•		
IQ08 – Difettosità in esercizio (per applicazione)		•	•
IQ09 – Densità dei commenti del software sviluppato	•		
IQ10 – Linee di codice inerte	•		
IQ11 – Essential Complexity	•		
IQ12 – Violazioni dell'incapsulamento da parte di una Classe	•		
IQ13 – Dipendenza di una Classe dai suoi Child	•		
IQ14 – Metodi implementati in una Classe	•		
IQ15 – Complessità Ciclomatica di una Classe	•		
IQ16 – Grado di Coesione dei Metodi di una Classe	•		
IQ17 – Tempestività di ripristino dell'operatività (categoria 1)		•	
IQ18 – Tempestività di ripristino dell'operatività (categoria 2)	•	•	
IQ19 – Tempestività di ripristino dell'operatività (categoria 3 e 4)	•	•	
IQ20 – Case recidivi (per area applicativa)		•	•
IQ21 – Tempo medio di risposta all'utente o di prima diagnosi	•		
IQ22 – Slittamento nella consegna di un deliverable o nell'erogazione di un servizio della fornitura	•	•	
IQ23 – Slittamento nell'inserimento / sostituzione del personale		•	
IQ24 – Personale della fornitura inadeguato			•
IQ25 – Turn over del personale	•		
IQ26 – Soddisfazione del committente			•
IQ26 – Rilievi sulla fornitura		•	

**Livelli di servizio**

Si tratta di elementi quantitativi volti a definire sogli minime di accettazione per i vari elementi della fornitura: misurano il valore rappresentato da un attributo di servizio da due punti di vista: tecnico (livelli di servizio) e utente (requisiti di servizio).

Sono esempi di livelli di servizio tecnici: disponibilità dei server (soglia 99%, periodo di osservazione trimestrale, finestra temporale feriale 8:00-18:00, penali 5K€ ogni punto % di diminuzione), disponibilità della rete, disponibilità delle postazioni di lavoro.

Esempi di requisiti di servizio utente sono: disponibilità complessiva del sito Web, tempo massimo di interruzione.

In fase di stesura del contratto (negoziazione) l'utente esprime le proprie esigenze che vengono formalizzate nei requisiti da raggiungere; la funzione informativa rappresenta i requisiti utente tramite livelli di servizio con l'obiettivo di correlare le esigenze di servizio per gli utenti con i criteri tecnici.

**7. Garanzie**

Il fornitore garantisce l'eliminazione di eventuali difetti riscontrati (con opportune modalità nell'ultimo anno).

**8. Subappalto**

Vengono indicate le prestazioni subappaltata e i subappaltatori.

**9. Pianificazione delle attività**

Gli interventi vengono pianificati in accordo tra le parti tramite un Piano di lavoro generale (subentro, trasferimento know-how, servizi continuativi) e/o un piano per obiettivo di sviluppo:

- il committente richiede al fornitore la quantificazione di un obiettivo
- il fornitore stima (impegno umano e punti funzione)
- il committente valuta l'offerta
- il fornitore predisponde un piano di lavoro, rendiconta e consuntiva.

## 10. Produttività e risorse impiegate

- Produttività del personale (espressa in punti funzione per giorno uomo)
- Indicazione del responsabile
- Curricula delle persone impegnate (valutazione del committente che può richiedere sostituzioni)
- Regole per la sostituzione da parte del fornitore.

## 11. Consegna dei prodotti

- Rispetto dei tempi e degli standard (base e migliorativi)
- Accettazione formale del committente
- Penali.

## 12. Collaudo e accettazione

Controllo di prodotto:

- Verifica dell'esatto e completo adempimento da parte del fornitore di quanto oggetto del contratto
- Verifica della conformità e del grado di funzionamento delle infrastrutture IT e dei software

Viene effettuato da esperti incaricati dal cliente: soggetti diversi da chi ha diretto l'esecuzione dei lavori, prevede il coinvolgimento degli utenti, viene effettuato alla presenza di incaricati del fornitore.

In caso di esito negativo (non vengono superate le prove funzionali e diagnostiche) è possibile prevedere delle penali. Le operazioni di collaudo vengono ripetute con le stesse condizioni e modalità entro un termine fissato contrattualmente. In caso di ulteriore esito negativo è prevista la risoluzione del contratto per inadempimento, l'incameramento del deposito cauzionale pagato dal fornitore e il diritto al risarcimento dell'eventuale danno causato.

In caso di contratto di outsourcing, l'analisi da parte del cliente riguarda quantità e qualità delle risorse impiegate, la produttività raggiunta in sede di esecuzione contrattuale e la documentazione prodotta. Si utilizzano strumenti di verifica diretta delle prestazioni, dell'efficienza e dell'efficacia attraverso un monitoraggio con sistemi di misura dei livelli di servizio.

## 13. Monitoraggio

Il controllo di processo vuole assicurare la qualità: consiste in un esame periodico delle prestazioni di servizio rese con realizzazione di rapporti sulla misura dei livelli di servizio, prevede verifiche ispettive sull'erogazione dei servizi e la rilevazione e diagnosi di problemi e l'individuazione di azioni correttive. Il controllo di processo è parallelo al ciclo di vita per la realizzazione del prodotto intervenendo su anomalie emerse in corso d'opera. Allo scopo di limitare i costi si utilizza il principio "Just Enough Quality" per cui la "qualità giusta" è quella sufficiente.

Il **monitoraggio** rappresenta l'azione continua e parallela all'esecuzione del contratto a supporto della direzione lavori. Si occupa di osservare:

- Modalità di conduzione del contratto
- Stato avanzamento dei lavori
- Quantità, qualità, livelli di servizio dei beni forniti e dei servizi erogati
- Processi messi in atto dal fornitore per erogare i servizi.

La **vigilanza in corso d'opera** è effettuata in relazione all'attuazione dei contratti informatici (è obbligatoria per i contratti di grande rilievo della P.A., art. 13 D.Lgs 12/39/93 Circolare 5/94); deve essere affidata ad una terza parte indipendente rispetto ai contraenti e qualificata dal CNIPA (P.A. Circolari 16/98, 17/98) e deve essere svolta sotto la responsabilità di un Direttore Tecnico. La vigilanza è complementare al collaudo e mira a garantire il raggiungimento degli obiettivi tramite prevenzione.

Grazie al controllo costante in tutte le fasi del ciclo di vita di beni e servizi forniti e all’analisi della conduzione del contratto (attuata dal fornitore) e della qualità dei prodotti forniti e dei servizi erogati durante tutte le fasi del ciclo di vita, è possibile mettere in piedi **azioni di prevenzione** all’insorgere di eventuali anomalie.

Un’azione di **diagnosi** permette di identificare le cause delle anomalie e suggerisce le conseguenti **azioni correttive** messe in atto a cura del fornitore sotto la responsabilità del cliente.

Azioni di consuntivo prevedono la raccolta sistematica dei dati relativi al contratto (anche a supporto della pianificazione e per stime di futuri progetti) allo scopo di monitorare l’andamento delle prestazioni erogate in termini di livelli di servizio, risorse utilizzate e problemi riscontrati durante lo svolgimento delle attività e individuare modalità di risoluzione secondo best practices.

Sono **obblighi contrattuali del fornitore**:

- l’accettazione che le azioni svolte in fase di esecuzione del contratto siano sottoposte a monitoraggio
  - i. garantire l’accesso a uffici o impianti presso i quali si svolgono le attività
  - ii. accettare verifiche ispettive
- la designazione di un responsabile dei rapporti con l’incaricato del monitoraggio
- la disponibilità a collaborare fornendo tempestivamente la documentazione:
  - i. di riscontro (piano di progetto, piano della qualità)
  - ii. di consuntivo (rapporti periodici, stato avanzamento lavori)
  - iii. di supporto alla misura dei livelli di servizio (registrazioni)
- la messa a disposizione degli elementi di fornitura (software, documentazione, ...)
- la partecipazione alle sedute congiunte di riesame delle attività.

#### 14. Penali

Il contratto può definire penalità pecuniarie da applicare in caso di inadempimenti:

- non hanno lo scopo di far risparmiare il committente in funzione di un minor servizio ricevuto
- servono a rafforzare l’impegno del fornitore nel rispettare i livelli di servizio previsti dal contratto
- devono essere correlate all’entità dell’inadempimento.

Alcuni esempi:

- ritardi nella consegna e messa in funzione dei sistemi
- collaudi negativi
- fermi hardware (non ripristinati nei tempi previsti)
- malfunzionamenti software (non ripristinati nei tempi previsti)
- mancato raggiungimento dei livelli di servizio previsti.

#### 15. Corrispettivo

Definisce il pagamento corrisposto dal committente al fornitore, eventualmente determinato in modo articolato.

**Corrispettivo “a corpo”** (prezzo fisso, rischio d’impresa, ordine chiuso): definisce un valore globale; si utilizza quando è possibile definire chiaramente la fornitura e il committente vuole assicurarsi un prezzo contrattuale definito al termine della stipula; sposta il rischio sul fornitore. Si utilizza anche, in maniera impropria, quando è difficile stimare a priori le spese necessarie o non si dispone delle specifiche e ma si desidera fissare un vincolo. Il pagamento può avvenire in fasi predefinite in funzione di raggiungimento di stati di avanzamento predeterminati.

**Corrispettivo “a misura”**: definisce valori unitari (su due categorie di misure fondamentali: prodotti finiti, software e volumi di servizio, e/o risorse umane e/o ICT utilizzate) e quantità e relative unità di misura, solitamente sono definiti limiti minimi e massimi. Il pagamento avviene su base periodica, tipicamente 3/6 mesi, calcolando il corrispettivo a consuntivo in base alla misurazione dei parametri predefiniti e quotati in sede contrattuale. Sposta il rischio sul committente.

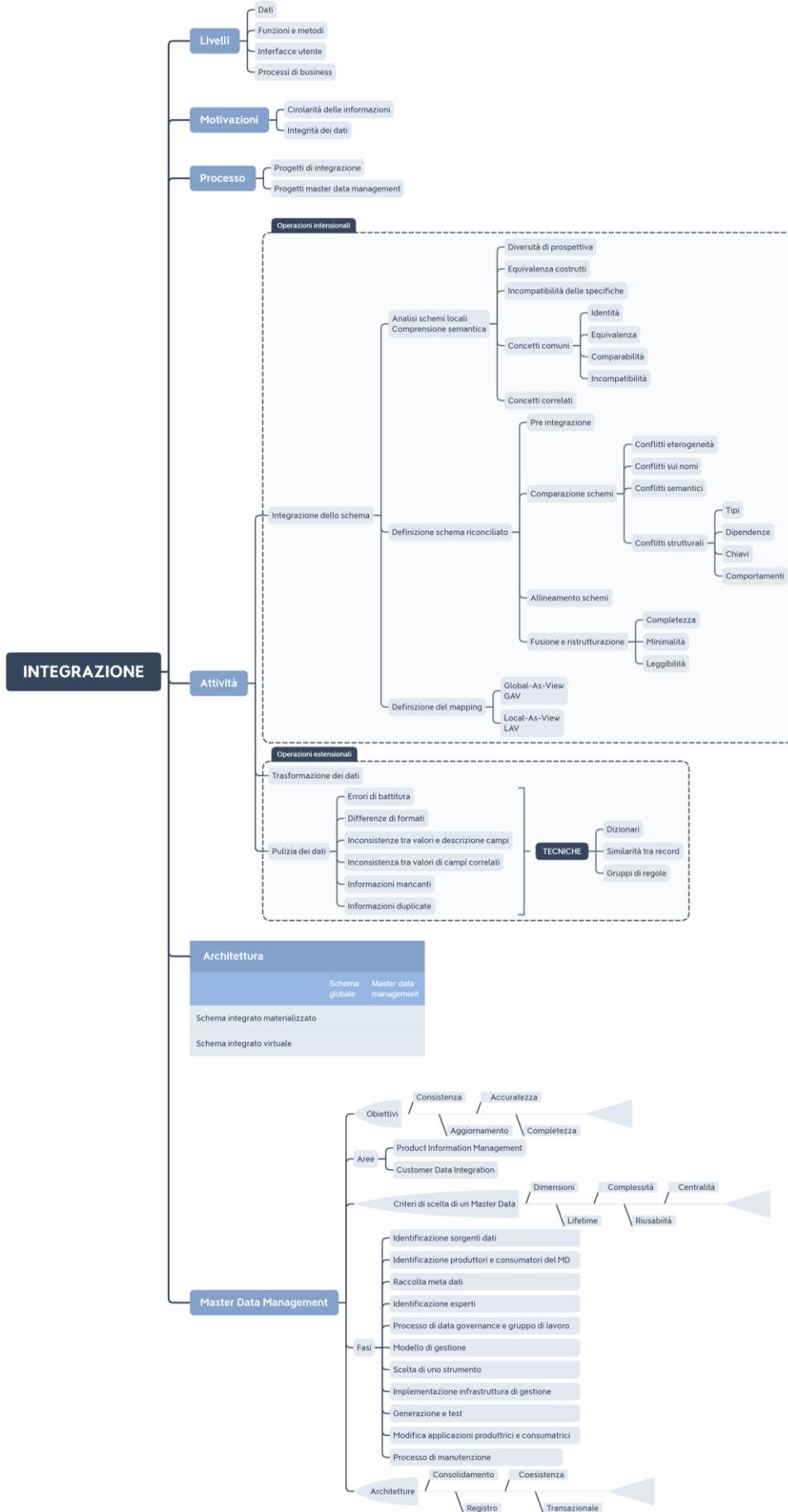
Alcune soluzioni intermedie possono essere: corrispettivo a corpo con correttivi, corrispettivo in parte a corpo in parte a misura, corrispettivo con premi legati ad indici di prestazioni.

	<b>Prodotti realizzati</b>	<b>Risorse utilizzate</b>
<b>Caratteristiche</b>	<p>Il contratto definisce il limite superiore di prezzo stimando <u>la quantità di software necessaria</u>.</p> <p>Le tariffe sono concordate per unità di prodotto (functional points, lines of code, ...).</p>	<p>Il contratto definisce il limite superiore di prezzo stimando <u>l'impegno necessario</u>.</p> <p>Le tariffe sono unitarie per figura professionale. Il corrispettivo è pagato a consuntivo in base alle risorse impiegate.</p>
<b>Applicazione</b>	Sviluppo e manutenzione evolutiva del software.	Quando non è possibile applicare il modello relativo ai prodotti realizzati.
<b>Modalità di controllo</b>	Verifica finale, collaudo del software prodotto, misura critica della dimensione di software sviluppato.	Verifica finale, collaudo del software prodotto, misura dell'impegno sostenuto (time sheets) e valutazione congruità.
<b>Vantaggi</b>	<ul style="list-style-type: none"> <li>✓ Riduzione dei costi</li> <li>✓ Garanzia realizzazione di tutti gli sviluppi previsti</li> <li>✓ Cliente protetto da possibili sorprese</li> <li>✓ Fornitore incentivato ad aumentare la produttività</li> <li>✓ Flessibilità rispetto all'instabilità delle normative</li> </ul>	<ul style="list-style-type: none"> <li>✓ Semplice gestione del contratto</li> <li>✓ Applicabile alla manutenzione adeguativa o correttiva</li> <li>✓ Fornitore protetto da possibili sorprese</li> <li>✓ Flessibilità rispetto all'instabilità delle normative</li> </ul>
<b>Svantaggi</b>	<ul style="list-style-type: none"> <li>✗ Complessa gestione del contratto</li> <li>✗ Non applicabile facilmente a manutenzione adeguativa o correttiva</li> </ul>	<ul style="list-style-type: none"> <li>✗ Il cliente si assume i rischi (volatilità delle specifiche, imprevisti, errata stima dell'impegno)</li> </ul>
<b>Criticità</b>	<ul style="list-style-type: none"> <li>• Definizione delle tariffe unitarie per unità di prodotto</li> <li>• Effettiva applicazione del ciclo di vita del software definito contrattualmente (sorveglianza sul processo produttivo del fornitore che potrebbe contrarre risorse e costi a discapito della qualità)</li> </ul>	<ul style="list-style-type: none"> <li>• Definizione delle tariffe unitarie per figura professionale</li> <li>• Assenza di correlazione tra prodotto ottenuto e spesa sostenuta (a meno di verifiche accurate sulla qualità dei prodotti)</li> </ul>
<b>Garanzie</b>	Correlazione tra prodotto sviluppato e costo sostenuto	

## 16. Fatturazione

Articolazione temporale dei pagamenti.

## Integrazione



## Cosa significa integrazione?

Con il termine **integrazione** si indica l'insieme delle attività necessarie per costruire una *versione completa e consistente* del sistema informatico.

In generale, per integrazione s'intende l'interazione funzionale di uomini, processi e tecnologie con l'obiettivo di creare un insieme unitario.

Con il termine inglese **system integrator** viene indicata un'azienda (o uno specialista) che si occupa dell'integrazione di sistemi. Il compito del system integrator è quello di far dialogare impianti diversi tra loro allo scopo di creare una nuova struttura funzionale che possa utilizzare sinergicamente le potenzialità degli impianti d'origine e creando quindi funzionalità originariamente non presenti. Gli integratori di sistemi possono lavorare in campi diversi, ma il termine è nato nel contesto delle tecnologie dell'informazione, dell'industria della difesa e dei media.

Il termine **Enterprise Application Integration** (EAI, integrazione d'applicazioni di impresa) si riferisce al processo d'integrazione tra diversi tipi di sistemi informatici attraverso l'utilizzo di software e soluzioni architetturali. L'EAI può avvenire a quattro livelli:

- **Integrazione orientata ai dati:** Avviene a livello di database o archivi dati, può essere in real-time o no e di solito è composta di trasferimenti batch, unioni di dati, repliche di dati o soluzioni complete ETL (Extract, Transform, Load);
- **Integrazione orientata a funzioni e metodi:** è di solito integrazione di applicazioni (A2A), e può essere diretta, con paradigma request/response o basata su strumenti di middleware o su codice custom, sviluppato ad hoc;
- **Integrazione di interfacce utente:** è la standardizzazione delle interfacce utente entro un unico modello, di solito basata sul browser (interfacce web), si parla infatti di enterprise business portal o enterprise application portal;
- **Integrazione dei processi business:** agisce direttamente al livello dei processi business ed è la più efficiente da un punto di vista funzionale, ma non è facile da applicare quando vi sono prodotti software con logica business rigida; per essere flessibile, conduce implicitamente alla Service-Oriented Architecture.

## Motivazioni

I sistemi informativi in azienda si sviluppano in maniera incrementale ed evolutiva ed è facile trovare scenari di compresenza di sistemi differenti e complementari che utilizzano molteplici tecnologie, che agiscono su svariati processi aziendali. Questa frammentazione è però in contrasto con una visione ed una gestione integrate che dovrebbero permettere la corretta conduzione di un'azienda. Affinché l'utilizzo delle informazioni presenti nei sistemi informativi aziendali sia efficiente, è necessario che il modello implementato dai sistemi sia consistente e veritiero (divario percettivo) e che i dati siano consistenti e corretti: per realizzare questo risultato (**circolarità delle informazioni e integrità dei dati**) è necessario un processo di integrazione, pulizia e trasformazione che può richiedere molto tempo e molte risorse.

## Processo di integrazione

Esistono due tipologie dei processi di integrazione:

- **Progetti di integrazione:** integrazione dei sistemi informativi aziendali
- **Progetti master data management:** integrazione dei soli dati di maggiore rilevanza

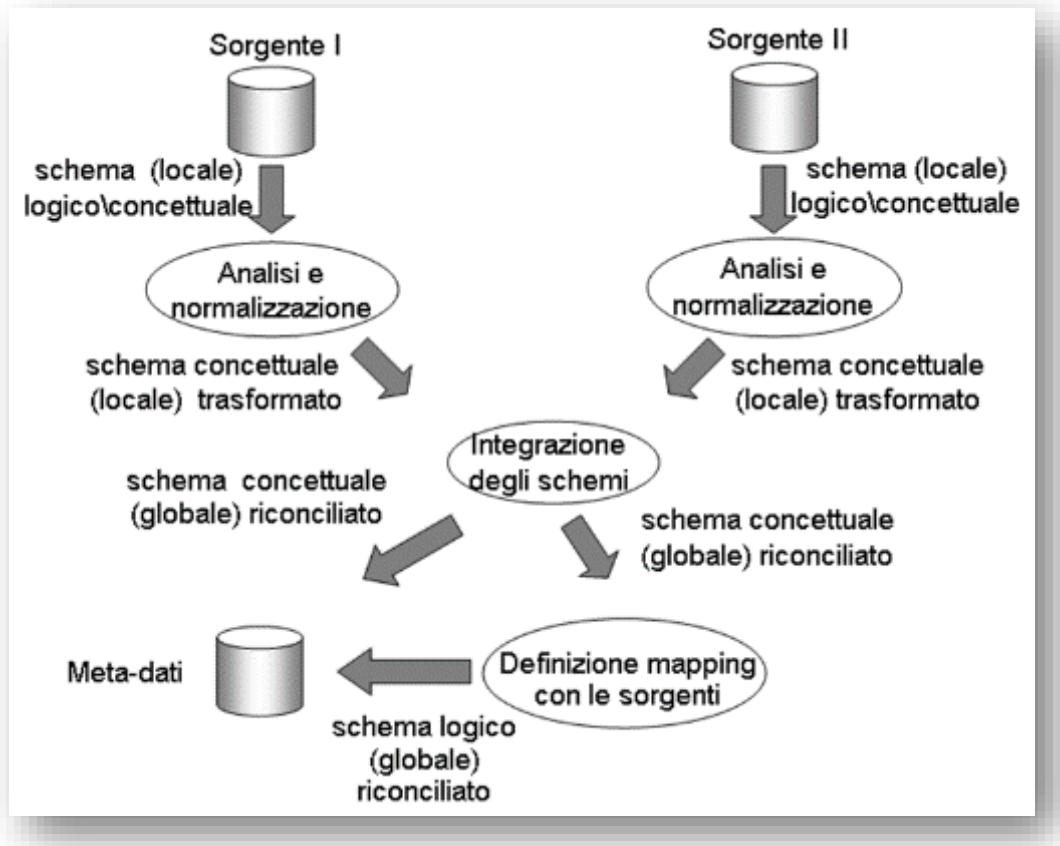
Le attività previste in un processo di integrazione dati sono:

- **Integrazione dello schema:** operazione intensionale che rende consistenti gli schemi dei diversi moduli
- **Trasformazione dei dati:** operazione estensionale che trasforma i dati degli schemi locali nei dati dello schema globale
- **Pulizia dei dati:** operazione estensionale che controlla ed elimina eventuali inconsistenze ed errori.

## Integrazione dello schema

Le fasi di integrazione dello schema prevedono:

- **L'analisi dei diversi schemi locali:** è un'attività di normalizzazione per produrre un insieme di schemi locali completi e consistenti
- **La definizione dello schema riconciliato** rappresenta la fase più importante: i diversi schemi locali vengono fusi in un unico schema globalmente consistente
- **La definizione del mapping** permette di dettagliare le relazioni tra i concetti degli schemi sorgenti e quelli dello schema riconciliato.



## Architettura per il Sistema Informativo Integrato

L'integrazione può essere vista come un livello intermedio tra i diversi sistemi e le interfacce e lo strato della logica applicativa: le applicazioni operano su viste dello schema integrato che può essere:

- **Materializzato:** i dati sono trasformati e memorizzati in una sorgente duplicata (es. Datawarehouse), principalmente per motivi di efficienza di utilizzo dei dati
- **Virtuale:** i dati rimangono negli schemi sorgente e viene definita una mappatura utilizzata dalle applicazioni per reperire e utilizzare le informazioni degli schemi locali effettuando un'integrazione “online”, viene creato un middleware, si tratta di una soluzione molto utilizzata (sistemi informativi distribuiti, federati, eterogenei).

Non è sempre possibile effettuare un'integrazione completa in un unico schema globale, in certi casi è sufficiente integrare solo alcune informazioni strategiche e di particolare importanza: il **Master data management** limita l'integrazione ai soli dati critici.

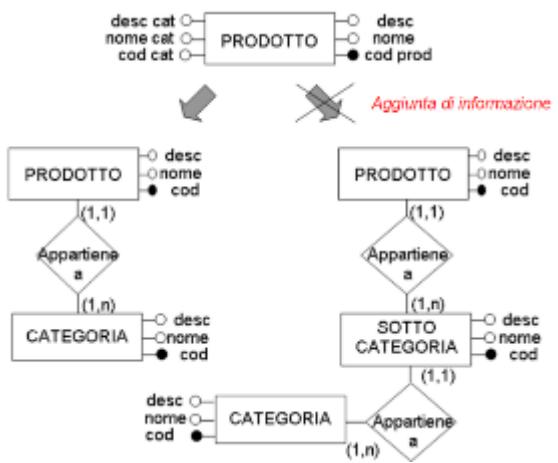
### Analisi e normalizzazione

La fase di analisi prevede lo studio degli schemi, il confronto con gli esperti del dominio applicativo, la ricerca delle correlazioni mancanti individuando le dipendenze funzionali non presenti o nuove associazioni tra le entità. Non si devono introdurre nuovi concetti ma solo esplicitare quelli ricavabili dai dati memorizzati.

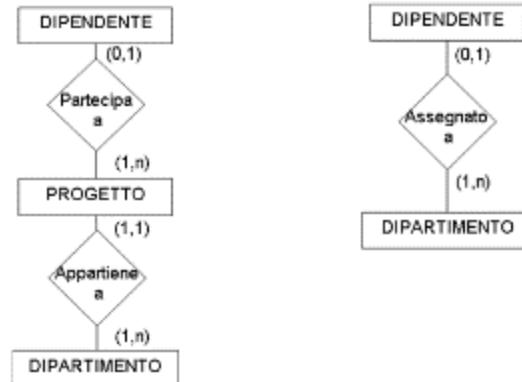
Si tratta di un'operazione di **comprendere della semantica di ogni entità della struttura dati**.

Si deve realizzare uno schema concettuale di tutte le sorgenti utilizzando uno stesso formalismo, eventualmente utilizzando tecniche di reverse engineering.

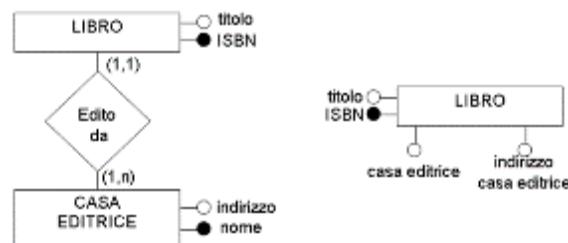
In linea teorica, se esistessero sorgenti “disgiunte” ovvero sorgenti che modellano porzioni indipendenti e distinte del mondo reale, il problema dell'integrazione non esisterebbe. In realtà ogni sistema informativo modella porzioni del dominio applicativo in funzione di processi che condividono parte delle informazioni. È necessario effettuare un tentativo di normalizzazione rispetto alla semantica e rispetto alla struttura dati, tenendo conto che diverse rappresentazioni possono nascere da diverse cause.



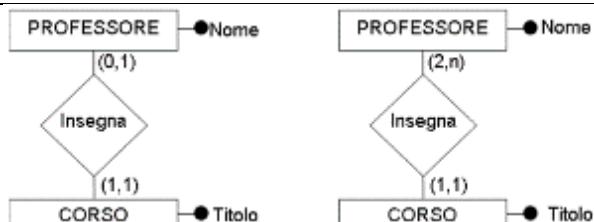
**Diversità di prospettiva**, diversi modelli possono risolvere diversamente lo stesso problema perché affrontato da persone diverse, in momenti diversi, in scenari differenti.



**Equivalenza di costrutti del modello**: concetti modellati utilizzando combinazioni differenti degli elementi di un formalismo possono tuttavia rappresentare una stessa idea definendo differenze puramente sintattiche.



**Incompatibilità delle specifiche**: elementi rappresentati in modelli diversi appaiono in contrasto tra loro. Tuttavia tale problema risiede solitamente nel fatto che vengono utilizzati stessi nomi per concetti diversi.



## Concetti comuni

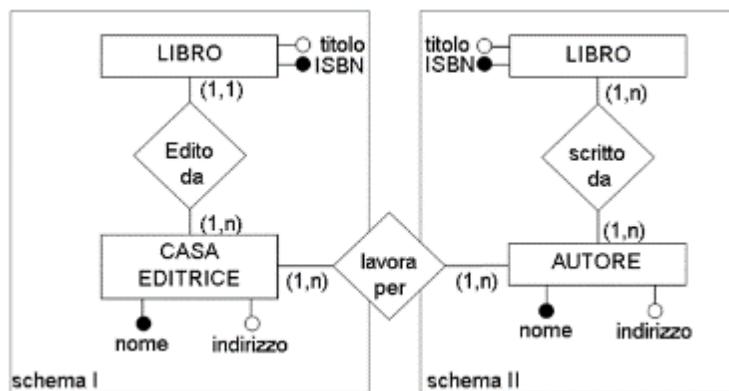
Uno stesso concetto può essere rappresentato diversamente in differenti schemi. Date due rappresentazioni  $R_1$  e  $R_2$  di uno stesso concetto, si definiscono quattro possibili relazioni tra esse:

- **Identità:**  $R_1$  e  $R_2$  coincidono; vengono utilizzati gli stessi costrutti, il punto di vista è lo stesso e non vengono commessi errori di specifica.
- **Conflitti:**
  - ✓ **Equivalenza:**  $R_1$  e  $R_2$  non sono uguali in quanto contengono costrutti diversi, ma non sono presenti errori di specifica né si verificano difformità di percezione.
  - ✓ **Comparabilità:**  $R_1$  e  $R_2$  non sono né identici né equivalenti, tuttavia i costrutti e i punti di vista realizzati non sono in contrasto tra loro.
  - ✓ **Incompatibilità:**  $R_1$  e  $R_2$  sono in conflitto in quanto le specifiche sono incoerenti;  $R_1$  nega  $R_2$ .

La fase di integrazione si occupa di risolvere i conflitti rilevati.

## Concetti correlati

L'integrazione può far emergere alcune proprietà inter-schema che devono essere esplicite, queste rappresentano concetti che non erano percepibili osservando gli schemi in forma isolata.

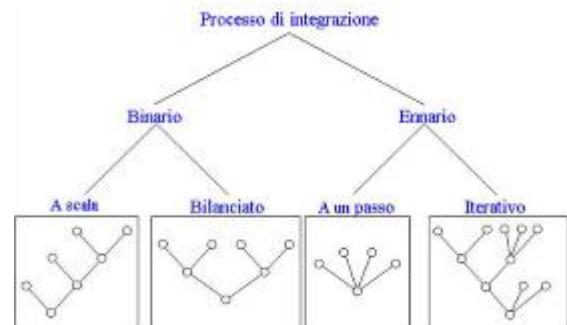


L'integrazione affronta tutti i concetti comuni in situazione di conflitto e applica una strategia per risolverli, inoltre fonde gli schemi dei diversi sistemi inserendo i concetti correlati che generano nuove proprietà.

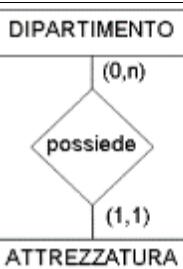
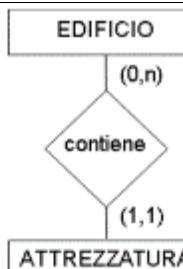
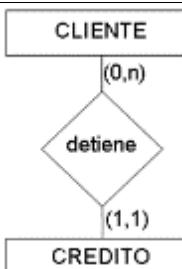
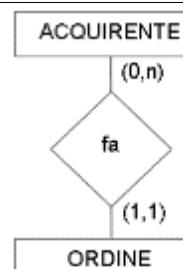
## Fasi dell'integrazione

Le attività descritte richiedono l'esecuzione di un complesso insieme di operazioni, per cui è necessario adottare una metodologia; le tante esistenti e proposte in letteratura concordano su una sequenza di passi:

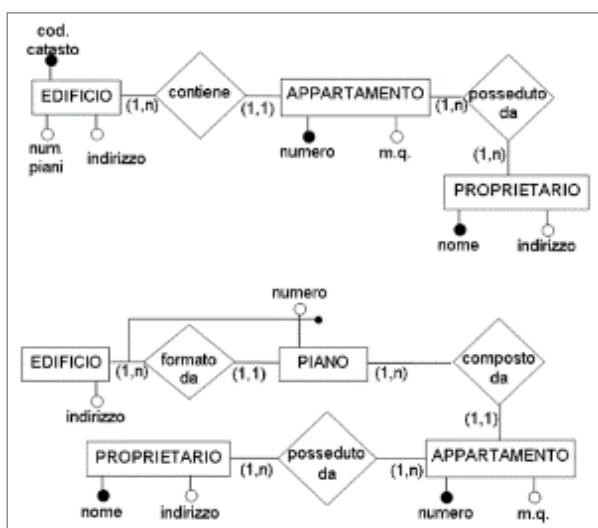
- **Pre-integrazione**, comprende:
  - ✓ analisi
  - ✓ scelta delle strategie di integrazione
    - **Tecniche ennarie:** presuppongono la disponibilità contemporanea di tutte le informazioni da tutte le basi di dati per ogni concetto da integrare, il processo è molto complesso
      - A un passo
      - iterativo
    - **Tecniche binarie:** semplificazione del processo, divide-et-impera su sistemi e basi di dati complessi ed articolati
      - Bilanciato
      - A scala: utilizzato quando esiste una base di dati di riferimento di particolare rilevanza



- **Comparazione degli schemi:** analisi comparativa e ricerca di conflitti e concetti correlati nei vari schemi locali:
  - ✓ **Conflitti di eterogeneità:** discrepanze dovute all'utilizzo di formalismi con diverso potere espressivo negli schemi locali
  - ✓ **Conflitti sui nomi:** differenze nelle terminologie utilizzate nei diversi schemi locali

Omonimie	Sinonimie
stesso termine utilizzato per denotare concetti diversi	nomi diversi denotano uno stesso concetto (l'individuazione di sinonimie richiede una conoscenza approfondita del dominio applicativo)
 	 

- ✓ **Conflitti semantici:** discrepanze dovute a modellazioni differenti (a differenti livelli di astrazione o dettaglio) di una stessa porzione del mondo reale



- ✓ **Conflitti strutturali:** differenze nelle modellazioni di uno stesso concetto o nell'applicazione di diversi vincoli di integrità.
  - **Conflitti di tipo:** stesso concetto modellato con costrutti diversi
  - **Conflitti di dipendenza:** due o più concetti correlati con dipendenze diverse
  - **Conflitti di chiave:** utilizzo di identificatori diversi per uno stesso concetto
  - **Conflitti di comportamento:** diverse politiche di cancellazione/modifica dei dati adottate per un stesso concetto.

- **Allineamento degli schemi:** risoluzione dei conflitti precedentemente individuati attraverso:

- ✓ Cambio dei nomi
- ✓ Cambio dei tipi degli attributi
- ✓ Modifica delle dipendenze funzionali
- ✓ Modifica dei vincoli esistenti sugli schemi

Le risoluzioni devono evitare la perdita di informazioni (es. intero vs float => float; uno-a-uno vs uno-a-molti => uno-a-molti, ...). In caso di conflitti è preferibile mantenere le modellazioni degli schemi che si ritengono

più centrali nella struttura del futuro schema riconciliato, utilizzando una strategia binaria a scala (migliore in questi casi) iniziando l'integrazione dagli schemi più importanti. In caso di conflitti non risolvibili è necessario un confronto con gli utenti.

- **Fusione e ristrutturazione degli schemi:** creazione di un unico schema globale partendo dai diversi schemi resi coerenti. La fusione si basa su questi principi:

- ✓ **Completezza:** l'integrazione può far emergere proprietà inter-schema in precedenza non rilevate, in questo caso è necessario esplicitare tali proprietà inserendo nuove associazioni o gerarchie
- ✓ **Minimalità:** dalla sovrapposizione degli schemi può verificarsi l'esistenza di concetti duplicati o derivabili l'uno dall'altro, oppure relazioni cicliche; tali fattispecie devono essere semplificate in fase di fusione e ristrutturazione
- ✓ **Leggibilità:** è una nozione puramente qualitativa, il miglioramento della leggibilità facilita e svelcisce le successive fasi di progettazione.

### Esempio di integrazione schemi

Dati due schemi come da seguente tabella...

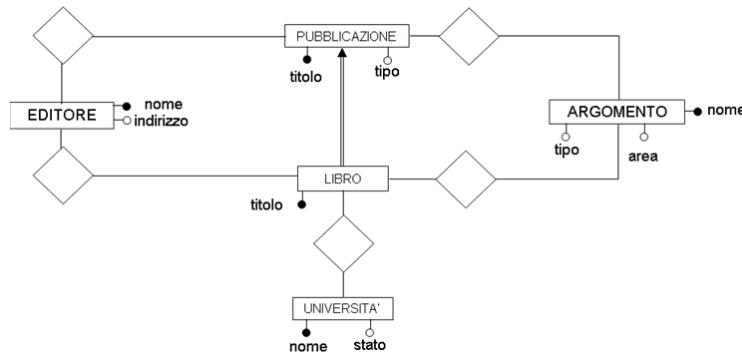
<p>Il primo schema riguarda i libri: i libri hanno un titolo. Sono pubblicati da editori che hanno un nome e un indirizzo. I libri sono adottati da Università che hanno un nome e appartengono a uno Stato. Ogni libro è relativo ad alcuni argomenti [...]</p>	<pre>     graph LR       EDITORE -- titolo --- LIBRO       LIBRO -- nome --- UNIVERSITA_       LIBRO -- stato --- UNIVERSITA_   </pre>
<p>Il secondo schema riguarda 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 [...]</p>	<pre>     graph LR       PUBBLICAZIONE -- titolo --- LIBRO       PUBBLICAZIONE -- tipo --- LIBRO       PUBBLICAZIONE -- nome --- PAROLE_CHIAVE       PUBBLICAZIONE -- area --- PAROLE_CHIAVE   </pre>

... è possibile identificare alcuni concetti comuni e/o correlati:

- Editore (nel primo schema è un'entità, nel secondo un attributo)
- Parole chiave e argomenti rappresentano probabilmente lo stesso concetto (seppure con qualche attributo differente)

<ul style="list-style-type: none"> <li>• Un libro (nel primo schema) è una specifica tipologia di pubblicazione (secondo schema), vi sono pertanto le stesse relazioni con editori e argomenti</li> </ul>	

Uno schema integrato potrebbe essere rappresentato in questo modo:



### Definizione del mapping

Il **mapping** è un insieme di corrispondenze tra elementi presenti negli schemi sorgenti e quelli dello schema destinazione; costituisce il secondo elemento – assieme allo schema riconciliato – risultante dall’analisi delle sorgenti e dell’integrazione.

Le relazioni descritte nel mapping sono utilizzate ogni volta che è necessario eseguire un’interrogazione, espressa nel modello descritto dallo schema riconciliato, sui database locali.

### Global-As-View (GAV)

La modalità GAV prevede che ad ogni concetto dello schema globale debba essere associata una vista definita in base ai concetti modellati negli schemi sorgenti.

Questa modalità riduce l’estensibilità dello schema riconciliato (ogni aggiunta/modifica prevede una revisione di tutti i concetti dello schema globale che impattano sull’oggetto modificato), tuttavia semplifica le definizioni delle interrogazioni in quanto è sufficiente utilizzare le sostituzioni definite nelle viste per operare rispetto agli schemi locali (*unfolding*).

Esempio:

SCHEMI SORGENTI	→	SCHEMA RICONCILIATO
// DB1 Magazzino <b>ORDINI2001</b> <code>(chiaveO, chiaveC, data ordine, impiegato)</code> <b>CLIENTE</b> <code>(chiaveC, nome, indirizzo, città, regione, stato)</code>		<b>CREATE VIEW CLIENTE AS</b> <code>SELECT CL1.chiaveC, CL1.nome,</code> <code>CL1.indirizzo, CL1.città,</code> <code>CL1.regione, CL1.stato, CL2.tel,</code> <code>CL2.fatturato</code> <code>FROM DB1.CLIENTE AS CL1, DB2.CLIENTE AS CL2</code> <code>WHERE CL1.chiaveC = CL2.chiaveC;</code>
// DB2 Amministrazione <b>CLIENTE</b> <code>(chiaveC, piva, nome, tel, fatturato)</code> <b>FATTURE</b> <code>(chiaveF, data, chiaveC, importo, iva)</code> <b>STORICOORDINI2000</b> <code>(chiaveO, chiaveC, data ordine, impiegato)</code>		<b>CREATE VIEW ORDINI AS</b> <code>SELECT * FROM DB1.ORDINI2001</code> <b>UNION</b> <code>SELECT * FROM DB2.STORICOORDINI2000;</code>

### Local-As-View (LAV)

Secondo questa modalità, lo schema globale è descritto indipendentemente dalle sorgenti: i concetti degli schemi sorgenti vengono descritti “retroattivamente” come viste sullo schema globale.

Questa modalità richiede trasformazioni complesse (query rewriting) per identificare gli elementi degli schemi sorgenti da considerare, favorisce l’estensibilità e la manutenzione (es. l’aggiunta di una nuova sorgente richiede la sola definizione della vista sullo schema globale).

Esempio:

SCHEMA RICONCILIATO	→	SCHEMI SORGENTI
// DB Riconciliato <b>ORDINI</b> (chiaveO, chiaveC, data ordine, impiegato)  <b>CLIENTE</b> (chiaveC, piva, nome, indirizzo, città, regione, stato, tel, fatturato)		// DB1 Magazzino <b>CREATE VIEW CLIENTE AS</b> <b>SELECT chiaveC, nome, indirizzo, città, regione, stato</b> <b>FROM DB.CLIENTE;</b>  <b>CREATE VIEW ORDINI2001 AS</b> <b>SELECT * FROM DB.ORDINI</b> <b>WHERE data &gt; '31/12/2000' and data &lt; '1/1/2002'</b>  // DB2 Amministrazione <b>CREATE VIEW CLIENTE AS</b> <b>SELECT chiaveC, piva, nome, fatturato</b> <b>FROM DB.CLIENTE;</b>  <b>CREATE VIEW STORICOORDINI2000 AS</b> <b>SELECT * FROM DB.ORDINI</b> <b>WHERE data &gt; '31/12/1999' and data &lt; '1/1/2001'</b>

#### *Pulizia dei dati*

L’insieme delle operazioni necessarie a garantire consistenza e correttezza dei dati presenti nel livello riconciliato è descritto dal termine **pulizia dei dati (data cleaning, data cleansing, data scrubbing)**.

#### *Cause di inconsistenza e non correttezza*

**Errori di battitura:** possibili se non vengono previste procedure di controllo e validazione degli input

**Differenza dei formati per lo stesso campo:** si verifica se l’informazione non è strutturata

*Esempi: I, IT, Italia; oppure: The Coca-Cola Company, Coca Cola, Coca-Cola Co.*

**Inconsistenza tra valori e descrizione dei campi:** generazione di contenuti inconsistenti, rispetto alla descrizione dei campi, a seguito dell’evoluzione dell’operatività aziendale o delle abitudini per cui un’informazione necessaria non prevista viene memorizzata in maniera inappropriata

**Inconsistenza tra valori di campi correlati:** valori singolarmente corretti ma inconsistenti tra loro

*Esempio: Città = Bologna; Regione = Lazio*

**Informazioni mancanti:** campi non obbligatori, secondari, poco legati all’operatività possono determinare mancanza di informazioni

**Informazioni duplicate:** informazioni uguali fornite da due sistemi sorgenti ma per record che utilizzano chiavi diverse

#### *Strategie di verifica e correzione*

#### **Tecniche basate su dizionari**

Per verificare la correttezza dei valori di un campo è possibile basarsi su tabelle di riferimento (look-up table) e dizionari per ricercare sinonimi e abbreviazioni. Si tratta di tecniche solitamente applicabili quando il dominio è conosciuto e limitato. Possono essere applicate anche a più campi contemporaneamente (per valutare inconsistenze tra campi correlati, es: città e regioni).

Quando è necessario combinare dati provenienti da sorgenti diverse in un’unica destinazione, in assenza di una chiave comune, è necessario effettuare un’operazione **join approssimato**, basato su un sottoinsieme di campi comuni, che tuttavia non garantisce certezza dell’efficacia dell’attività.

Quando è necessario fondere insieme due istanze di uno schema si parla di **purge/merge problem**: due istanze non disgiunte e record duplicati in una o in entrambe, determinano un problema di unicità dei dati che necessita di un'operazione di deduplicazione.

### Similarità tra record

Quando si procede alla valutazione di campi per identificare record duplicati, è possibile utilizzare **tecniche basate su funzioni di similarità** che comparano le sottostringhe di due valori dati calcolandone la similarità in base alla funzione ricorsiva:

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

La comparazione tra i sottocampi elementari si basa sul confronto tra stringhe e sulla verifica di abbreviazioni ricercate in base a pattern ben definiti:

- Un'abbreviazione può essere una sottostringa (Univ. è abbreviazione di Università)
- Un'abbreviazione può combinare un prefisso e un suffisso (Dott.ssa è abbreviazione di Dottoressa)
- Un'abbreviazione può essere un acronimo di una stringa (DEIS è acronimo di Dipartimento di Elettronica Informatica e Sistemistica)
- Un'abbreviazione può essere la concatenazione di più prefissi contenuti nella stringa (UNIBO per Università di Bologna).

È anche possibile utilizzare **tecniche basate su gruppi di regole** che valutano caratteristiche quali, ad esempio, la *edit-distance* tra due stringhe, o la differenza di valore tra campi numerici.

L'idea applicativa è di fondere le coppie di record per le quali il valore di similarità supera una certa soglia.

### Master Data Management (MDM)

Il termine **master data** identifica l'insieme dei dati che descrivono le entità fondamentali per il business di un'azienda, gestiti da varie applicazioni, sia operazionali che analitiche.

Il **master data management (MDM)** è l'insieme delle discipline, delle tecnologie e delle soluzioni che supportano la gestione e il mantenimento della consistenza, l'aggiornamento, l'accuratezza e la completezza dei dati di importanza critica e che forniscono agli utenti, alle applicazioni e ai processi una visione consolidata e unica sia all'interno che all'esterno dell'azienda.

Esistono due aree specifiche:

- **Product Information Management (PIM)**: gestione focalizzata sui Master Data di prodotto
- **Customer Data Integration (CDI)**: gestione focalizzata sui Master Data relativi a clienti, fornitori, utenti

La dispersione delle informazioni cruciali sulle diverse applicazioni provoca problemi di ridondanza, consistenza e inefficienza nei processi aziendali. Questi problemi possono essere gestiti spostando i Master Data “al di fuori” delle singole applicazioni con diverse implicazioni:

- deve essere creata una nuova base di dati “master” sincronizzata con quelle esistenti mediante tecniche di data integration real-time
- i processi di business che *impattano* sui Master Data devono essere modificati per alimentare e sfruttare al meglio la nuova base di dati “master”
- deve essere definita una *ownership* dei Master data e dei processi di alimentazione e gestione relativi.

## Scegliere i Master Data

I Master Data possono essere identificati in funzione di certe caratteristiche:

**Quantità:** la gestione dei Master Data è giustificata solo per grandi quantità di informazioni.

**Lifetime:** i Master Data sono solitamente meno volatili dei dati transazionali.

**Valore:** maggiore è il valore di un'informazione maggiore è la probabilità di considerarla un Master Data.

**Complessità:** potrebbe non essere una buona scelta realizzare Master Data per dati facilmente gestibili.

**Riusabilità:** è molto utile identificare tra i Master Data i dati utilizzati da diversi sistemi aziendali.

**Centralità:** i Master Data sono *centrali* a più applicazioni che li creano, modificano, leggono, cancellano.

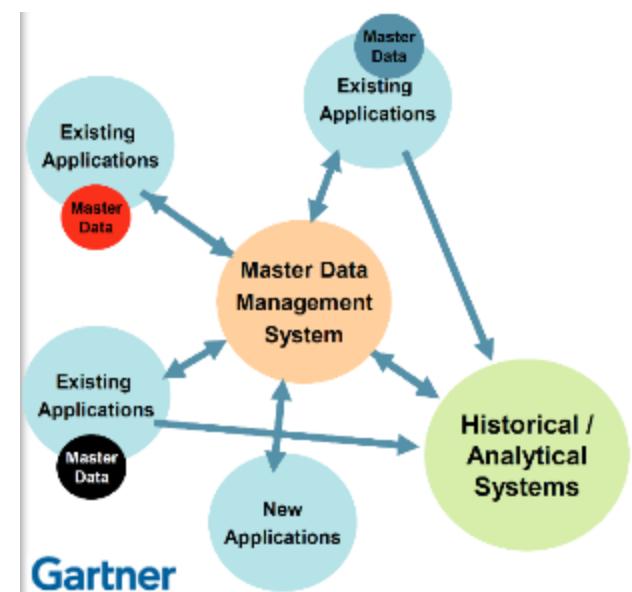
Il Master Data Management, oltre ad essere un problema di natura tecnologica, ha solitamente un impatto significativo sui processi aziendali e determina problemi organizzativi e politici:

- Chi è il proprietario dei dati?
- Chi è responsabile della loro manutenzione, pulizia?

Un progetto di Master Data Management è durevole, iterativo, sempre attivo, non si tratta di un'attività *una tantum* in quanto opera su una parte di Sistema Informativo che deve essere mantenuta nel tempo e che è funzionale al business.

Le fasi principali di un progetto di Master Data Management sono:

1. **Identificare le sorgenti dati:** spesso il numero dei database che memorizzano MD è superiore alle aspettative
2. **Specificare produttori e consumatori di MD:** tipologie di operazioni e numerosità di attori coinvolti nella gestione dei MD sono elementi che incidono sulla complessità della gestione
3. **Raccogliere meta dati sui MD:** per la definizione univoca e formale dei MD a livello sintattico e semantico
4. **Identificare gli esperti di MD:** per ogni sorgente è utile identificare le persone capaci di descrivere come i MD debbano essere trasformati per renderli compatibili secondo lo schema riconciliato
5. **Definire il processo di data-governance e il relativo gruppo di lavoro:** competenze e autorità per prendere decisioni relative alla gestione dei MD
6. **Sviluppare un modello di gestione dei MD:** è, dal punto di vista tecnico, la fase più importante del progetto, definisce:
  - Quale architettura utilizzare
  - Quali formati saranno utilizzati per rappresentare i MD
  - Quali record saranno incluso o esclusi
7. **Scegliere uno strumento:** in funzione della tipologia di MD e delle scelte fatte nella fase precedente
8. **Progettare e implementare l'infrastruttura di gestione:** per collezionare e rendere i MD disponibili alle applicazioni esistenti
9. **Generare e testare i Master Data:** processo evolutivo iterativo che affina progressivamente la soluzione “grezza” considerando e modellando via via tutte le specificità dei MD



### TIPOLOGIE DI DATI

**Non strutturati:** e-mail, documenti pdf, pagine web, ...

**Transazionali:** ordini, fatture, ticket, ...

**Metadati:** descrizione di attributi, glossari, ...

**Gerarchici:** relazioni tra concetti

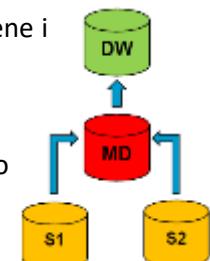
**Master:** persone (clienti, impiegati, fornitori, ...), cose (prodotti, materie prime, locali, immobili, ...), luoghi, concetti, ....

10. **Modificare le applicazioni produttrici e consumatrici di MD:** sfruttando le interfacce disponibili, aderendo alle politiche di gestione delle applicazioni ed eventualmente modificarle
11. **Definire il processo di manutenzione:** per verificare la qualità dei MD ad opera degli esperti, correggere errori; tramite una definizione di ruoli, modalità e tempi di gestione e tramite la previsione di strumenti attuativi. Un eventuale strumento di controllo dovrebbe mostrare le inconsistenze, supportare attività di data tracing e data auditing per identificare i punti sorgenti di incoerenze e suggerire correzioni.

## Architetture per Master Data Management

### *Consolidamento*

Questo approccio prevede di istanziare (fisicamente) un'architettura con un hub centrale che contiene i *golden record*.



- I dati rimangono di proprietà delle applicazioni sorgenti che li gestiscono in autonomia
- L'aggiornamento dei Master Data è asincrono rispetto agli eventi che li creano o li aggiornano
- I Master Data sono utilizzati principalmente a scopo di reportistica
- I Master Data possono essere periodicamente scaricati sulle applicazioni che li utilizzano come versioni di riferimento (vedi architettura di coesistenza)
- I Master Data sono utilizzati per armonizzare il comportamento di più applicazioni senza essere utilizzati durante transazioni nei sistemi sorgenti.

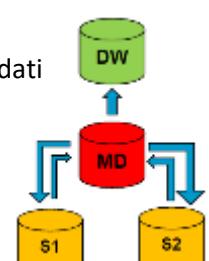
### *Registro*

Questa architettura prevede la costruzione di 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
- I dati rimangono di proprietà delle applicazioni sorgenti che li gestiscono in autonomia
- L'hub esegue algoritmi di pulizia e match tra i record, assegna identificatori univoci ai record, associa versioni corrispondenti nelle diverse applicazioni
- Non vengono inviati aggiornamenti alle sorgenti
- Il miglioramento della qualità è limitato
- Richiede una logica complessa
- Ha un impatto limitato su applicazioni e processi
- I Master Data sono utilizzati per armonizzare il comportamento di più applicazioni senza essere utilizzati durante transazioni nei sistemi sorgenti.

### *Coesistenza*

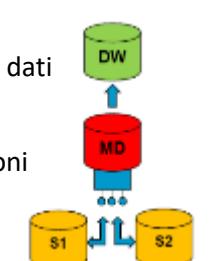
In questo caso è prevista la creazione di un hub centrale che mantiene una versione aggiornata dei dati successivamente riversata sulle sorgenti (in modalità asincrona).



- I dati rimangono di proprietà delle applicazioni sorgenti che li gestiscono in autonomia
- L'aggiornamento dei Master Data è asincrono rispetto agli eventi che li creano o li aggiornano pertanto l'architettura non garantisce che siano aggiornati
- I Master Data sono utilizzati per armonizzare il comportamento di più applicazioni e come punto di riferimento centralizzato

### *Transazionale*

Questo approccio prevede la creazione di un hub centrale che mantiene una versione aggiornata dei dati utilizzata, in modo sincrono, da tutte le applicazioni.



- I dati sono di proprietà dell'hub: è l'unica entità che li modifica sulla base delle segnalazioni effettuate dalle sorgenti
- I dati devono essere resi disponibili in modo sincrono alle sorgenti

- Si sfrutta, solitamente, il meccanismo publish-subscribe dei web service
- Quando possibile, l'hub stesso aggiorna i database degli applicativi sorgenti allo scopo di limitare le modifiche alle applicazioni
- L'hub utilizza politiche transazionali per garantire la coesistenza delle informazioni
- I Master Data sono utilizzati sia nella normale operatività delle applicazioni operazionali sia per armonizzare il comportamento tra più applicazioni.

#### *Esempio: il caso di un'Azienda USL*

Un'Azienda USL gestisce i servizi erogati agli assistiti in differenti sistemi informativi (es. CUP per la prenotazione delle prestazioni ambulatoriali, sistema di gestione dell'assegnazione dei medici di medicina generale, sistema Accettazione-Dimissione-Trasferimenti ricoveri, pronto soccorso, sistema flussi informativi regionali, data warehouse aziendale, ...). Ognuno di questi sistemi utilizza necessariamente un insieme di dati anagrafici degli assistiti che accedono ad un servizio erogato dall'Azienda; gli assistiti possono essere residenti nel territorio di afferenza dell'azienda oppure non residenti che utilizzano i servizi dell'azienda.

Soltanente tutte le operazioni di erogazione di servizi sanitari vengono associate ad un record di anagrafica:

- per comodità, rapidità e riduzione degli errori gli assistiti del territorio di afferenza dell'Azienda sono solitamente presenti nell'anagrafica assistiti dei sistemi che potrebbe essere integrata con i sistemi informativi dei comuni
- le anagrafiche dei pazienti extra territoriali sono solitamente inserite e aggiornate “al bisogno” ovvero quando il paziente accede alla struttura
- alcuni servizi potrebbero dover essere erogati “senza collegamento anagrafico”, ad esempio accessi al PS per pazienti incoscienti o in stato confusionale
- i punti di accesso, anche per uno stesso servizio, potrebbero essere molteplici (diversi CUP, diversi ambulatori, diversi uffici ricoveri, ...).

Le Aziende Sanitarie comunicano i dettagli delle prestazioni erogate a residenti presso altri territori alle Aziende di riferimento e da esse ricevono i dettagli delle prestazioni erogate ai propri residenti attraverso i *flussi di mobilità regionale*.

#### *Situazione attuale*

Si considera **anagrafica principale** quella gestita dal sistema di gestione dell'assegnazione dei medici di medicina generale in quanto la scelta (ed eventuale revoca) del medico di base è un'operazione obbligatoria per tutti e soli i residenti, tale operazione deve essere fatta di persona e vale anche per tutti gli assistiti che non ricevono prestazioni sanitarie.

Tale anagrafica viene aggiornata mensilmente e resa disponibile tramite rete intranet in formato “csv”, tuttavia:

- non esiste una logica unificata per la fusione degli archivi
- gli archivi dei sistemi informativi degli altri servizi sono spesso più aggiornati in quanto è più facile che un assistito riceva una prestazione sanitaria piuttosto che revocare o cambiare il proprio medico di base.

Ogni applicativo ha una propria struttura relazionale con campi e formati diversi.

Inoltre, per gli assistiti extra-territoriali, fanno fede le informazioni raccolte al primo contatto (in ogni servizio/applicativo) tramite tessera sanitaria e riconoscimento tramite documenti.

I flussi regionali in entrata sono sempre in ritardo di un mese.

#### *Diagnosi situazione attuale*

- i. Siccome è possibile inserire nuovi assistiti in ogni applicativo, è possibile effettuare errori e duplicazione di dati anagrafici con la conseguenza di limitare la capacità di analisi delle prestazioni effettuate da un paziente
- ii. Le eventuali incoerenze nella raccolta dati generano problematiche operative, ad esempio un assistito che effettui una prenotazione CUP potrebbe ritrovarsi, in caso di inserimento dati errati:

- a. Senza prenotazione (se il codice sanitario è errato)
  - b. Impossibilità a ricevere comunicazioni e risultati tramite posta se l'indirizzo è errato o cambiato
  - c. Impossibilità a ricevere comunicazioni e variazioni delle prenotazioni se il numero di telefono è errato o cambiato.
- iii. Solo gli aggiornamenti effettuati nell'applicativo di gestione dei medici di base vengono poi diffusi a tutti gli applicativi
  - iv. Ogni applicazione deve prevedere una propria procedura di fusione dei dati
  - v. Il livello di affidabilità delle applicazioni è estremamente vario:
    - a. L'anagrafica del servizio di scelta/revoca MMG è spesso obsoleta e mancante dei dati degli assistiti extra territoriali
    - b. Il Pronto Soccorso è il reparto con i dati meno attendibili
    - c. Il CUP è l'ufficio con i dati più aggiornati in quanto la maggiore parte delle prestazioni parte da una prenotazione.
  - vi. La frequenza (mensile) degli aggiornamenti non è sufficiente.

#### Soluzione coesistenza

- ✓ Si costruisce un nuovo database per i master data che raccoglieranno tutte le informazioni comuni lasciando le informazioni utili solo localmente distribuite nei diversi applicativi.
- ✓ I processi non vengono modificati ma operano su dati di qualità migliore.
- ✓ Si definiscono flussi e interfacce da e verso ogni applicazione: in ogni applicazione dovrà essere implementata una procedura per l'integrazione dei dati in ingresso.
- ✓ Si definiscono la struttura standard dei dati e le regole di standardizzazione.
- ✓ Si definiscono le politiche di pulizia dei dati
- ✓ Si definisce una politica di merge dei dati presso l'HUB
  - a. Criteri per identificazione dei dati più aggiornati
  - b. Valutazione di informazioni discordanti tramite livelli di affidabilità delle sorgenti
- ✓ Si definiscono modalità e tempistiche di aggiornamento presso le sorgenti.

#### Soluzione transazionale

- ✓ Si costruisce un nuovo database per i master data
- ✓ Si modificano tutte le applicazioni per permettere la gestione transazionale dei dati, ad esempio:
  1. L'utente del CUP *invoca* il web service di sistema di MDM richiedendo i dati anagrafici della tessera sanitaria n. 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 (eventualmente le modifiche possono essere autorizzate solo a specifici uffici/utenti)
  5. Il sistema verifica i dati e accetta o rifiuta l'aggiornamento
  6. Il lock viene rilasciato e il dato diventa visibile a tutti.

#### Differenze tra le due soluzioni

Coesistenza	Elementi di valutazione	Transazionale
Minore	Impatto sulle applicazioni	Maggiore
Possibile	Presenza di applicazioni legacy	Impossibile o molto complesso
Non necessario	Sistemi in rete	Necessario (eventuale utilizzo di cache locali)
Necessario, minore impatto	Definizione dell'owner di gestione dei MD	Necessario (eventuale supervisione)

Basso	Allineamento e circolarità delle informazioni	Elevato
No	Centralizzazione gestione dati e politiche di aggiornamento	Si

Da discutere con il prof...

<https://ec.europa.eu/digital-single-market/en/news/recommendation-european-electronic-health-record-exchange-format>

## Seminario: Big Data

Con il termine **data science** si intende l'estrazione di informazioni e conoscenza dai dati. Utilizza teorie e tecniche largamente utilizzate nelle aree di studio della statistica dell'information technology.

Il **data scientist** è un esperto di queste aree con una forte attitudine nella comprensione del business e dei dati. Il data scientist è in grado di trasformare le informazioni nascoste nei dati in vantaggio competitivo: il suo obiettivo finale è di creare nuovi modelli di business nella cosiddetta **data-driven economy**.

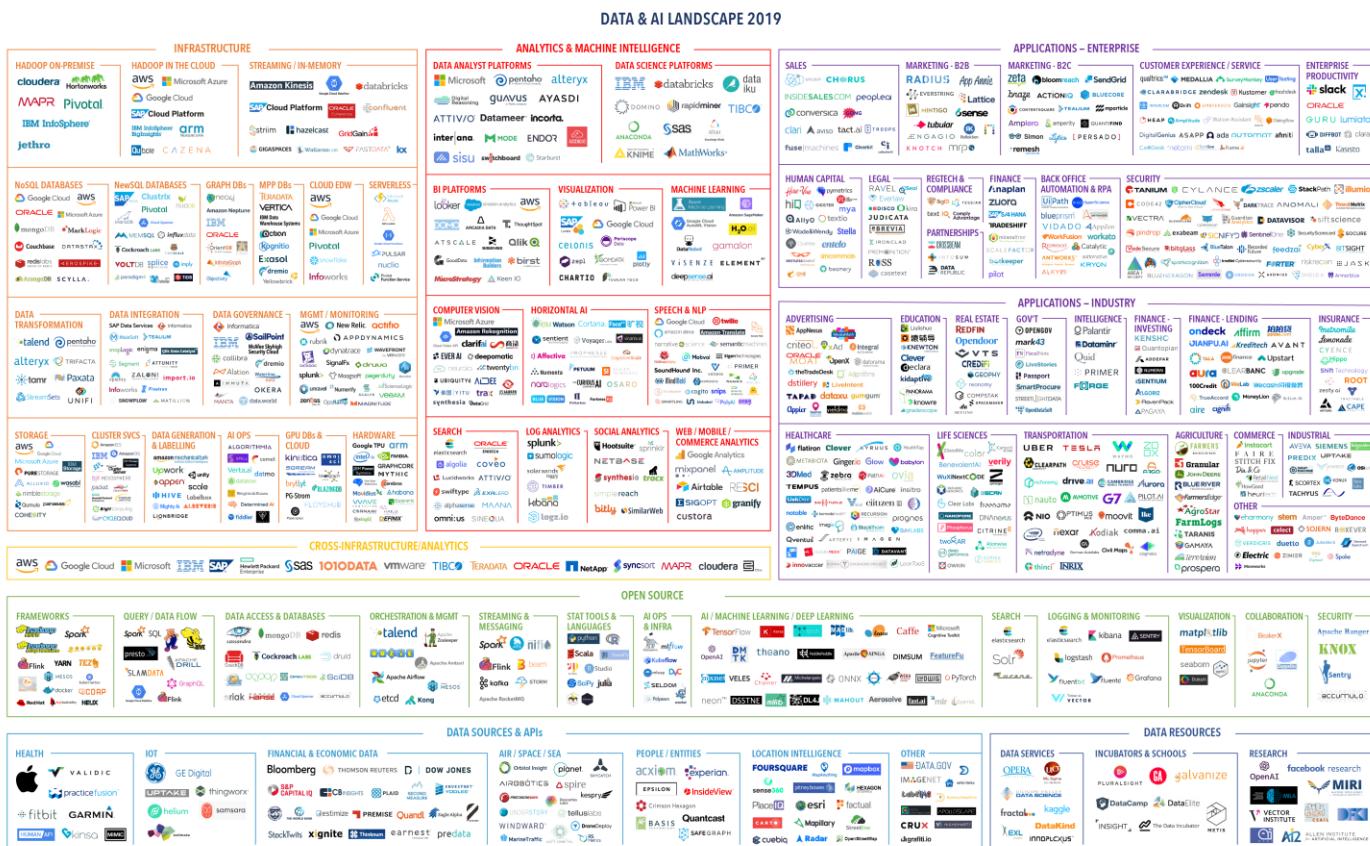
La data science è un'evoluzione della **Business Intelligence** che si prefigge di superare alcune limitazioni:

- utilizzo di dati strutturati (come per la BI) ma anche non strutturati
- implementazione di processi di estrazione flessibili, agili, estemporanei anziché rigidi e complessi ETL
- data scientists e data enthusiasts implementano e sfruttano le tecniche di data science mentre la BI è implementata da ingegneri informatici e sfruttata da utenti e analisti
- la BI utilizza DB e DBMS relazionali, i progetti di data science sfruttano tecnologie BigData e NoSQL.

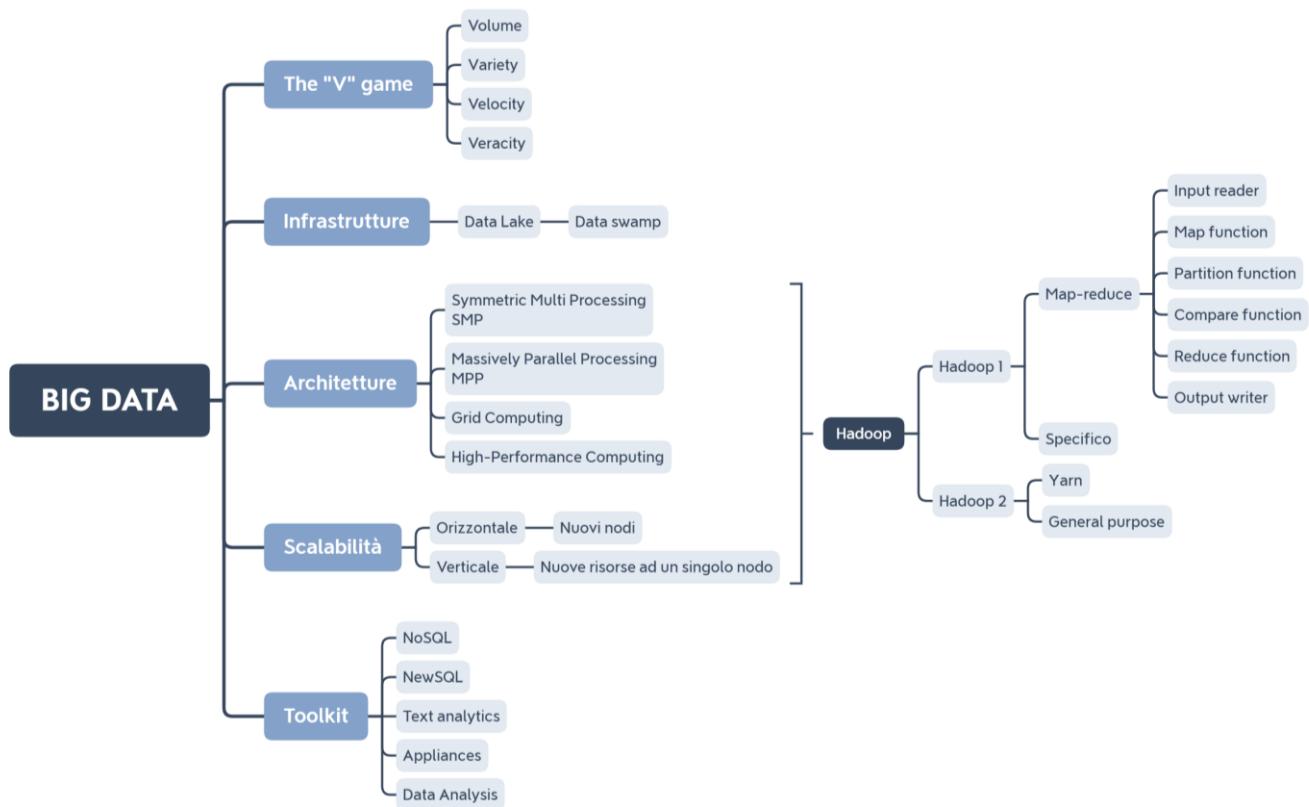
La nascita e l'evoluzione della Data Science è legata principalmente alla disponibilità di **grandi dataset** (contenuti generati dagli utenti, Internet of Things, Smart Cities, Smart Grid), di **nuove tecnologie** dedicate alla manipolazione e gestione di grandi moli di dati, di elaborazioni statistiche, di gestione di informazioni distribuite (Hadoop, NoSQL, R, ...) e all'evoluzione di **tecniche di analisi dati specifiche** (Text mining, Natural Language Processing, Data Mining, Explorative Analysis, ...).

Se la maggior parte di tecniche correlate ai Big Data è già nota, le tecnologie sono nuove e richiedono uno studio e una padronanza che genera un gap culturale, colmare questo gap potrebbe richiedere diversi anni e, addirittura, la creazione di una nuova generazione di managers.

<http://dfkoz.com/ai-data-landscape/>



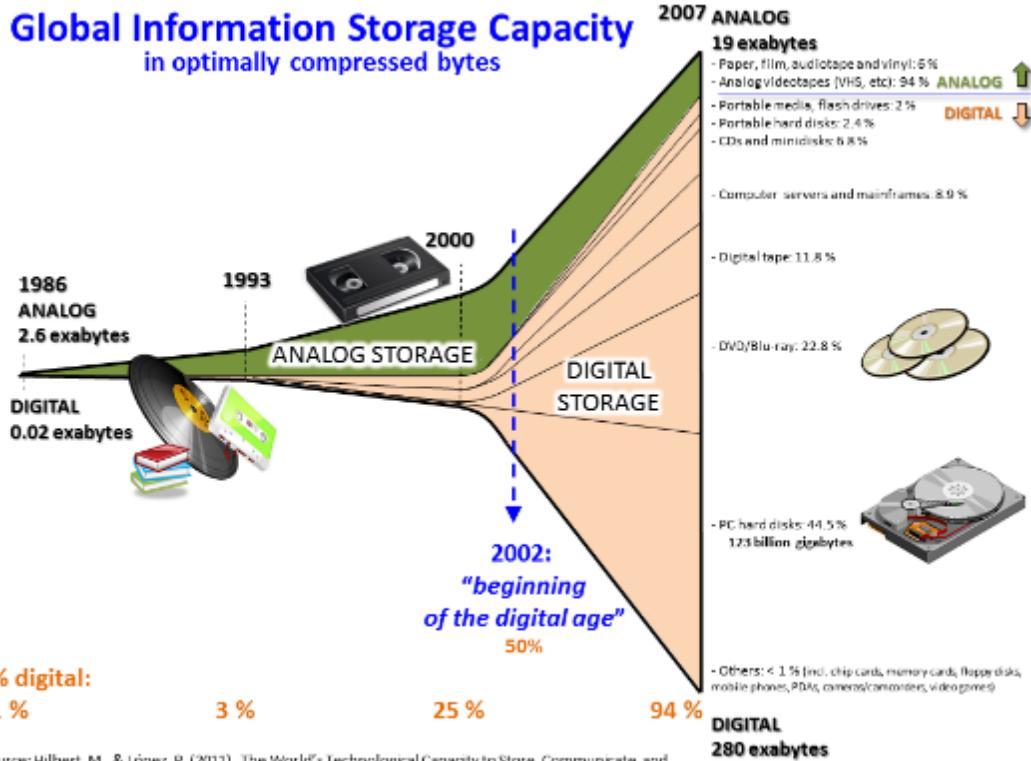
## Definizione di Big Data



I Big Data sono insiemi di dati caratterizzati da una serie di caratteristiche, solitamente descritte da termini che iniziano con la lettera V (non necessariamente tutti).



<b>VOLUME</b>	<p>Quantità di dati generati e memorizzati. La quantità di dati determina il valore e la potenziale capacità di approfondimento.</p> <table border="1"> <thead> <tr> <th>TeraByte</th><th>PetaByte</th><th>ExaByte</th><th>ZettaByte</th></tr> </thead> <tbody> <tr> <td><math>10^{12}</math></td><td><math>10^{15}</math></td><td><math>10^{18}</math></td><td><math>10^{21}</math></td></tr> <tr> <td>~ 1 HD</td><td>1000 HD</td><td>1 million of HD</td><td>1 billion of HD</td></tr> </tbody> </table>	TeraByte	PetaByte	ExaByte	ZettaByte	$10^{12}$	$10^{15}$	$10^{18}$	$10^{21}$	~ 1 HD	1000 HD	1 million of HD	1 billion of HD
TeraByte	PetaByte	ExaByte	ZettaByte										
$10^{12}$	$10^{15}$	$10^{18}$	$10^{21}$										
~ 1 HD	1000 HD	1 million of HD	1 billion of HD										
<b>VARIETY</b>	<p>Tipologia e natura dei dati. I big data spaziano da testi a immagini, audio, video, ... possono essere strutturati, semi-strutturati, non strutturati e permettono di colmare l'assenza di informazioni attraverso la loro fusione: aiutano le persone che analizzano i dati nella comprensione.</p>												
<b>VELOCITY</b>	<p>Velocità con la quale i dati vengono generati ed elaborati per soddisfare le esigenze e le sfide di crescita e sviluppo. I big data sono spesso disponibili in tempo reale, vengono prodotti continuamente. Due tipologie di velocità relative ai big data sono: la <i>frequenza di generazione</i> e la <i>frequenza di gestione, registrazione e pubblicazione</i>.</p>												
<b>VERACITY</b>	<p>Misura della qualità e del valore dei dati. Si tratta di una caratteristica aggiuntiva e supporta l'accuratezza delle analisi.</p>												



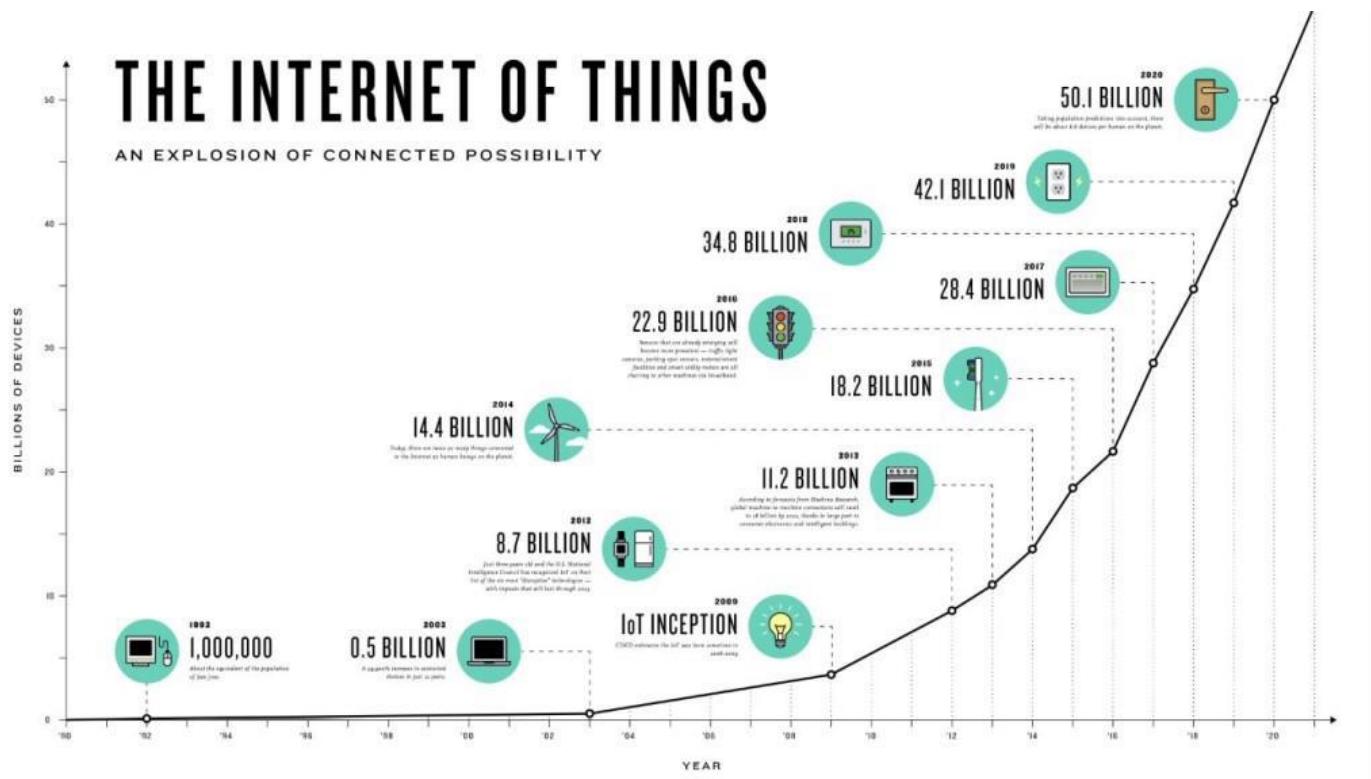
By Myworkforwiki - Own work, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=29452425>

## 2019 This Is What Happens In An Internet Minute

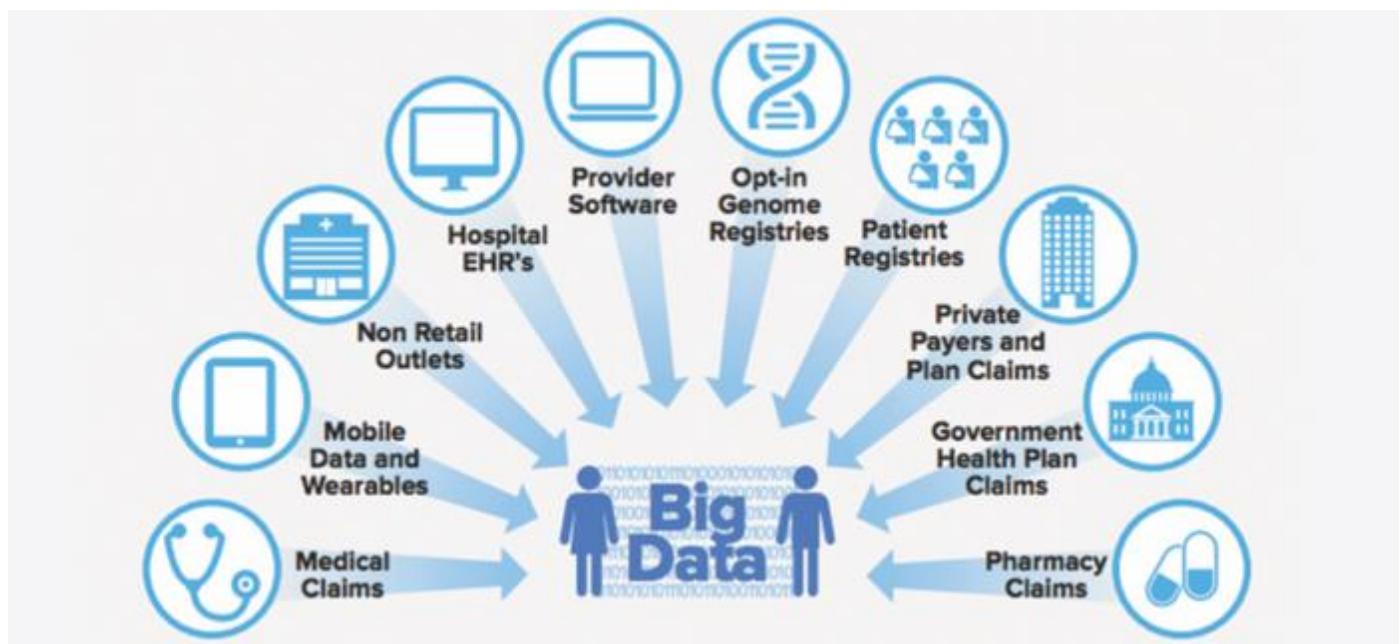


## Chi genera big data?

L'Internet of Things è rappresentata dall'utilizzo evoluto di Internet, in cui gli oggetti acquisiscono una qualche forma di intelligenza: si rendono riconoscibili (identità), possono raccogliere informazioni su se stessi (coscienza), sull'ambiente locale in cui si trovano (conoscenza) e su altri oggetti connessi (relazioni) e possono effettuare elaborazioni sui dati (analisi, trasformazione, collegamento) e comunicare attraverso la rete.



Naturalmente, oltre alla IoT, esistono svariate fonti di generazione di dati, ad esempio nel mondo healthcare i dati sono generati (e spesso correlati) tramite sistemi informativi ospedalieri (EHR), app per pazienti o professionisti, sistemi informativi per gli assistiti (fascicolo sanitario), registri di patologia, assicurazioni, sistemi informativi governativi (flussi ministeriali), aziende sanitarie e fornitori, pharma, ...



## Un'infrastruttura per i dati

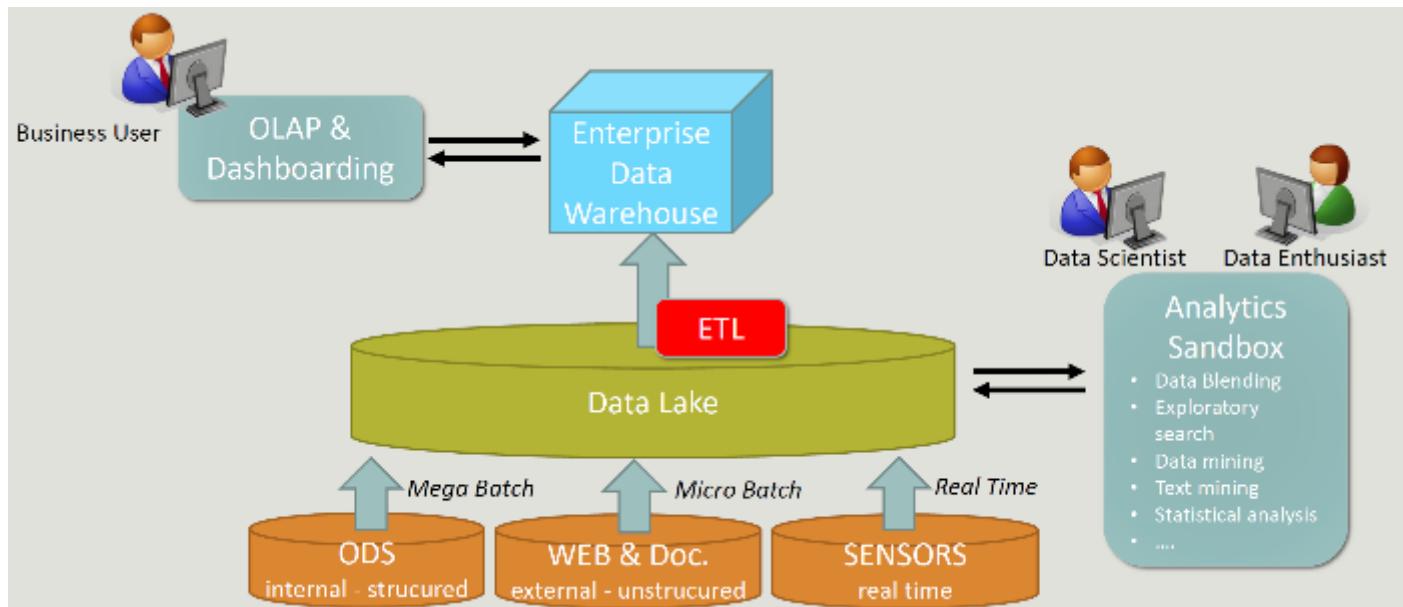
Il termine **data lake** si intende un sistema o un repository nel quale i dati sono memorizzati nel loro formato naturale, solitamente oggetti “blob” oppure file (può includere dati strutturati come database relazionali, dati semi-strutturati come file VSC, log, XML, JSON, ... o dati non strutturati come email, documenti, file pdf, ... e dati binari come immagini, file audio, video, ...). Un data lake è solitamente un singolo contenitore per tutti i dati aziendali, incluse copie grezze dei dati sorgenti e dati trasformati utilizzati per attività come il reporting, la visualizzazione, l'analisi avanzata e il machine learning.

Il termine **data swamp** si usa per identificare data lake deteriorato e non gestito inaccessibile agli utenti e incapace di fornire valore. James Dixon, CTO di Pentaho, ha presumibilmente coniato il termine in contrasto al termine *data mart*, che rappresenta piccoli repository derivati da dati grezzi che presentano problemi quali l’“information siloing”. PricewaterhouseCoopers ritiene che i data lakes potrebbero risolvere questo problema; tuttavia esistono diverse modalità controverse per gestire i big data e non è chiaro se i data lakes siano effettivamente un'iniziativa di successo. La sfida principale non risiede nella creazione di data lakes, bensì nell'ottenere un vantaggio dalle opportunità che essi presentano.

In un data lake i dati dovrebbero essere memorizzati per il tempo utile: è pertanto necessario individuare anche regole e metodologie per identificare i dati da eliminare, scartare. Questo anche per il fatto che gestire moli di dati in continuo aumento determina verosimilmente una decadenza delle performance (per questo motivo sono utilizzate innovative tecniche di gestione dei dati). In ogni caso i data lakes supportano analisi molto avanzate.

Un “analytics sandbox” è una piattaforma che permette agli analisti di condurre ricerche e *situational analysis*<sup>1</sup>; l'utilizzo è destinato agli analisti e a *power users* ai quali fanno riferimento i vari gruppi aziendali per ottenere informazioni e risposte.

Il data lake raccoglie quindi i dati prodotti dalle diverse fonti con modalità (interni, social e web, IoT, sensori, ...) e tempi differenti (batch, real time, ...) e popola il DataWarehouse aziendale attraverso script ETL per la realizzazione di analisi OLAP e dashboard per gli utenti di business; contemporaneamente fornisce i dati per la piattaforma di analytics sandbox.

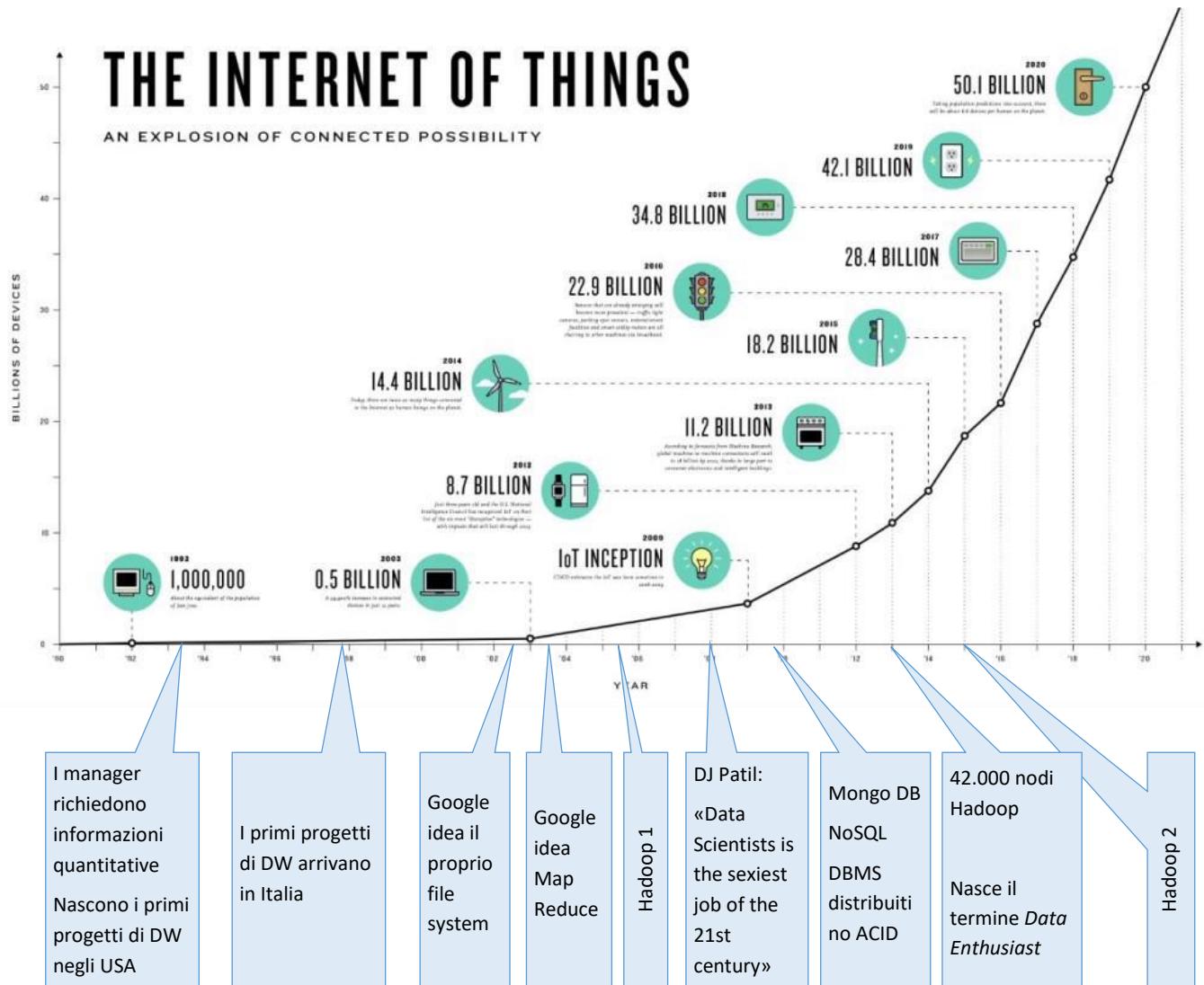


L'infrastruttura brevemente descritta rappresenta una possibile soluzione applicativa per progetti di data science che nascono e derivano da una serie di fattori abilitanti questa rivoluzione: sempre maggiore bisogno di informazioni e

<sup>1</sup> Situation analysis è una collezione di metodi utilizzati dai manager per analizzare l'ambiente interno ed esterno di un'organizzazione, per comprenderne le capacità e le sfide da gestire con i competitor, i clienti, i collaboratori e l'ambiente di lavoro (5C analysis, SWOT analysis, ...).

sempre maggiore richiesta di tempestività per garantire reazioni efficaci, sempre maggiore disponibilità di dati (IoT), sempre maggiore disponibilità di potenza computazionale (tecnologie).

## Evoluzione dai DataWareHouse ai Big Data



## Hadoop

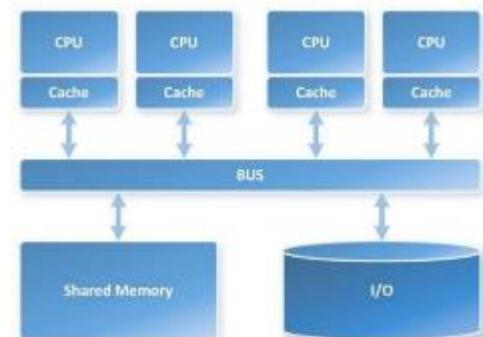
La libreria software Hadoop è un framework che permette di processare grandi dataset in modalità distribuita attraverso clusters di computer, utilizzando semplici modelli di programmazione. Hadoop è progettato per adattarsi ad architetture costituite da singoli server a migliaia di macchine ognuna dotata di capacità computazionale e spazio di memorizzazione dati. Piuttosto che fare affidamento sull'hardware per garantire alta disponibilità e affidabilità, la libreria è progettata per individuare guasti nello strato applicativo allo scopo di fornire un servizio di alta disponibilità complessiva su un cluster di computer che possono essere soggetti singolarmente a guasti.

### Architetture a confronto

#### Architettura SMP

I RDBMS tradizionali si basano sull'architettura Symmetric Multi Processing (SMP) nella quale diversi processori condividono la stessa RAM, lo stesso bus I/O e lo stesso array di dischi.

*Il limite principale di questa architettura dipende dal numero di dispositivi fisici che possono essere montati e dal collo di bottiglia generato dal BUS.*



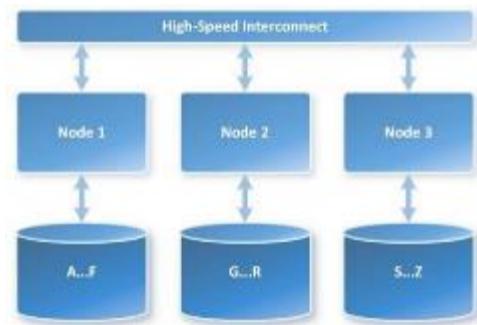
### Architettura MPP

In un'architettura Massively Parallel Processing (MPP), diversi processori equipaggiati con una propria RAM e propri dischi collaborano a risolvere singoli problemi suddividendoli in multipli tasks indipendenti. Questa architettura è anche detta *shared nothing architecture*.

Alcuni esempi: Teradata, Nettezza, ...

#### Hadoop vs MPP

In MPP l'elaborazione dei dati è distribuita sui diversi nodi per un'elaborazione parallela, i risultati vengono assemblati per produrre una soluzione finale, tuttavia i sistemi che sfruttano MPP sono basati su HW di alto livello e software proprietari.



Hadoop, invece, si basa su HW comune e software open source; è nativamente controllato tramite codice imperativo mentre le appliances MPP sono interrogate attraverso query dichiarative.

D'altra parte SQL è più semplice e produttivo rispetto alla realizzazione di codice MPP o job MapReduce e i professionisti di database sono più abbondanti e meno costosi di specialisti Hadoop.

### Architettura Grid Computing

L'architettura Grid Computing prevede l'utilizzo di una collezione di risorse di calcolo da diverse posizioni per raggiungere un unico obiettivo. Si distingue dalle classiche architetture di cluster ad alta performance in quanto:

- ogni nodo esegue diversi task o applicazioni
- i computer sono eterogenei e geograficamente dispersi
- pur potendo dedicare una grid ad una singola applicazione, normalmente una grid è utilizzata per vari scopi differenti
- le grid sono spesso realizzate con computer general-purpose e librerie di software per il middleware.

### Architettura HPC

Il termine High-Performance Computing indica sistemi paralleli specificatamente dedicati alla risoluzione di tasks CPU-intensive:

- spesso HPC viene utilizzato come sinonimo di Super Computer anche se questo è un termine più generale che definisce principalmente un sistema con capacità di elaborazione contemporanea la cui architettura può cambiare nel tempo
- le principali applicazioni HPC sono simulazioni scientifiche, previsioni meteorologiche, modellazione 3D, ...
- in molti casi i processori condividono lo stesso insieme di dischi.

### Scalabilità

La **scalabilità** è la capacità di un sistema, una rete, un processo di gestire una crescita della quantità di lavoro in modo appropriato o di accrescere le proprie capacità per rispondere a questa crescita.

I metodi per aggiungere risorse ad una particolare applicazione appartengono a due ampie categorie:

- **horizontal scaling**: aggiunta di nodi ad un sistema, ad esempio aggiunta di computer ad un'applicazione software distribuita
- **vertical scaling**: aggiunta di risorse ad un singolo nodo del sistema, tipicamente CPU o memoria.

## NoHadoop: non solo Hadoop

Il *toolkit* del data scientist include molte tecnologie, per la memorizzazione e la gestione dei dati, che sono nate attorno all'idea che non esiste uno strumento valido per tutti gli scopi.

NoSQL (Not Only SQL)	NewSQL	Text analytics	Appliance	Data Analysis
MongoDB	VoltDB	Elastic Search	Oracle Exadata & Exalytics	Alteryx
Neo4J		Oracle Endeca	SAP HANA	R
Hbase, BigTable			Teradata Aster	Tableau
Redis			SAS LASR Analytic Server	Brandwatch, Traxx

## Hadoop 1 vs Hadoop 2

Hadoop 1 (es. Map-reduce) è nato per gestire alcuni tasks molto specifici, era difficile da programmare, robusto ed efficiente per molti jobs ma non adatto per l'elaborazione online.

Hadoop 2 (es. Yarn) è nato con lo scopo di rendere Hadoop un framework più general-purpose:

- Map-reduce è solo uno dei possibili paradigmi di programmazione
- Supporta SQL
- È adatto per l'elaborazione online
- La stessa comunità di sviluppo cerca di rendere più semplice l'utilizzo per gli utenti non esperti:
  - ✓ Realizzando console grafiche di controllo
  - ✓ Realizzando connettori per molti software esterni (es. OLAP Microstrategy).

## Distribuzioni principali

Molte aziende forniscono prodotti che includono Apache Hadoop, o suoi derivati, supporto commerciale e/o strumenti e utility relativi.

**Cloudera** offre una piattaforma di progetti open-source chiamata Cloudera's Distribution che include Apache Hadoop o CDH. Inoltre Coludera offre ai suoi clienti enterprise una famiglia di prodotti e servizi che completano la piattaforma Hadoop; quest includono sessioni formative, servizi architetturali e supporto tecnico.

**Hortonworks** offre una piattaforma Hadoop 100% open-source.

**MapR** vende un framework map-reduce ad alte performance basato su Apache Hadoop che include molti componenti dell'eco-sistema standard.

**IBM InfoSpher BigInsight** “porta il potere di Apache Hadoop alle aziende”.

In molti casi Hadoop non è sfruttato da un cluster locale ma tramite un provider in cloud; questa scelta può essere determinata da diversi fattori:

- Costi
- Completezza del market place
- Altri progetti “hosted” sul cloud che producono dati
- ...

## Un'analisi disincantata di Hadoop

Hadoop è un sistema distribuito che rispetto alle precedenti architetture distribuite rende più semplice la gestione del cluster e la programmazione, si basa su commodity HW e software open source.

---

Il modello di programmazione map-reduce rende semplice l'utilizzo della computazione parallela ma non tutti i problemi possono essere parallelizzati in maniera efficiente su queste architetture, alcuni non sono affatto parallelizzabili.

Hadoop è nato nell'era della cloud computing: i fornitori di servizi cloud spingono per l'utilizzo di Hadoop e Big Data dal momento che richiedono molta capacità computazionale e abbondante spazio disco che possono essere acquistati in cloud.

Hadoop non è più efficiente delle architetture centralizzate (es. ORACLE DBMS) or appliance (es. Teradata), ma è in grado di adattare le proprie caratteristiche ai diversi scopi a prezzi contenuti.

Hadoop è la risposta delle grandi aziende di informatica (es. Google) per le attività di data modeling, memorizzazione e gestione dei dati:

- favorisce la programmazione alla modellazione rispetto ai tradizionali sistemi di database
- richiede un cambiamento radicale nel modo in cui i dati vengono modellati e trasformati
- impone un paradigma di programmazione con un insieme limitato di funzionalità

### Il paradigma Map-Reduce (esempio)

Map-reduce è un modello di programmazione (e un'implementazione) per processare e generare insiemi di big data tramite un algoritmo distribuito e parallelo in un cluster di calcolatori.

Un programma Map-reduce è costituito da:

- una procedura o metodo “map” che esegue azioni di filtro e ordinamento
- una procedura o metodo “reduce” che esegue azioni di aggregazione (conteggio, calcolo frequenze, ....)

Il dataflow può essere sintetizzato in:

#### 1. Input reader

Divide l'input, da uno storage tipicamente distribuito, in parti di dimensioni appropriate (64MB o 128MB) e assegna ogni parte ad una diversa funzione Map generando coppie chiave/valore.

#### 2. Funzione Map

Processa una serie di coppie chiave/valore generando zero o più coppie chiave/valore in output.

#### 3. Funzione partition

Ogni output della funzione Map è allocato ad uno specifico reducer dalla funzione partition dell'applicazione, questa funzione prende in input la chiave e il numero dei reducers e restituisce l'indice del reducer desiderato. Tra gli stadi di map e reduce, i dati vengono mischiati (ordinati, scambiati tra i nodi) allo scopo di muoverli dal nodo che effettua il mapping a quello che dovrà eseguire la reduce; talvolta questa attività richiede molto tempo essendo dipendente dalla bandwidth della rete e dalla velocità delle CPU.

#### 4. Funzione compare

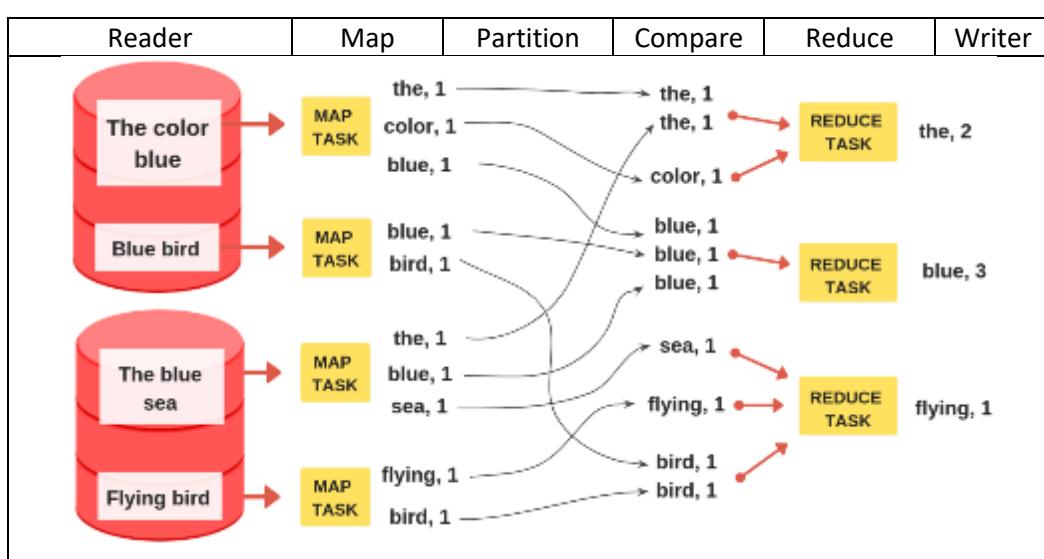
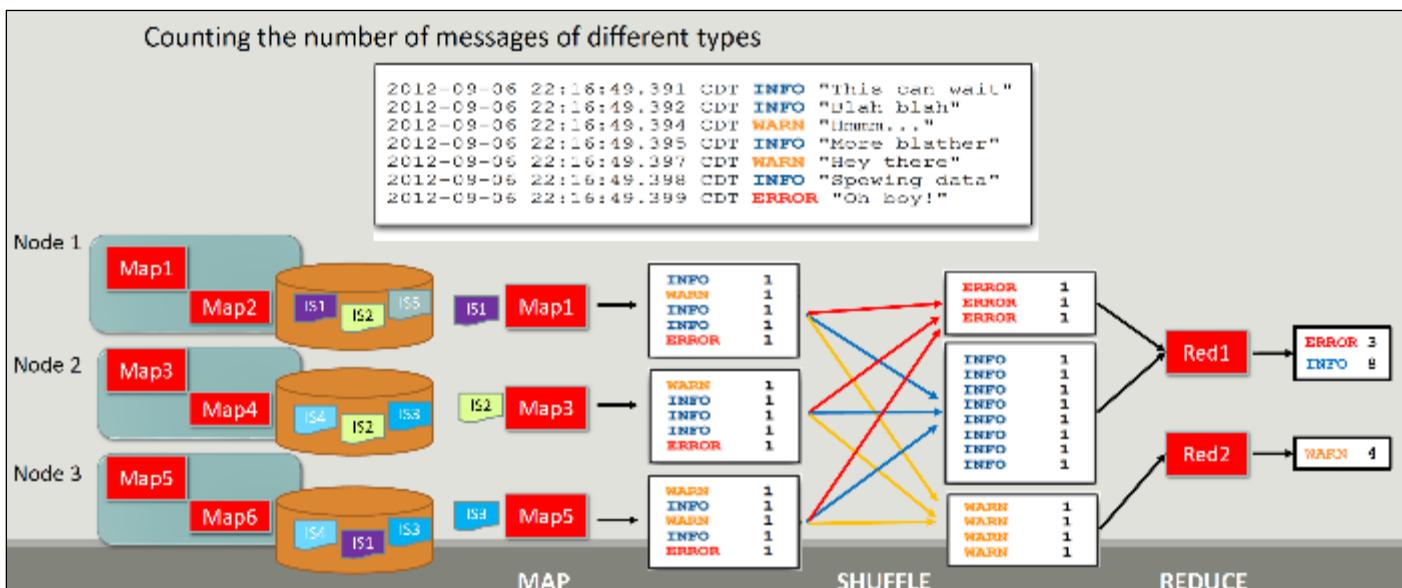
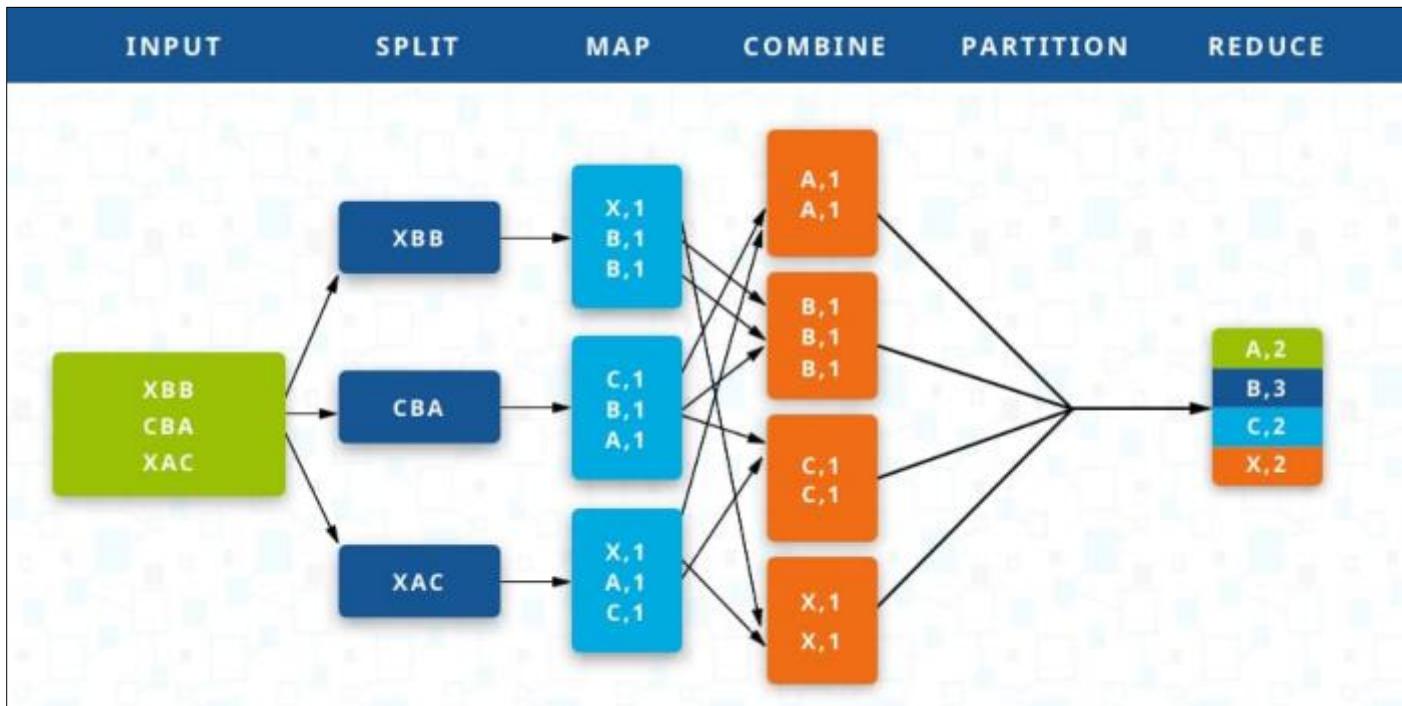
L'input di ogni reduce è preso dalla macchina nella quale è stata eseguita la funzione di Map e ordinato utilizzando la funzione compare dell'applicazione.

#### 5. Funzione Reduce

Il framework invoca la funzione reduce dell'applicazione una volta per ogni chiave unica secondo l'ordinamento effettuato. La reduce può iterare per elaborare i valori che sono associati alla stessa chiave e può produrre zero o più output.

#### 6. Output writer

Il risultato viene scritto su memoria di massa stabile.

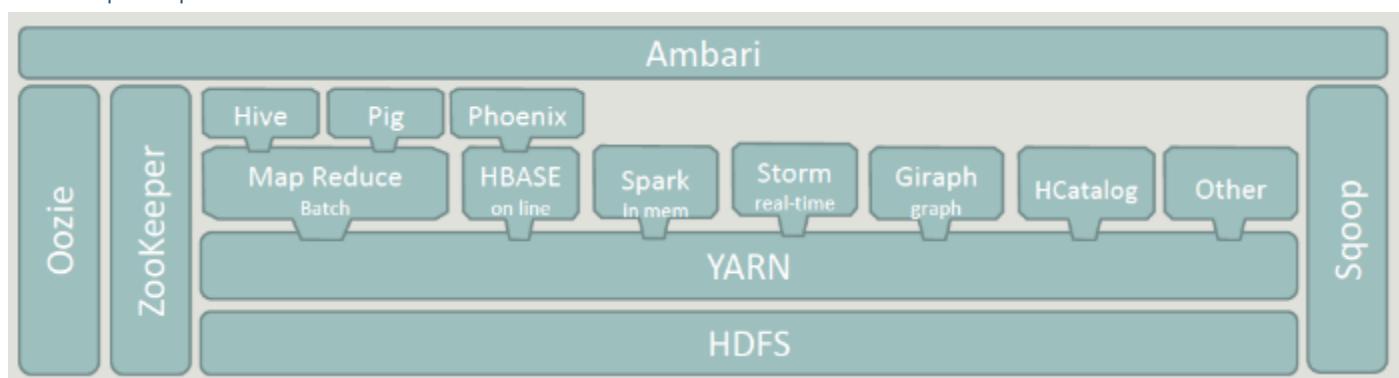


## Hadoop 1 vs Hadoop 2

La versione 1.0 di Hadoop diventa disponibile a Novembre 2012, la versione 2.2 quasi un anno dopo, mentre si è dovuto attendere la fine del 2017 per la versione 3.0.

Hadoop 1	Hadoop 2
Supporta unicamente modelli di elaborazione <b>MapReduce (MR)</b>	Permette di lavorare con <b>MR e altri modelli di computazione distribuita</b> come Spark, Hama, Giraph, MPI, HBase coprocessors
MR effettua sia le <b>elaborazioni</b> che la <b>gestione</b> del cluster	La gestione delle risorse del cluster viene effettuata tramite <b>YARN (Yet Another Resource Negotiator)</b> mentre le elaborazioni sono effettuate tramite i diversi modelli di computazione
La scalabilità è limitata a <b>4.000</b> nodi per cluster	La scalabilità è incrementata a <b>10.000</b> nodi per cluster
Utilizza il concetto di <b>slot</b> : gli slot possono eseguire sia un'attività di map che di reduce	Utilizza il concetto di <b>container</b> : un container può eseguire task generici
Utilizza un <b>singolo Namenode</b> per gestire l'intero namespace.	<b>Multipli server Namenode</b> gestiscono multi namespaces
Ha un <b>Single-Point-of-Failure (SPOF)</b> a causa del singolo Namenode e in caso di guasto richiede un intervento manuale	È configurato per un <b>ripristino automatico</b> in caso di guasto: per superare SPOF utilizza un Namenode in standby pronto in caso di guasto
Le <b>API MR sono compatibili</b> in tutta la major Hadoop1x.	<b>Sono necessari file aggiuntivi</b> per eseguire le API MR per un programma scritto in Hadoop 1x
Esistono <b>limitazioni nell'utilizzo come piattaforma</b> per elaborazione di eventi, streaming ed operazioni real-time.	Può essere utilizzato come piattaforma per servire una <b>larga varietà di analytics</b> : processazione di eventi, streaming, operazioni realtime
<b>Un guasto al Namenode impatta su tutto lo stack.</b>	<b>Lo stack Hadoop è equipaggiato per gestire guasti nel Namenode</b>
Non supporta <b>Microsoft Windows</b>	È stato aggiunto il supporto per <b>Microsoft Windows</b>

## Moduli principali



<b>Common</b>	Utility comuni che supportano gli altri moduli
<b>Distributed File System (HDFS)</b>	File system distribuito che fornisce accesso ai dati con elevati throughput
<b>YARN</b>	Un framework per schedulare i job e gestire le risorse del cluster
<b>Map Reduce</b>	Un sistema basato su YARN per l'elaborazione parallela di grandi insiemi di dati
<b>Ambari</b>	Strumento web-based per fornire, gestire e monitorare i cluster Apache Hadoop, include un supporto per HDFS, Map Reduce, Hive, HCatalog, HBase, ZooKeeper, Oozie, Pig e Sqoop
<b>ZooKeeper</b>	Servizio di coordinazione ad alte performance per applicazioni distribuite
<b>Oozie</b>	Schedulatore di workflow per gestire i job di Apache Hadoop, i jobs di Oozie sono graphi aciclici direzionali (Directed Acyclical Graphs, DAGs) di azioni
<b>Sqoop</b>	Strumento progettato per effettuare trasferimenti massivi di dati tra Hadoop e datastore strutturati (DB relazionali) in maniera efficiente
<b>Pig</b>	Piattaforma per l'analisi di grandi insiemi di dati, include un linguaggio di alto livello (Pig-Latin) per esprimere programmi di analisi dati. Opera in maniera simile a flussi ETL per trasformare i dati. I programmi Pig-Latin sono compilati in jobs Map Reduce
<b>Hive</b>	Fornisce un meccanismo per strutturare i progetti basandosi su dati su HDFS, interroga i dati utilizzando un linguaggio SQL-like chiamato HiveQL che permette anche di aggiungere plugin di mappatura e riduzione personalizzate per migliorare le perfomance
<b>HBase</b>	DBMS distribuito non relazionale. Fornisce capacità BigTable-like per Hadoop. Include funzionalità di compressione, operazioni in memoria, tabelle colonnari, schemi flessibili (key/value datastore)
<b>Spark</b>	<p>Spark migliora le performance fino a 100 volte per certe applicazioni rispetto al paradigma Map Reduce. Opera in memoria e permette agli utenti di caricare i dati nella memoria del cluster e interrogarli ripetutamente. Adatto particolarmente per algoritmi di machine learning. Inizialmente implementato presso l'AMPLab dell'Università di Berkely nel 2009, consiste di diversi componenti:</p> <ul style="list-style-type: none"> <li><b>Spark SQL</b>, che introduce una nuova astrazione chiamata SchemaRDD che fornisce supporto per dati strutturati e semi-strutturati; fornisce un linguaggio specifico per manipolare tali schemi in Scala, Java o Python, supporto per il linguaggio SQL, interfacce CLI e server ODBC/JDBC</li> <li><b>Spark Streaming</b> eleva la velocità del core di Spark per eseguire analytics in streaming: acquisisce dati in mini-batches ed effettua trasformazioni RDD</li> <li><b>Mlib</b> è un framwork per il machine learning distribuito basato su Spark</li> <li><b>GraphX</b> è un framework di elaborazione distribuita grafico basato su Spark, fornisce API per realizzare computazioni che possono modellare l'astrazione Pregel</li> </ul>

<b>Storm</b>	Sistema per la computazione distribuita real-time per elaborare grandi volumi di dati ad alta velocità, utile in scenari di real-time analytics, machine learning e monitoraggio continuo
<b>Giraph</b>	Sistema grafico iterativo costruito per l'alta scalabilità (es. utilizzato da Facebook per analizzare i grafi sociali degli utenti)
<b>HCatalog</b>	Crea un'astrazione relazionale dei dati su HDFS e assicura che l'utente non si preoccupi di dove sono i dati o in quale formato siano memorizzati.
<b>Phoenix</b>	Strato di database relazionale basato su HBase realizzato come un client embedded in JDBC con l'obiettivo di garantire bassa latenza sulle query. Prende una query SQL, la compila in una serie di scansioni HBase e orchestra l'esecuzione delle scansioni per produrre risultati validi JDBC.

## Aree applicative

Miglioramento della sicurezza e delle leggi	Smart Cities	Smart Factories	Miglioramento delle scienze e della ricerca
<ul style="list-style-type: none"> <li>Ricerca e individuazione di transazioni fraudolente</li> <li>Ricerca e prevenzione di attacchi cibernetici</li> <li>Fusione informazioni da sensori per applicazioni cibili e militari</li> </ul>	<ul style="list-style-type: none"> <li>Ottimizzazione del flusso di traffico basata su informazioni in tempo reale (social media, dati meteorologici, ...)</li> <li>Ottimizzazione energetica intelligente</li> </ul>	<ul style="list-style-type: none"> <li>Sistemi SCADA raccolgono milioni di tuple al giorno dagli impianti</li> </ul>	<ul style="list-style-type: none"> <li>Il data center del CERN ha 65.000 processori per l'analisi di 30 petabyte di dati. Utilizza anche la potenza di migliaia di computer distribuiti in 150 data center dislocati in tutto il mondo.</li> </ul>

## Hadoop Distributed File System

Il File System distribuito di Hadoop (HDFS) è progettato per essere eseguito su “commodity hardware”:

- i guasti hardware sono considerati normali anziché eccezioni, per questo motivo la rilevazione dei guasti e il recupero rapido fa parte dell'architettura core dell'HDFS
- le applicazioni in esecuzione su HDFS hanno bisogno di accesso in streaming ai propri dati, HDFS è progettato più per i processi batch che per i processi interattivi per gli utenti: l'enfasi è sugli alti livelli di throughput di accesso ai dati piuttosto che la bassa latenza
- le applicazioni che utilizzano HDFS gestiscono grandi moli di dati (nell'ordine di grandezza dei GigaByte e dei TeraByte), per questo motivo HDFS è regolato per gestire file di grandi dimensioni
- le applicazioni HDFS richiedono un modello di accesso ai file di tipo “write-once-read-many”: una volta che un file viene creato, scritto e chiuso, non si ha bisogno di effettuarne variazioni; questa assunzione semplifica i problemi di coerenza di dati e permette elevati throughput di accesso ai dati
- in generale, una elaborazione di dati richiesta da un'applicazione è maggiormente efficiente se effettuata *in prossimità* dei dati; HDFS fornisce le interfacce necessarie per muovere le applicazioni in prossimità dei dati (data locality).

A seguito del rilascio di YARN in Hadoop 2.0, stanno emergendo nuove applicazioni da eseguire sullo stesso Hadoop cluster su dati in HDFS. Questa gamma di applicazioni ha diversi modelli di accesso ai dati e requisiti si stanno muovendo oltre il semplice batch.

In HDFS 1 tutto i dischi venivano gestiti allo stesso modo in ciascun nodo. HDFS 2 è una tecnologia che permette di sfruttare tutti i dischi e le memorie hardware: dischi, unità a stato solido, RAM e memoria esterna.

L'amministratore di sistema del cluster sarà in grado di configurare il supporto di archiviazione disponibile su ciascuno nodo. HDFS consentirà quindi ai set di dati di ricevere una preferenza per il livello di archiviazione. Le applicazioni saranno in grado di specificare una preferenza per un supporto di memoria durante la creazione di file.

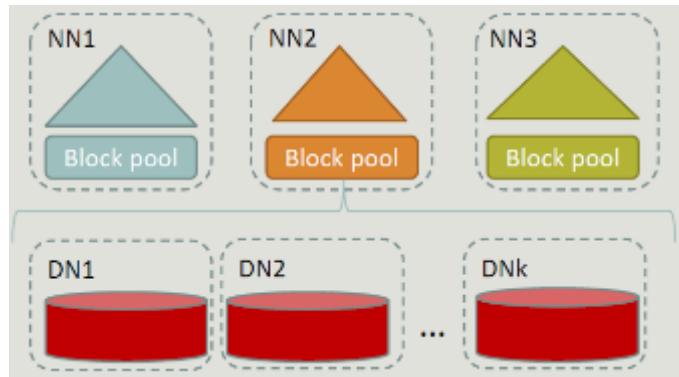
## HDFS Federation

HDFS 2 implementa una “federazione” di namespaces.

Un **name space (NS)** è una gerarchia di file e directory (due differenti NS possono avere file con lo stesso nome in directory con lo stesso nome).

I metadati per ogni NS sono memorizzati in un **Name Node (NN)** che gestisce e mantiene il NS tree e la mappatura dei blocchi di dati (**data blocks**) e dei **Data Nodes (DN)**.

I NS utilizzano blocchi raggruppati in **block pool**. Un DN può fornire blocchi a più di un NS.



Questa metodologia fornisce alcuni vantaggi:

- **scalabilità**: siccome il NN mantiene tutti i NS e le locazioni dei blocchi in memoria, la dimensione dell'heap NN limita il numero di file e il numero di blocchi utilizzabili; limita inoltre la somma della memoria che può essere supportata da un NN
- **performance**: NN è un punto singolo per la memorizzazione e la gestione dei meta-dati: può diventare un collo di bottiglia quando deve gestire un grande numero di file, specialmente un grande numero di piccoli file
- **disponibilità**: è possibile separare i NS di differenti applicazioni incrementando la disponibilità generale del cluster
- **manutenibilità, sicurezza e flessibilità**: l'astrazione block pool permette l'utilizzo dei blocchi di memoria a differenti applicazioni; i NS potrebbero cambiare ed ognuno di essi è isolato e non consapevole dell'esistenza degli altri.

Le applicazioni possono leggere o scrivere su differenti NS.

## Topologie di cluster

Allo scopo di prendere decisioni appropriate, il cluster deve sapere quale topologia è stata definita durante la fase di configurazione. L'allocazione dei processi e i blocchi di storage sono attività che necessitano di conoscere tale topologia.

I nodi sono organizzati in rack a loro volta installati presso data center.

Il modello di Hadoop prevede una organizzazione gerarchica ad albero e calcola la distanza tra i nodi come la loro distanza sull'albero.

## Blocchi di dati e repliche

Il contenuto dei file è suddiviso in grandi blocchi di dati (default 128MB) e ogni blocco è replicato, in maniera indipendente, su molteplici Data Nodes allo scopo di migliorare performance e robustezza. La replica prevede la conoscenza della topologia del cluster. Per ogni blocco di dati il NN aggiorna la lista dei dati in esso memorizzati. Il valore predefinito di replica è 3:

- La copia 1 è memorizzata nel nodo (N1) nel quale il client scrive tramite i relativi comandi
- la copia 2 è memorizzata in un nodo (N2) in un rack diverso da quello in cui è installato il nodo (N1)
- la copia 3 è memorizzata in un nodo, diverso da N2 ma che appartiene al rack in cui è installato il nodo N2.

I blocchi di dati in HDFS sono la più piccola unità di dati in un NN, variano tra i 64MB e 1GB. Blocchi di dati grandi riducono il costo di gestione delle richieste a livello del cluster; in HDFS queste richieste viaggiano attraverso la rete con molto overhead: ogni richiesta deve essere processata dal NN per capire dove sono i blocchi di dati necessari.

Localmente in un data node, una richiesta di blocchi di dati, viene convertita in molte richieste di blocchi a livello del sistema operativo; il sistema operativo compone un blocco di dati accessibile in maniera indipendente e, se un file è più piccolo della dimensione di un blocco di dati viene memorizzato in un file più piccolo sul disco.

## Località dei dati

Hadoop sfrutta la topologia e la replica dei blocchi di dati per applicare il **principio della data locality**:

*“quando una computazione riguarda grandi insiemi di dati, è meno costoso (più veloce) muovere il codice in prossimità dei dati piuttosto che fare il contrario”.*

Il gestore delle risorse preferisce i seguenti casi secondo l'ordine di seguito riportato:

1. Processi e dati nello stesso nodo
2. Processi e dati in diversi nodi dello stesso rack
3. Processi e dati in differenti rack dello stesso data center
4. Processi e dati di differenti data centers.

## Namespace e Namenode

I metadati del name space, detti **immagini**, includono l'albero del name space, la mappatura dei blocchi nel data node e gli **Inode** contenenti attributi quali i permessi, le date e gli orari di modifica e accesso, i name space e la quota di spazio disco.

Il name node memorizza le variazioni nell'HDFS in **log write-ahead** detto **journal**.

Un name node secondario si connette regolarmente con quello primario e costruisce uno snapshot delle informazioni delle directory del name node primario. Questa *immagine di check-point* può essere utilizzata per riavviare un name node primario guasto senza dover rileggere tutto il journal log.

## High-Availability

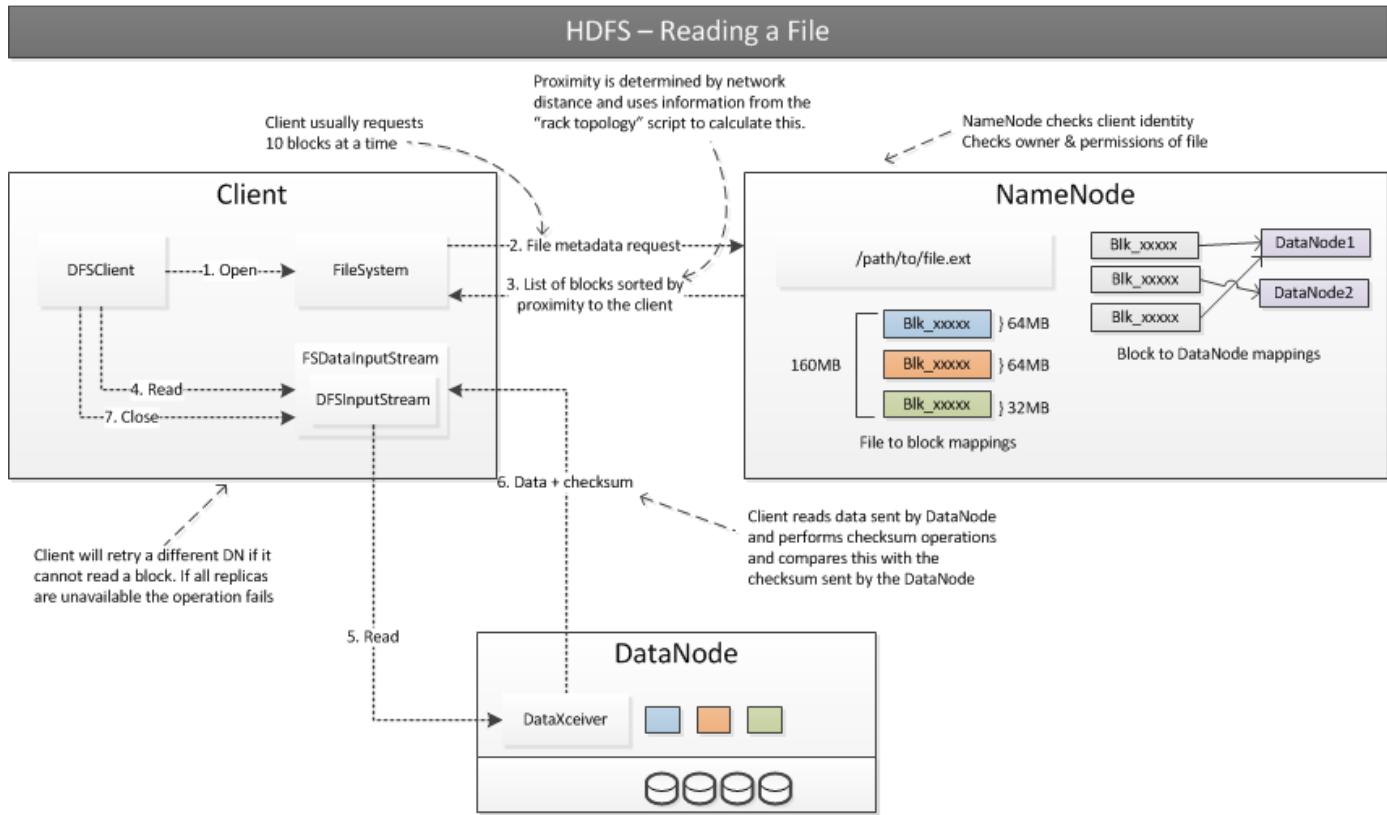
Prima di Hadoop 2 il Name Node era un single point of failure in un cluster HDFS.

In un tipico cluster HA, due diverse macchine sono configurate come NN: in un determinato istante solo una delle due macchine è un NN in stato attivo e l'altra è posta in standby.

Il NN in standby conosce i metadati del name space:

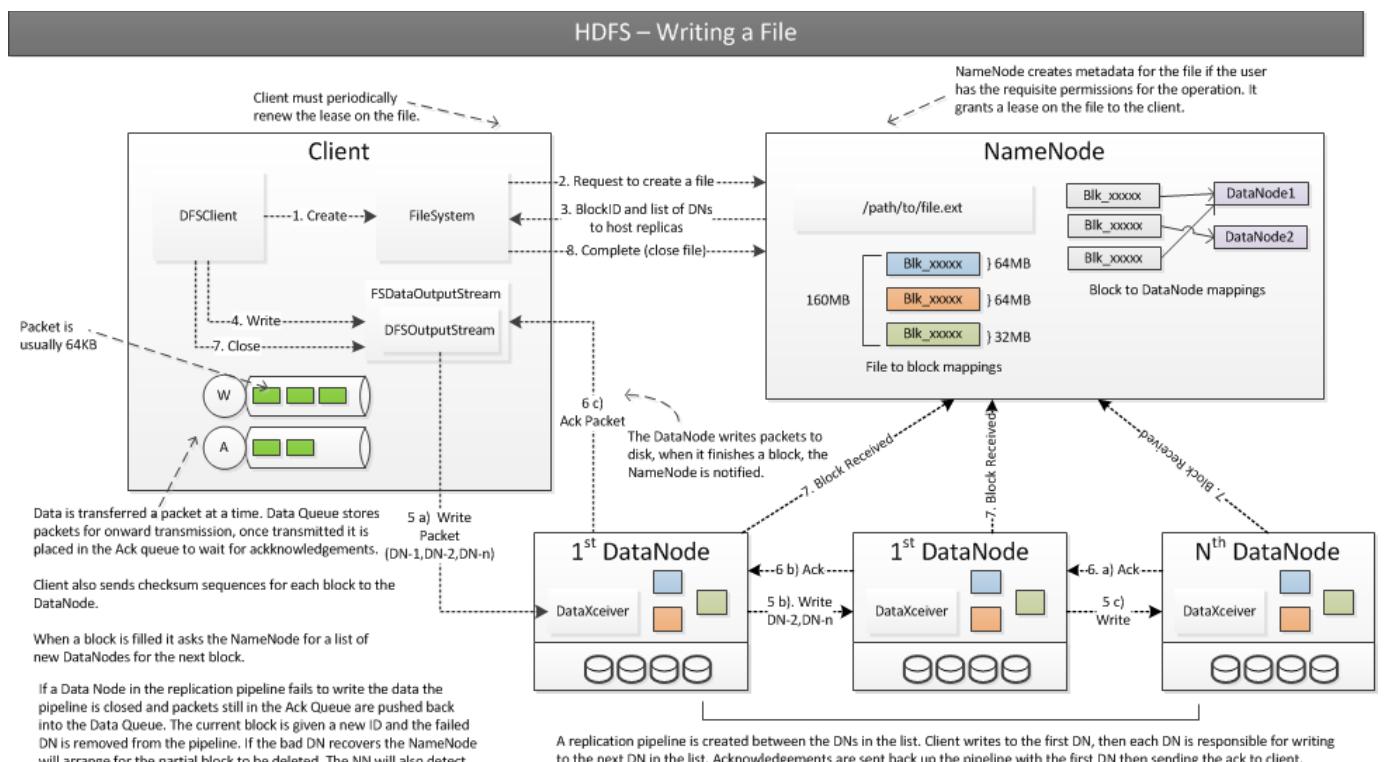
- Legge il file di log scritto nel NN attivo, in caso di failover, il NN secondario si assicura di aver letto tutte le variazioni nei dati prima di portarsi in stato attivo; in questo modo lo stato nel namespace è completamente sincronizzato prima del failover
- Riceve informazioni sulle locazioni dei blocchi di dati e la directory di heartbeats direttamente dal DN configurato con la locazione di entrambi i NN.

## Lettura da HDFS



Vijay Thakorla, V0.1.  
CC BY 2.0

## Scrittura su HDFS



Vijay Thakorla, V0.1.  
CC BY 2.0

## Il meccanismo Hear Beat

Gli **Heart Beats** sono segnali periodicamente inviati dai Data Node al Name Node per aggiornarlo sui loro stati.

L'assenza di un segnale genera diverse azioni nel cluster:

- Se il NN non riceve segnali Heart Beats da un DN, lo considera inattivo e crea dei suoi blocchi di dati una replica, su differenti DNs
- L'Application Master deve inviare segnali Heart Beats al Resource Manager per informarlo di essere attivo ed in esecuzione.

## YARN

YARN è la piattaforma che si occupa di gestire le risorse responsabili di calcolare le risorse nel cluster e utilizzarle per schedulare le applicazioni.

Mentre in Hadoop 1 le funzionalità di scheduling e monitoraggio dei job erano entrambe in carico al job tracker, in Hadoop 2 sono ridistribuite su diversi agenti:

- un **Resource Manager (RM)** globale è l'autorità ultima che assegna le risorse alle applicazioni nel sistema; le risorse vengono negoziate, assegnate e utilizzate in base alla nozione astratta di **container** che incorpora elementi quali memoria, CPU, disco, rete, ...;
- un nodo “schiavo”, detto **Node Manager (NM)**, responsabile dei containers e del monitoraggio dell'uso delle loro risorse, riporta queste informazioni al RM
- Un nodo per ogni applicazione, detto **Application Manager (AM)**, si occupa di negoziare le risorse con il RM e lavorare con i diversi NM per eseguire e monitorare le attività.

Il Resource Manager ha due componenti principali:

- lo **Scheduler** responsabile dell'allocazione delle risorse alle differenti applicazioni in esecuzione:
  - ✓ non effettua monitoraggio o né traccia lo stato delle applicazioni
  - ✓ non fornisce garanzie circa il riavvio di tasks a causa di failure delle applicazioni o dell'hardware
- l'**Applications Manager (AsM)** responsabile di accettare la sottomissione di job, negoziare il primo container per l'esecuzione dello specifico AM di applicazione e fornisce i servizi per riavviare l'AM container in caso di failure.

## Application Master

L'Application Master garantisce a YARN:

- una maggiore **scalabilità**: molte funzionalità sono distribuite in diversi AMs in modo che l'RM non sia un collo di bottiglia del cluster; RM è puramente uno scheduler e, ad esempio, non tenta di fornire tolleranza ai guasti per le risorse
- una maggiore **apertura**: spostando tutto il codice del framework applicativo nell'AM, il sistema è generalizzato e può supportare diversi framework come MapReduce, MP e Graph Processing.

RM non può fidarsi degli AM in quanto contengono essenzialmente codice utente; il sistema YARN (RM e NM) deve proteggere se stesso e le risorse da AMs guaste o malevole.

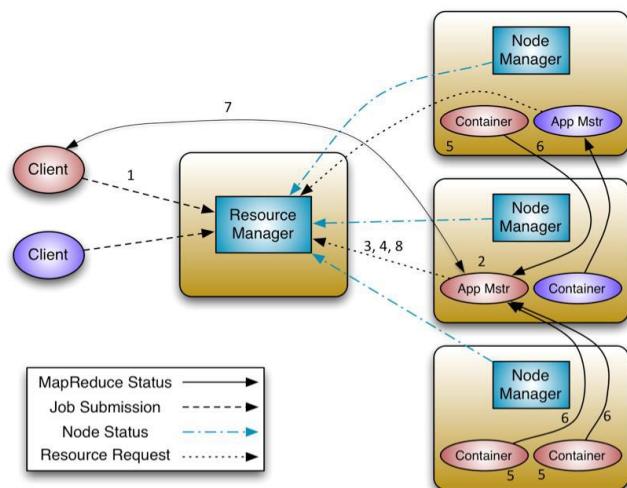
In base ai Containers che riceve da RM, l'AM può aggiornare il proprio piano di esecuzione in caso di uso delle risorse scarso o eccessivo. L'allocazione e la de-allocazione dei container può essere dinamicamente adattata durante i progressi dell'esecuzione delle applicazioni.

## YARN - Walkthrough

L'esecuzione delle applicazioni prevede i seguenti passi:

- sottomissione dell'applicazione
- bootstrapping dell'istanza dell'AM per l'applicazione
- esecuzione dell'applicazione gestita dall'istanza AM.

1. Un programma client **sottomette l'applicazione** includendo le specifiche necessarie per eseguire l'AM specifico dell'applicazione
2. Il RM si assume la responsabilità di negoziare uno specifico container nel quale avviare l'AM ed eseguirlo
3. L'AM, all'avvio, **si registra presso il RM** allo scopo di permettere al client di interrogare l'RM per dettagli, l'RM può comunicare direttamente con l'AM
4. Durante le normali operazioni l'AM **negozi** le appropriate risorse tramite il protocollo resource-request
5. Al completamento dell'allocazione del container, l'AM esegue il container fornendo le specifiche di esecuzione al NM; le specifiche, tipicamente, includono le informazioni necessarie per permettere al container di comunicare con il proprio AM
6. L'esecuzione del codice dell'applicazione all'interno del container, fornisce le informazioni necessarie all'AM (avanzamento, stato, ...)
7. Durante l'esecuzione dell'applicazione, il client che ha sottomesso il programma **comunica** direttamente con l'AM per ottenerne lo stato, l'avanzamento, ....
8. Al completamento dell'applicazione, quando tutti i lavori necessari sono terminati, l'AM si **de-registra** dal RM e termina, abilitando la ri-assegnazione del container.



## Hadoop File format

L'ecosistema di Hadoop supporta molti diversi file format:

- AVRO: un sistema di serializzazione dei dati
- ORC: Optimized Row Columnar
- Parquet

## AVRO

AVRO è un framework di Remote Procedure Call e serializzazione dati utilizzato:

- come formato di serializzazione per i dati persistenti
- come formato per la comunicazione tra i nodi Hadoop e tra i programmi client e i servizi di Hadoop
- durante la fase di shuffle del Map-Reduce.

Si basa su JSON per definire i tipi di dati e i protocolli, serializza i dati in formati binari compatti (o testo in chiaro per il debug). AVRO è progettato per essere indipendente dal linguaggio e binding per molti linguaggi (Java, C, C++, Python, Ruby, ...).

AVRO si basa sugli schemi. Quando i dati vengono letti, lo schema utilizzato in fase di scrittura è sempre presente. I vantaggi principali con Avro rispetto a sistemi simili sono:

- **Untagged data:** dato che lo schema è presente quando il dato viene letto, un importante quantità di informazioni non deve essere codificata con i dati, generano un dato serializzato più piccolo
- **Tipizzazione dinamica:** i dati sono sempre associati a schemi che ne permettono il completo utilizzo senza generazione di codice, tipizzazione statica, ... questo semplifica la costruzione di sistemi e linguaggi generici

per l'elaborazione dei dati, le differenze possono essere risolte utilizzando i nomi dei campi; è possibile utilizzare schemi differenti per la serializzazione e la deserializzazione. Avro può gestire i campi mancanti, aggiuntivi, modificati

- **Evoluzione degli schemi:** quando un schema cambia, sia il vecchio che il nuovo sono sempre presenti durante l'elaborazione

<pre>{   "type": "record",   "name": "Person",   "fields": [     {"name": "first", "type": "string"},     {"name": "sport", "type": "string"}   ] }</pre>	<pre>{   "type": "record",   "name": "Person",   "fields": [     {"name": "first", "type": "string"},     {"name": "age", "type": "int", "default": 0}   ] }</pre>
Serialized data "Matteo", "Sailing"	Deserialized data "Matteo",

## Parquet

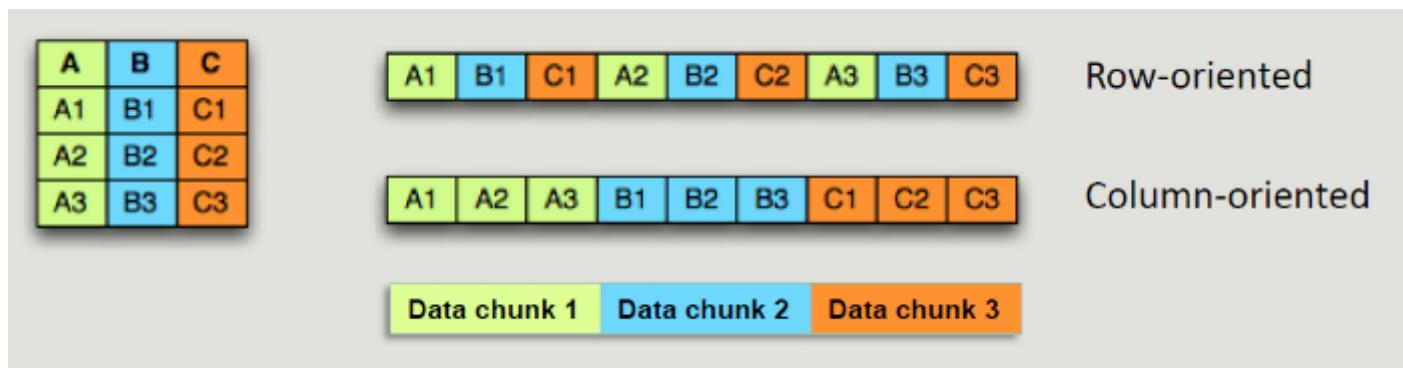
Parquet è uno storage a formato colonnare per migliorare l'efficienza delle query di **nested structures** in formato "appiattito".

Parquet deriva dal formato **Google Dremelfile**, è open source e assicura interoperabilità tra diversi framework e motori di ricerca.

## Storage colonnare

I formati di memorizzazione colonnari offrono diversi vantaggi:

- la memorizzazione organizzata per colonne garantisce una maggiore omogeneità dei dati migliorandone la compressione e permettendo di risparmiare davvero tanto spazio nei sistemi Hadoop
- l'Input/Output è ridotto in quanto è possibile analizzare efficientemente solo un sottoinsieme di colonne che riguardano i dati, una migliore compressione riduce inoltre la capacità di banda richiesta per leggere l'input
- è possibile continuare ad utilizzare i moderni processori e le pipeline rendendo le istruzioni di branching maggiormente prevedibili.



## Data model

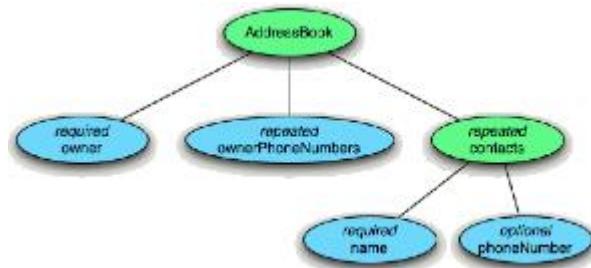
Per supportare i record nidificati è necessario memorizzare il livello per cui il campo è nullo. Questo è lo scopo del livello di definizione: da 0 alla radice dello schema fino al livello massimo per questa colonna. Quando viene definito un campo, vengono definiti anche tutti i suoi genitori, ma quando è nullo dobbiamo registrare il livello al quale ha iniziato a essere nullo per poter ricostruire il record.

In uno schema piatto, un campo opzionale è codificato su un singolo bit usando 0 per null e 1 per valore definito. In uno schema nidificato, utilizziamo un valore aggiuntivo per ogni livello di annidamento (come mostrato nell'esempio), infine se è richiesto un campo non è necessario un livello di definizione.

```
message ExampleDefinitionLevel{
    optional group a {
        optional group b {
            optional string c;
        }
    }
}
```

### Unnesting

Per archiviare strutture di dati nidificati in formato colonna, è necessario mappare lo schema su un elenco di colonne in modo da poter scrivere record su colonne piatte e rileggerli alla loro struttura di dati nidificata originale. Parquet crea una colonna per ogni campo di tipo primitivo nello schema. Se rappresentassimo lo schema come un albero, i tipi primitivi sarebbero le foglie di questo albero.



Column	Type	AddressBook	
owner	string	owner	ownerPhoneNumbers
ownerPhoneNumbers	string		contacts.name
contacts.name	string		
contacts.phoneNumber	string		

La struttura del record viene acquisita per ciascun valore da due numeri interi chiamati **livello di ripetizione** e **livello di definizione**. Usando i livelli di definizione e ripetizione, è possibile ricostruire completamente le strutture nidificate.

### Definition level

```
message ExampleDefinitionLevel{
    optional group a {
        optional group b {
            optional string c;
        }
    }
}
```

Value	Definition Level
a: null	0
a: { b: null }	1
a: { b: { c: null } }	2
a: { b: { c: "foo" } }	3 (actually defined)

Contiene una colonna: a.b.c nella quale tutti i campi sono opzionali e possono essere nulli. Quando c viene definito, vengono necessariamente definiti anche a e b, ma se c è nullo, è necessario salvare il livello del valore null: esistono 3 campi opzionali nidificati, quindi il livello di definizione massimo è 3.

Nel prossimo esempio il livello massimo di definizione è 2. Il valore del livello di definizione per i campi sotto b cambia come segue:

```
message ExampleDefinitionLevel{
    optional group a {
        required group b {
            optional string c;
        }
    }
}
```

Value	Definition Level
a: null	0
a: { b: null }	Impossible, as b is required
a: { b: { c: null } }	1
a: { b: { c: "foo" } }	2 (actually defined)

Ridurre i livelli di definizione è importante in quanto l'obiettivo è quello di memorizzare i livelli nel minor numero di bit possibile.

### Repetition level

La presenza di campi ripetuti richiede di effettuare la memorizzazione in colonne di valori per le nuove liste.

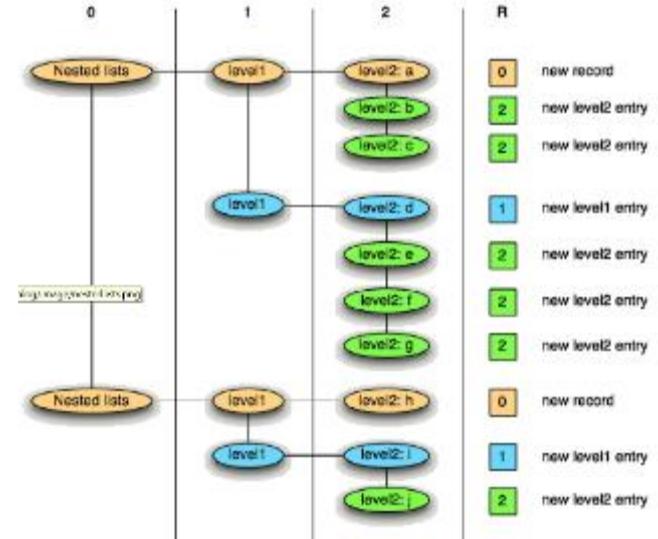
Il livello di ripetizione indica a quale livello deve essere creato un nuovo elenco per il valore corrente. In altre parole, il livello di ripetizione può essere visto come un indicatore di quando iniziare un nuovo elenco e a quale livello. Ad esempio, considerare la seguente rappresentazione di un elenco di elenchi di stringhe:

- il valore 0 indica nuovi record e implica una nuova lista a livello 1 e livello 2
- il valore 1 indica ogni nuova lista a livello 1 e implica la creazione anche di una lista di livello 2
- il valore 2 indica ogni nuovo elemento nella list2

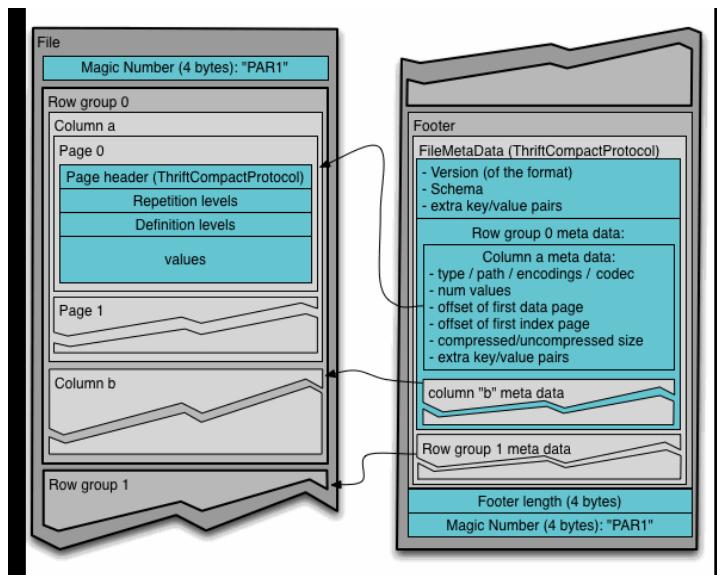
Schema:	Data:
	<pre> {     level1: {         level2: a         level2: b         level2: c     },     level1: {         level2: d         level2: e         level2: f         level2: g     } }, {     level1: {         level2: h     },     level1: {         level2: i         level2: j     } } </pre>
<pre> message nestedLists {     repeated group level1 {         repeated string level2;     } } </pre>	
Repetition level	Value
0	a
2	b
2	c
1	d
2	e
2	f
2	g
2	h
0	i
1	j
2	

La presenza di campi ripetuti richiede l'archiviazione quando i nuovi elenchi iniziano in una colonna di valori. È il livello al quale deve essere creato un nuovo elenco per il valore corrente. In altre parole, il livello di ripetizione può essere visto come un indicatore di quando iniziare un nuovo elenco e a quale livello.

Solo i livelli che si ripetono richiedono un livello di ripetizione: i campi opzionali o obbligatori non vengono mai ripetuti e possono essere saltati durante l'attribuzione dei livelli di ripetizione.



## File format



**File:** un file HDFS che include anche i metadati.

**Row group:** partizione logica orizzontale dei dati in righe; non esiste una struttura fisica che garantisca il raggruppamento: un gruppo è una parte di colonna per ogni colonna del dataset.

**Column chunk:** parte di dati per una specifica Colonna; esiste un row group specifico ed è garantito essere contiguo all'interno del file.

**Page:** i gruppi di colonne sono suddivisi in pagine, una pagina è un'unità concettualmente indivisibile (in termini di compressione e codifica); possono esistere diversi tipi di pagine intervallate in gruppi di colonne.

## Unit of parallelization

La parallelizzazione dell'accesso ai dati in Parquet può essere effettuata a diversi livelli:

- MapReduce-File/Row Group
- IO -Column chunk
- Encoding/Compression -Page

## Configuration

**Row group size:** gruppi di righe più grandi consentono blocchi di colonna più grandi, il che rende possibile eseguire IO sequenziali più grandi.

Si consigliano gruppi di righe di grandi dimensioni (512 MB -1 GB) poiché potrebbe essere necessario leggere un intero gruppo di righe, vogliamo che si adatti completamente a un blocco HDFS. Un'impostazione di lettura ottimizzata sarebbe: gruppi di righe da 1 GB, dimensione del blocco HDFS da 1 GB, 1 blocco HDFS per file HDFS.

**Data page size:** le pagine di dati dovrebbero essere considerate indivisibili quindi:

- pagine di dati più piccole consentono una lettura più dettagliata (ad es. Ricerca a riga singola).
- i formati di pagina più grandi comportano un minore sovraccarico di spazio (meno intestazioni di pagina) e un minore sovraccarico di analisi (elaborazione delle intestazioni).

Nota: le scansioni sequenziali non sono tenute a leggere una pagina alla volta in quanto non lavora a livello IO. Si consigliano 8KB per le dimensioni della pagina.