

# Sistemi Informativi

telegram/github/twitter: @martinopiaggi

2021-2022

# Contents

<b>1</b>	<b>Introduzione</b>	<b>3</b>
<b>2</b>	<b>Dati, informazioni e progettazione del SI</b>	<b>3</b>
2.1	Piramide DIKW . . . . .	3
2.2	Modelli per rappresentazione SI . . . . .	3
2.2.1	La Piramide di Anthony . . . . .	4
2.2.2	La catena del valore di Porter . . . . .	4
2.2.3	Risorsa . . . . .	4
2.2.4	Varie classificazioni SI: . . . . .	5
2.3	OLAP e OLTP . . . . .	5
2.4	Progettazione SI . . . . .	5
2.4.1	Framework di Zachmann . . . . .	5
2.5	Progettazione e gestione SI . . . . .	6
<b>3</b>	<b>Approccio architetturale ai SI</b>	<b>7</b>
3.1	BOAT . . . . .	7
3.1.1	Prospettiva di Business . . . . .	7
3.1.2	Prospettiva organizzativa . . . . .	9
3.2	Prospettiva architetturale . . . . .	11
3.3	Prospettiva tecnologica . . . . .	11
<b>4</b>	<b>BPMN per modellazione processi</b>	<b>12</b>
<b>5</b>	<b>Aspetto tecnologico dei SI</b>	<b>16</b>
5.1	Tecnologie a livello applicativo . . . . .	16
5.1.1	ERP . . . . .	16
5.1.2	CRM . . . . .	17
5.1.3	Data Warehouse . . . . .	17
5.1.4	Approccio Make or Buy . . . . .	19
5.2	Livello di piattaforma . . . . .	19
5.3	Architettura fisica . . . . .	20
5.3.1	Layer e tier . . . . .	20
5.3.2	Scalabilità . . . . .	20
5.3.3	Server Farm . . . . .	20
5.3.4	Cloud Computing . . . . .	20
5.3.5	Insourcing e Outsourcing . . . . .	21
5.4	Sicurezza dei sistemi informativi . . . . .	22
5.4.1	Proprietà della sicurezza . . . . .	22
5.4.2	Attacchi di rete . . . . .	22
5.4.3	Attacchi Applicativi . . . . .	22
5.4.4	Crittografia simmetrica e asimmetrica . . . . .	23
5.4.5	Impronta e firma digitale . . . . .	23
5.4.6	Certificati digitale . . . . .	23
5.4.7	Regole e politiche d'accesso . . . . .	23
5.4.8	Politiche d'accesso . . . . .	24
5.4.9	Firewall . . . . .	24
5.5	Archimate . . . . .	25

# 1 Introduzione

Cos'è un sistema informativo?

Un insieme informativo è l'insieme di mezzi, conoscenza e competenze usati per gestire le informazioni aziendali.

Il sistema informatico **non** è un sistema informativo, un sistema informatico è una parte del SI.

## 2 Dati, informazioni e progettazione del SI

### 2.1 Piramide DIKW

Dato: valore dettagliato di un evento. Informazione: un dato a cui si associa una certa importanza/contesto

Conoscenza: giudizio dell'informazione Saggezza: conoscenza++



### 2.2 Modelli per rappresentazione SI

Esistono vari modelli per rappresentare l'intero sistema informativo.

### 2.2.1 La Piramide di Anthony



Il modello di Anthony (la piramide riportata qua sopra) è un modello gerarchico organizzativo diviso in livelli:

- strategico
- tattico
- operativo

Ovviamente devono esserci scambi d'informazione, sia verticali che orizzontali. Tale modello però non consente di avere una visione globale sull'intera organizzazione.

### 2.2.2 La catena del valore di Porter

Altro metodo di segmentazione e rappresentazione del SI è la **catena del valore di Porter**. Modello che permette di descrivere la struttura di una organizzazione come un insieme di processi, divisi in 5 primari e 4 di supporto. Si adatta però solo a grandi organizzazioni che vendono bene e non offrono servizi.

### 2.2.3 Risorsa

**Risorsa** è tutto ciò con cui un'organizzazione opera, sia materiale che digitale, per raggiungere i suoi obiettivi. Si possono classificare in:

- Esterne
- Interne
- Struttura
- Gestione

#### 2.2.4 Varie classificazioni SI:

Spesso i sistemi informativi sono divisi anche in due tipi di attività diverse:

- **Orizzontali**, cioè tutti quei processi che non cambiano i settori diversi (HR, aspetti amministrativi, ...)
- **Verticali**, cioè che variano secondo il contesto, solitamente le attività produttive. Un processo è l'insieme delle attività che l'organizzazione svolge per raggiungere il proprio obiettivo (ben definito).

Altra classificazione in 2 categorie:

- **Sistemi operazionali**: svolgono operazioni transazionali semplici, lavoro di ufficio, contabilità.
- **Sistemi decisionali o informativi**: fanno da supporto delle attività decisionali e strategiche aziendali, utilizzando i dati.

### 2.3 OLAP e OLTP

In un SI possiamo avere sistemi:

- **OLAP (Online Analytical Processing)**: ha come metrica il **response time**. Operazioni non strutturate, complesse che operano su tanti dati aggregati. Non è importante rispettare la proprietà ACID in quanto le operazioni sono solo di sola lettura e casuali.
- **OLTP (Online Transaction Processing)**: orientate alle transazioni. Tante transazioni, tanti utenti. La metrica di valutazione è il **throughput**.

	<b>OLTP</b>	<b>OLAP</b>
Utente	Impiegato (molti)	Dirigente (pochi)
Funzione	Operazioni giornaliere	Supporto alle decisioni
Progettazione	Orientata all'applicazione	Orientata al soggetto
Dati	Correnti, aggiornati, dettagliati, relazionali, omogenei	Storici, aggregati, multidimensionali, eterogenei
Uso	Ripetitivo	Casuale
Accesso	Read-Write, indicizzato	Read, sequenziale
Unità di lavoro	Transazione breve	Interrogazione complessa
Metrica	Throughput	Tempo di risposta

### 2.4 Progettazione SI

#### 2.4.1 Framework di Zachmann

Il framework di Zachman è uno strumento efficace per descrivere il sistema informativo dell'organizzazione nel suo complesso. Esso è organizzato in:

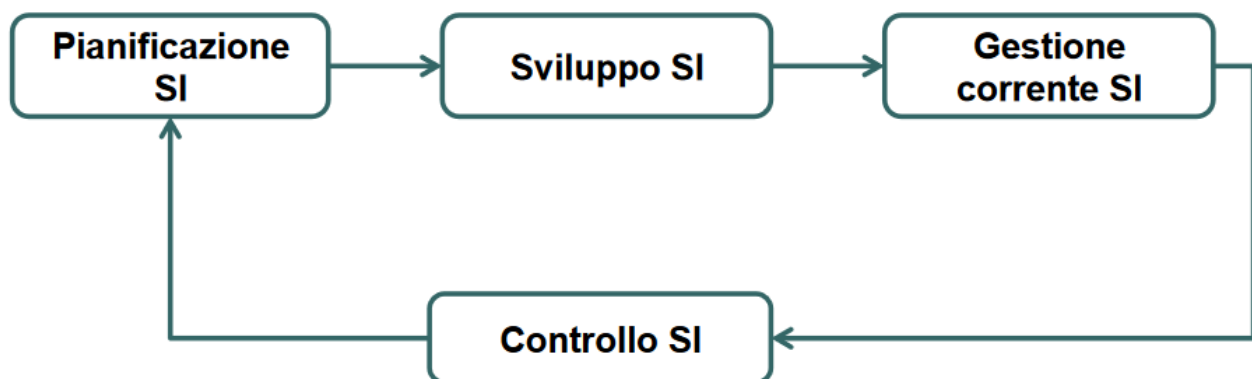
- **Righe**: definiscono i diversi punti di vista che possono interessare diversi stakeholder (*coloro che hanno interesse al sistema considerato perché ad esempio ne sono proprietari o fruitori, finanziatori o utenti*).
- **Colonne**: definiscono gli aspetti da analizzare.

Nel framework di Zachman non c'è alcun ordinamento tra le colonne, e ciascuna di esse (dall'alto verso il basso) mostra tutti i passaggi a partire dal business per arrivare all'effettiva realizzazione del sistema.

	What Cosa? DATI	How Come? FUNZIONI	Where Dove? RETE	Who Chi? PERSONE	When Quando? TEMPO	Why Perché? MOTIVAZIONE
Contestuale AMBITO <i>Pianificatore</i>	Liste elementi importanti per l'impresa	Lista processi eseguiti dall'impresa	Lista località in cui opera l'azienda	Lista unità organizzative importanti per l'azienda	Lista eventi	Lista obiettivi, strategie
Concettuale Modello Impresa <i>Owner</i>	Modello Entità-Relazie	Modello dei processi	Rete logica	Modello della struttura organizzativa	Modello eventi	Relazioni tra obiettivi
Logico Modello sistema <i>Progettista</i>	Modello dei dati	Diagramma dei processi	Architettura del sistema distribuito	Diagramma ruoli e relazioni	Diagramma eventi	Diagramma regole
Fisico Modello tecnologico <i>Costruttore</i>	Progettazione dei dati	Specifiche delle funzioni	Architettura del sistema	Interfaccia uomo-macchina	Specifiche eventi	Specifiche regole
Dettaglio Componenti <i>Sottocontraente</i>	Es. Definizione dei dati	Es. Applicazione (codice)	Es. Architettura di rete	Es. Architettura sicurezza	Dettagli eventi	Dettagli regole

Le celle del framework di Zachman create dall'intersezione tra righe e colonne definiscono le **Viste**.

## 2.5 Progettazione e gestione SI



C'è l'esigenza di progettare e gestire in modo strutturato un sistema informativo nel suo funzionamento e nella sua evoluzione. Identifichiamo 4 fasi:

- **Pianificazione:** fase in cui si delineano le linee guida strategiche, il ruolo delle componenti organizzative e

le istruzioni operative per la realizzazione. A sua volta diviso in:

- Pianificazione strategica: in questa fase si identificano gli obiettivi che richiedono interventi di natura informatica.
- Studio di fattibilità: una volta appurato il bisogno di un intervento di natura informatica, si identificano le alternative progettuali.
- **Sviluppo**: raccolta e analisi dei requisiti e sviluppo del software e definizione dell'architettura hardware.
- **Gestione corrente**: interventi di routine eseguiti per il mantenimento del sistema.
- **Controllo**: operazioni effettuate in modo periodico per valutare l'adeguatezza del sistema informativo.

## 3 Approccio architetturale ai SI

### 3.1 BOAT

La progettazione di un SI prevede quattro fasi quali la pianificazione, lo sviluppo, la gestione corrente e controllo. Per progettare la Enterprise Architecture (modello dell'organizzazione con varie prospettive che permette di fornire una panoramica sui propri processi, sistemi, tecnologie e strutture) possiamo usare l'approccio boat. Business, Organization, Architecture and Technology. Perché l'approccio Boat? L'approccio boat ci permette di analizzare il SI da tutti e 4 i punti di vista. Un SI viene progettato partendo dalla prospettiva Business, per poi (iterativamente) passare alle altre (B-O-A-T-B-O-A ...).

#### 3.1.1 Prospettiva di Business

Il business può essere quindi il punto di partenza per la progettazione del SI ma anche punto di partenza per la sua modifica (requirement pull), in alternativa a quando si modifica a causa di una spinta tecnologica (technology push). I partecipanti in uno scenario di Business possono essere:

- Business
- Government
- Citizen/Consumer

Abbiamo quindi tutti i vari casi di interazione tra i partecipanti .. B2B, B2G, B2C , ... Gli oggetti dell'interazione possono essere:

- prodotti fisici
- prodotti digitali
- prodotti finanziari
- servizi
- prodotti ibridi

L'orizzonte temporale di tali interazioni possono essere:

- statico = stabile nel tempo, solitamente regolata da un contratto
- semidinamico = stabile con possibili cambiamenti
- dinamico = rapporto limitato ai singoli ordini
- ultradinamico = la relazione può variare persino all'interno di un singolo ordine

**3.1.1.1 Business Driver** I Business Driver sono quelli che **motivano** effettivamente il progetto del SI o una sua modifica. I business Driver possono essere divisi in 2: Efficacia: definita come

$$\frac{OutputEffettivo}{OutputAtteso}$$

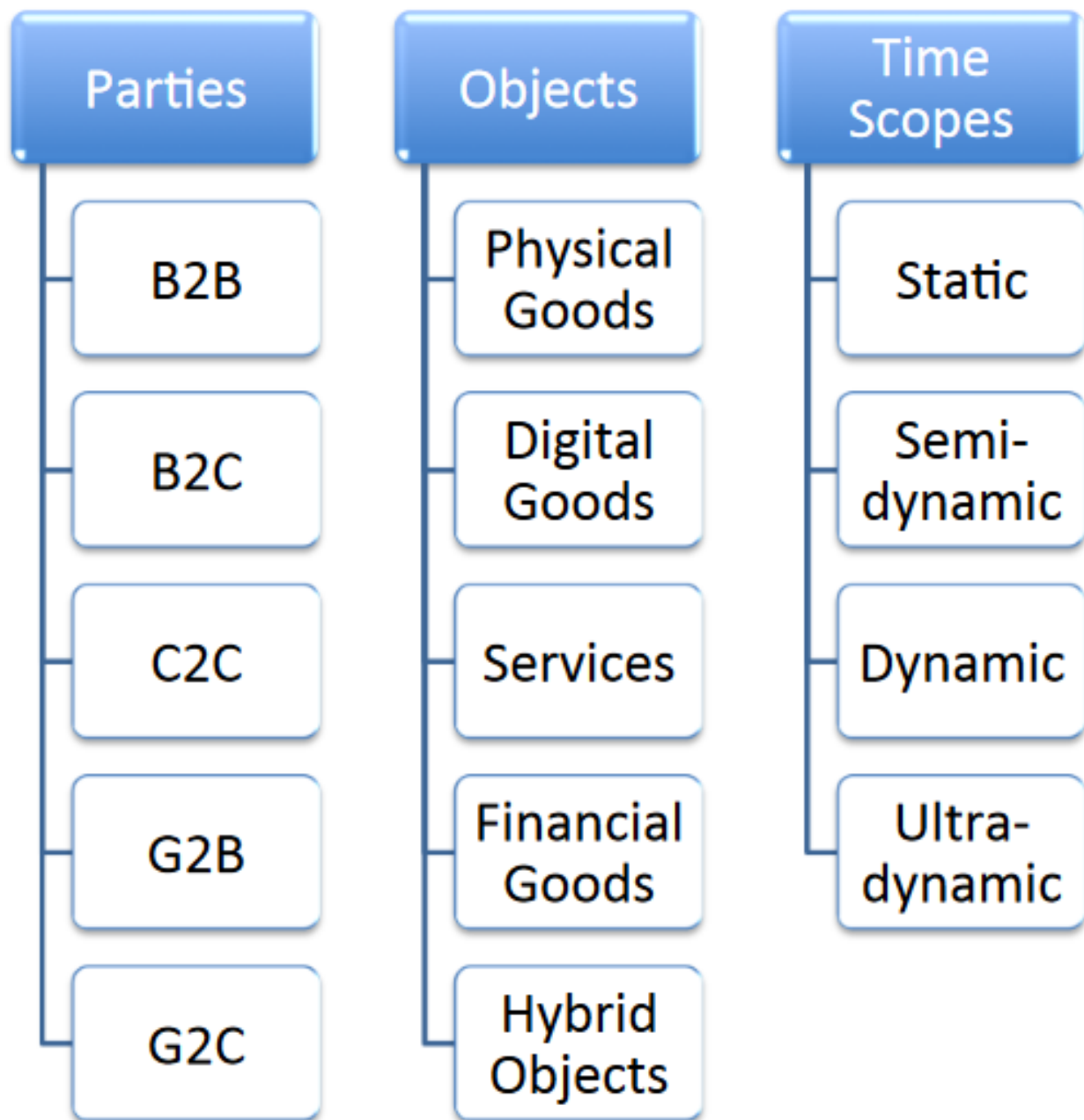
Nella valutazione dell'efficacia inoltre definiamo due qualità :

- Reach: l'estensione dell'efficacia in termini di estensione es:(temporale/geografica/multicanale)
- Richness: la qualità e ricchezza della comunicazione, cioè la frequenza, il dettaglio e la tipologia di dati.  
es:(frequenza, interattività)

Efficienza: definita come

$$\frac{OutputEffettivo}{Input}$$



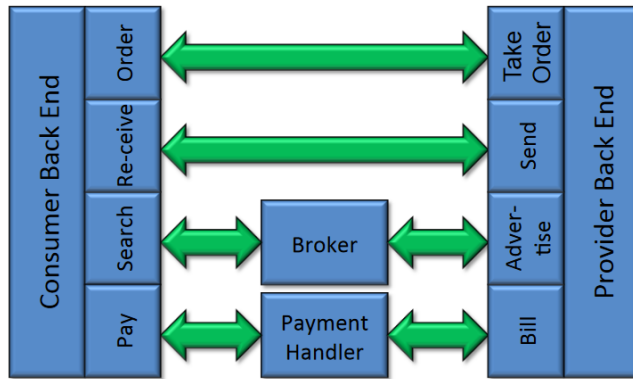


### 3.1.2 Prospettiva organizzativa

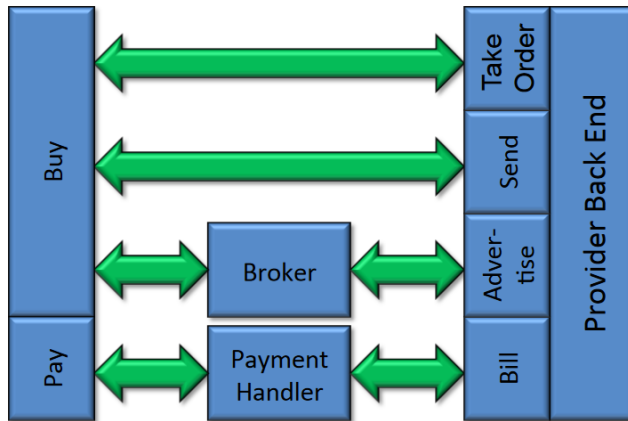
Per modellizzare l'organizzazione si procede per livelli, affinando sempre di più il dettaglio. Al *livello 0* si vede l'intero mercato come una blackbox. Al *livello 1* si distinguono i 3 attori della situa organizzativa:

- consumatore
  - intermediario
  - fornitore
- Al *livello 2* si specificano ulteriormente i vari intermediari. Al *livello 3* mostra anche gli aspetti intraorganizzativi, che si dividono in 2 categorie:

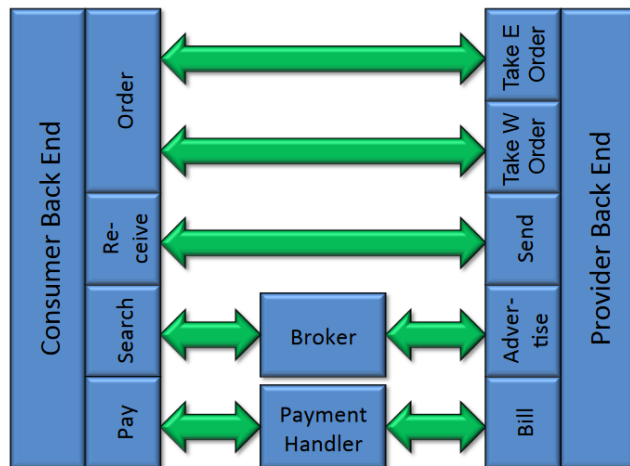
- front-end : tutte le componenti dell'organizzazione nel quale sono presenti interazioni con entità esterne
- back-end : tutte le componenti chiave all'interno dell'organizzazione che non interagiscono con partecipanti esterni ma solo con altre componenti interne. Al *livello 4* vengono modellizzate le singole componenti interne dell'organizzazione evidenziando anche le interazioni con i partecipanti esterni e intermediari.
- Esempio B2B a lv. 4



- Esempio B2C a lv. 4



- Esempio B2B a lv. 4 multicanale:



### 3.2 Prospettiva architetturale

La prospettiva di architettura ha diversi livelli di aggregazione:

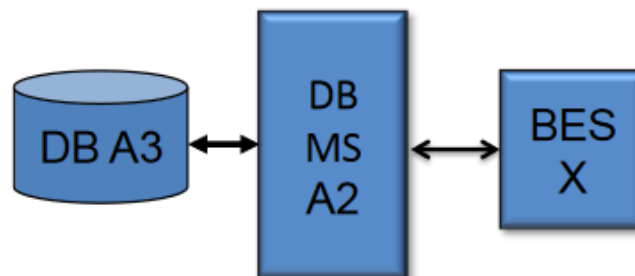
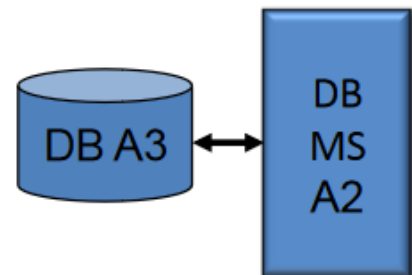
- **Architettura market-level** descrive la struttura dei sistemi a livello dei vari partecipanti, descrivendo le interazioni e i messaggi scambiati tra di essi. La possiamo vedere come una estensione della prospettiva O-4 (la quale modella le interazioni fra organizzazioni) in quanto specifica le interazioni tra i sistemi di tali organizzazioni. Si tratta di una estensione, e nonostante spesso c'è una corrispondenza tra i moduli delle due prospettive può non esserci. Per specificare quali moduli sono comuni utilizziamo una matrice.
- **Architettura party-level** descrive la singola organizzazione rappresentando le interfacce verso il mondo esterno, il quale è trasparente in questo livello di astrazione. Si tratta quindi di una visione intra-organizzativa nel quale descriviamo interfacce BE/BE, FE/FE. Specifichiamo anche i vari componenti tra cui i DB con annessi DBMS, e tutti i messaggi scambiati tra i componenti FE e BE (restano ancora visibili quelli scambiati con gli altri partecipanti);
- architettura system-level non ci interessa.

- **Generic component**

*Backend system X*



- **databases and DBMS**



### 3.3 Prospettiva tecnologica

Dividiamo la parte tecnologica del livello tecnologico del Boat in:

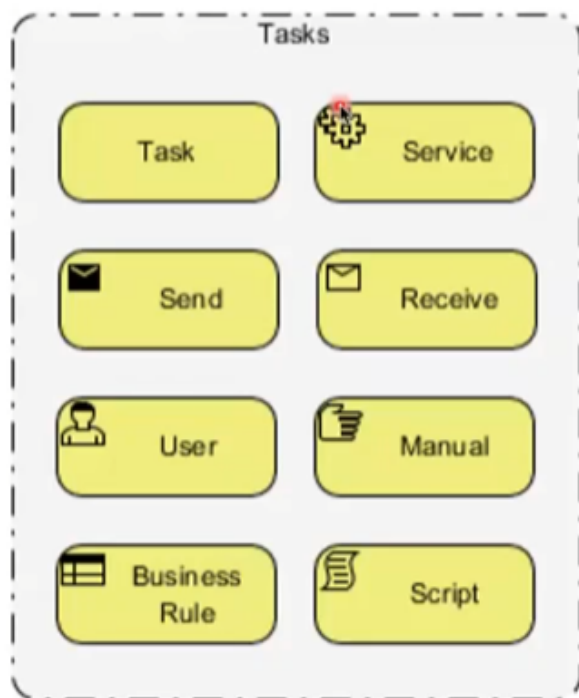
- livello applicativo
- livello di piattaforma
- livello di architettura fisica

## 4 BPMN per modellazione processi

Un processo di business è un insieme di attività eseguite secondo un ordine specifico per ottenere un certo obiettivo di business. Utilizziamo per modellare processi BPMN. Business Process Modeling Notation (BPMN). Nella notazione BPMN gli eventi sono modellati con un cerchio al cui interno è disegnato un simbolo che ne definisce la tipologia.

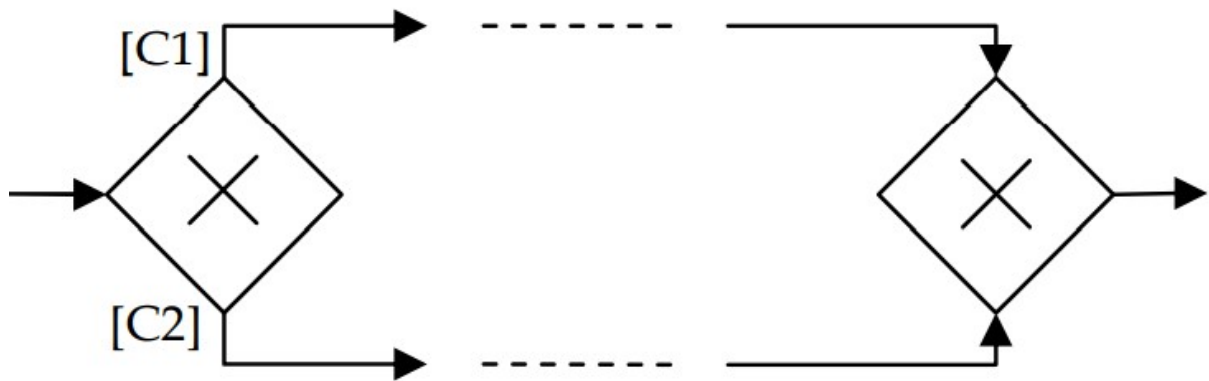


Tipi di attività:

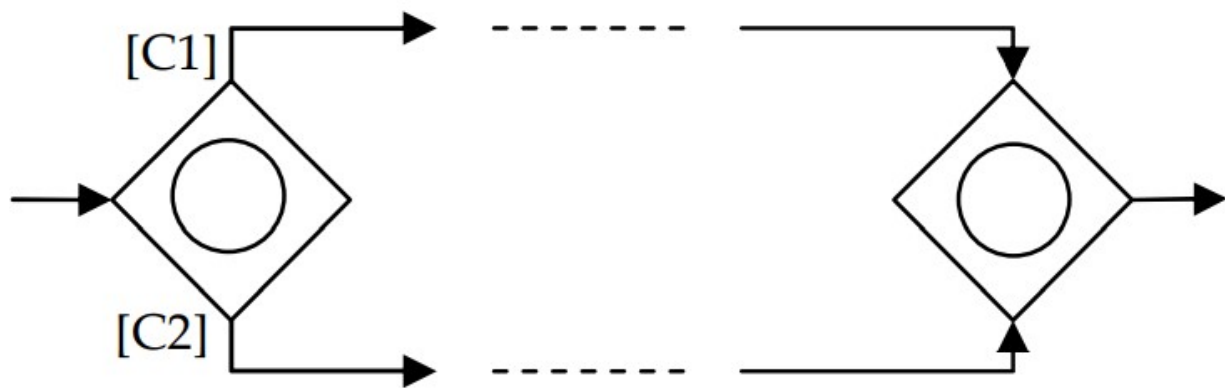


Tipi di Gateway:

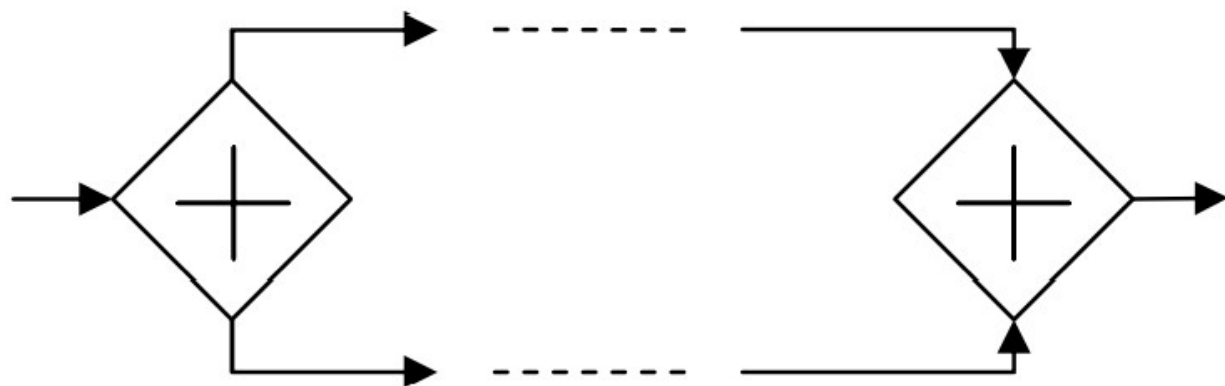
- Gateway XOR



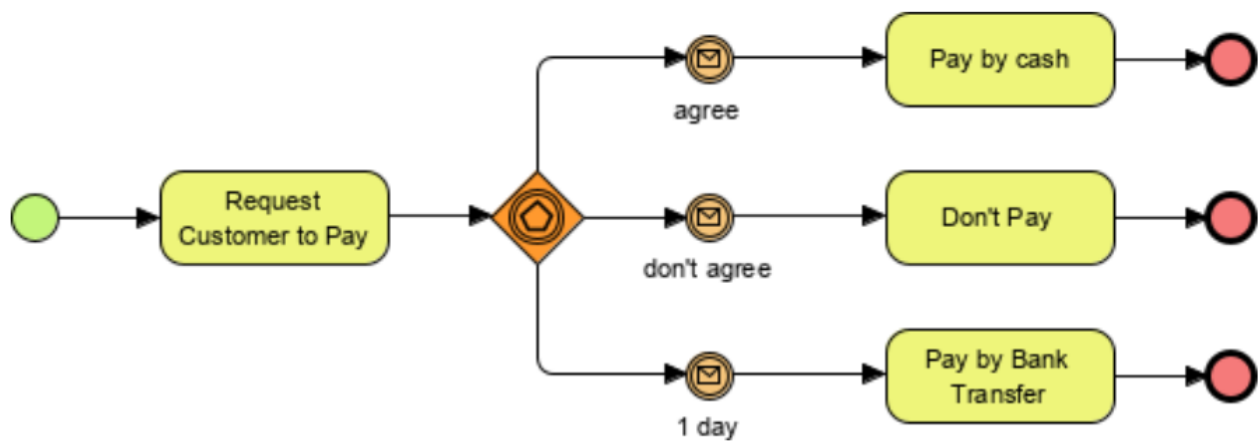
- Gateway OR



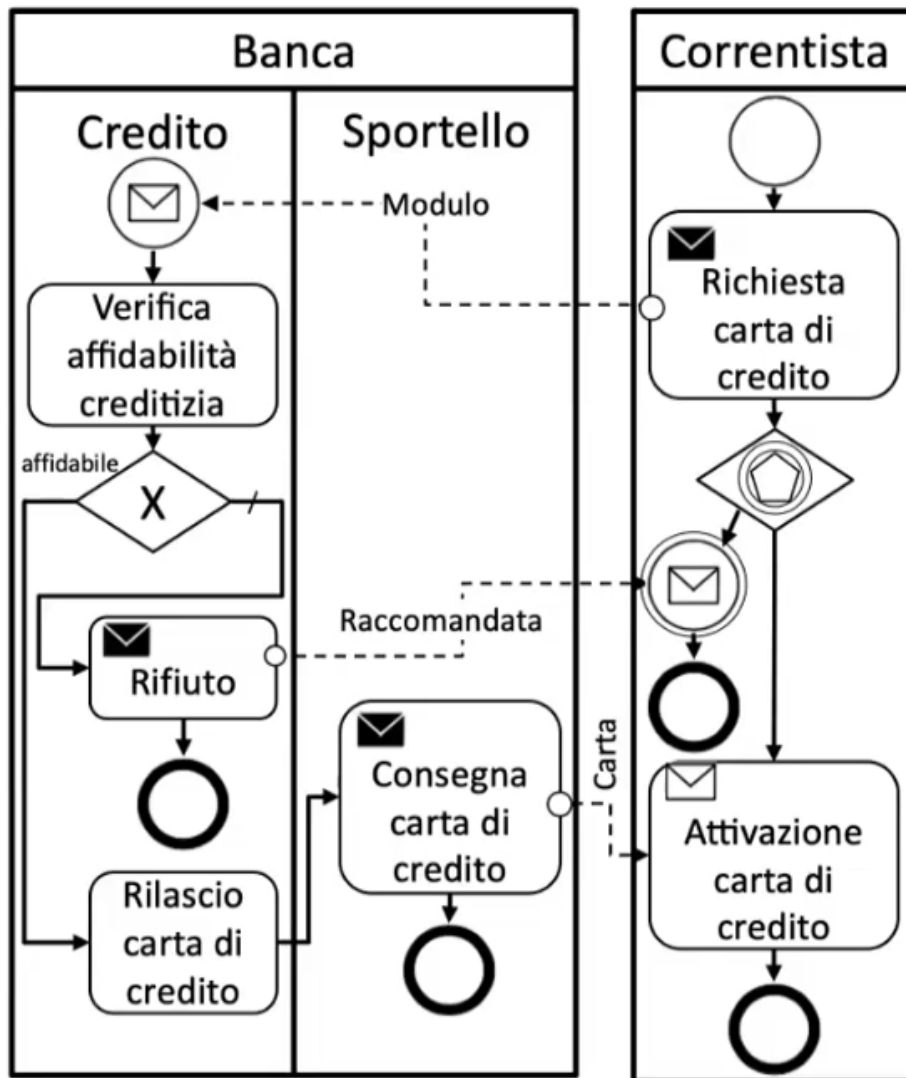
Gateway AND

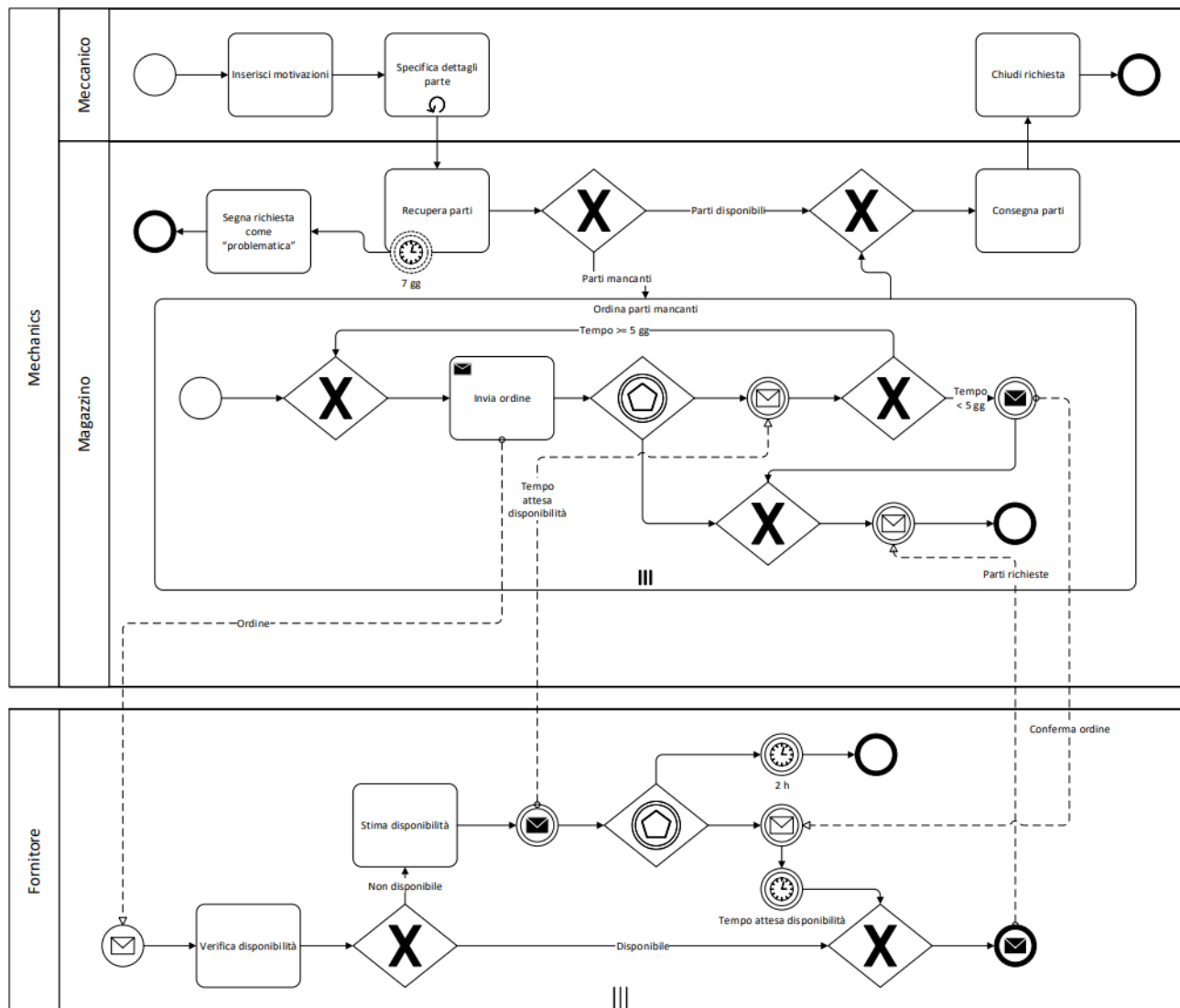


Gateway EVENT BASED



Esempi di BPMN collaborativo, cioè con più pools e più lanes.





## 5 Aspetto tecnologico dei SI

### 5.1 Tecnologie a livello applicativo

#### 5.1.1 ERP

Enterprise Resource Process é una suite software che offre i vari moduli software per il SI riguardo il sistema operativo interno. Il ruolo dei dati in un sistema ERP è fondamentale, in quanto uno degli scopi principali per cui viene installato un tale sistema in una azienda è il poter garantire l'unicità dei dati tra i vari moduli che gestiscono le diverse transazioni aziendali. È proprio l'unicità dei dati a livello operativo garantita dall'ERP che consente di ottenere unicità dei dati (e quindi eliminare ridondanze e incongruenze) anche a livello direzionale, e quindi poter considerare i dati, nel loro complesso, una risorsa aziendale. Gli ERP sono caratterizzati da:

- unicità dell'informazione : zero rindondanza e quindi zero incogruenza dei valori
- prescrittività : incorporano la logica di funzionamento dell'impresa. Norma cioè i processi.
- modularità : l'azienda 'sceglie' i moduli in base alle proprie esigenze.



I moduli di un ERP si possono classificare in:

- **ERP CORE:**
  - sistemi istituzionali : orizzontale,
  - settoriali : verticali, specifici
  - direzionali : orizzontale
- **ERP ESTESO:**
  - SCM (Supply Chain Management)
  - PLM (Product Lifecycle Management)

### 5.1.2 CRM

Customer Relationship Manager:

- CRM analitico: analisi sul CRM operativo
- CRM operativo: si divide in:
  - marketing
  - servizio clienti
  - vendite automatizzate
- CRM collaborativo: si occupa della fidelizzazione clienti principalmente

### 5.1.3 Data Warehouse

Il Data Warehouse é una grossa repository ottimizzata per analisi dei dati. Mantiene lo storico per garantire le analisi.

**5.1.3.1 DataMart** In un'architettura centralizzata esiste solo un grande DW mentre in caso di una soluzione distribuita abbiamo diverse sotto unità chiamate DataMart. Il DW può essere quindi:

- **Centralizzato** (non compare il DM)
- **Distribuito** (DW + DM)
- **DW virtuale** (il DM lo ricopre interamente)

**5.1.3.2 Tecniche ETL** Extraction - Transformation - Loading In che senso trasformazione? I dati non solo devono essere estratti da più sorgenti e integrati nel DW, ma devono anche subire diversi trasformazioni.

- **estrazione:** statica o incrementale a seconda che i dati siano considerati per intero o solamente quelli più recentemente modificati.
- **trasformazione:**
  - pulizia dei dati
  - riconciliazione
  - standardizzazione
  - ricerca duplicati
- **caricamento:** si occupa di trasferire i dati nel DW secondo il modello multidimensionale da esso definito.

L'ETL carica i dati in modo periodico, ad eccezioni magari di alcuni critici, i cui visualizzazione e analisi é necessaria.

### 5.1.3.3 Modello logico del DW

**5.1.3.3.1 Motori OLAP** Il data warehouse è un sistema di tipo OLAP (On Line Analytical Processing). Il paradigma OLAP è caratterizzato da poche transazioni e interrogazioni (query) complesse che richiedono aggregazione di dati. I sistemi OLAP hanno come misura di efficienza e di efficacia il tempo di risposta.

Un sistema OLAP memorizza dati in formato aggregato, memorizza dati storici, archiviati secondo schemi multi-dimensionali (generalmente schemi detti a stella). Spesso, le interrogazioni accedono a grandi quantità di dati per rispondere a quesiti complessi come, per esempio, quale è stato il profitto netto realizzato dall'azienda in una certa area geografica nello scorso anno. I sistemi OLAP sono utilizzati per l'elaborazione di dati orientata al supporto decisionale, quindi sono adeguati a funzionalità collocate a livello di pianificazione e strategico della piramide di Anthony.

Molap: Modello Multidimensionale

Il fatto è l'elemento rilevante. Possiamo utilizzare un modello multidimensionale. Cioè vedere appunto le dimensioni di un fatto (dimensione temporale, spaziale, tipi di prodotto, clienti .. ecc.) come dimensioni spaziali. Possiamo quindi collocare i fatti in un ipercubo, dove ogni fatto è l'incrocio dei valori nelle varie dimensioni.

Rolap: Modello Relazionale

Ridondanza inversa al tempo di accesso. Diversi schemi:

- schema a stella
- schema a fiocco di neve

Holap: Modello Ibrido

Sia Molap (vantaggioso per tempo di risposta), sia Rolap (vantaggioso quando ci sono tanti dati). Utilizzo un modello Molap per gli utenti nel DM, i quali necessitano tempi di risposta molto veloci ma pochi dati. E utilizzo un modello Rolap per il DW.

Operazioni Olap

- Slice : vincoli sulle dimensioni
- Dice : selezione su più dimensioni
- Rollup : tolgo granularità, cioè aggrego, perdendo dettagli
- Drill Down: aggiungo granularità

Ovviamente i dati nel DW ci sono sempre, tutte queste operazioni sono a un livello di visualizzazione.

**5.1.3.4 Data Mining** Nel DW possiamo fare Data Mining. Cioè passare dalle informazioni alla conoscenza. Due tipi di Data Mining: - supervised: conoscenze aggiuntive esterne - unsupervised: non ha bisogno di conoscenze aggiuntive esterne

Tecniche di DM:

- predittive: predicono eventi
- descrittive: ti descrive pattern
- prescrittive: "del futuro", ti dice le decisioni da prendere

Regole associative. Analisi dei dati e si cercano elementi che compaiono più spesso insieme. Cioè si cerca di capire pattern del tipo:  $x \implies y$ . Confidenza sarà:  $\frac{x \implies y}{x}$  Cioè quanto è rilevante la regola.

Regole di classificazione. Cioè aggiungo classi ai miei dati. Un algoritmo mi aiuta a riconoscere tali classi. Uso l'80% dei miei dati come training set. Poi il 20% dei miei dati come test set. Possiamo definire una matrice diagonale. True Positive, True Negative, False Positive, False Negative. Dove pos/neg si riferisce all'appartenza alla classe.

$$precision = \frac{TP}{TP + FP}$$
$$recall = \frac{TP}{TP + FN}$$

Vogliamo precision e recall i piú alti possibili.

Proprietá di classificazione: - accuratezza - velocità - scalabilità - robustezza

Algoritmo di classificazione:

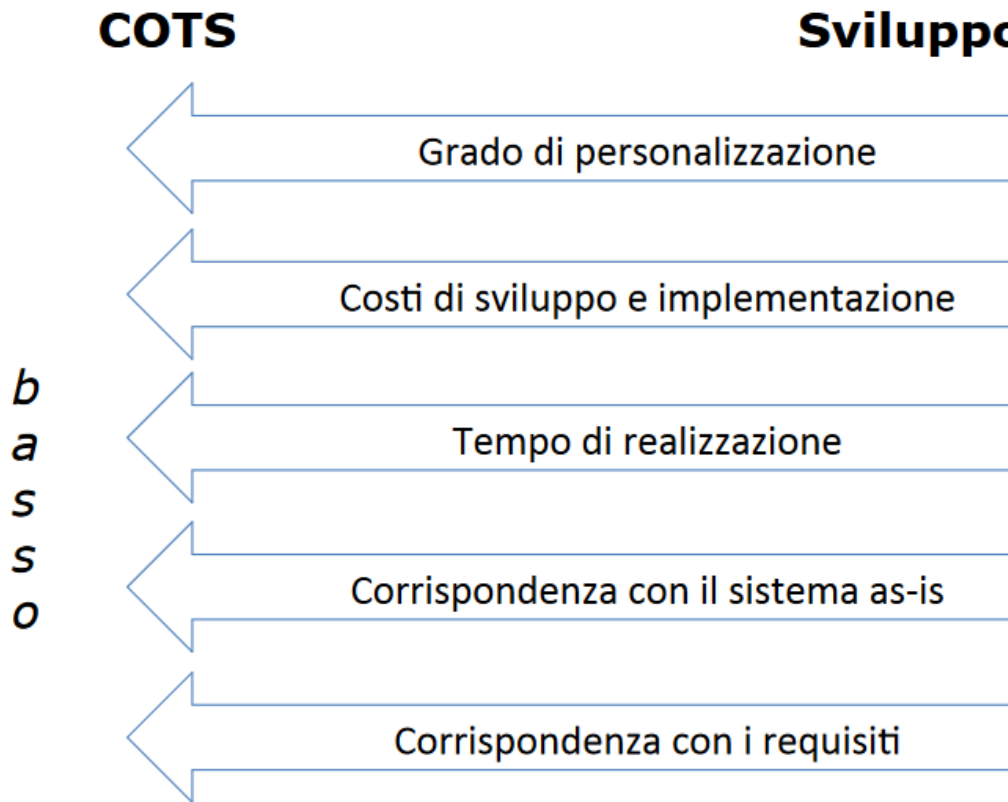
- clustering: descrittiva, non supervisionata. Funzione di similaritá, cioè cerca sinergia tra i dati.

Machine Learning, può essere:

- supervised
- unsupervised
- reinforcement learning
- transfer learning

#### 5.1.4 Approccio Make or Buy

L'analisi Make-or-Buy prende in considerazione il fatto che sul mercato per alcuni interventi esistono dei prodotti software già pronti: i cosiddetti COTS (Commercial Off-The-Shelf o Commercially available Off-The-Shelf). L'adozione di questi software (alternativa Buy) si pone in alternativa alla possibilità di progettare e realizzare un



prodotto su misura (alternativa Make).

## 5.2 Livello di piattaforma

Mentre l'ERP ha unificato i dati tra i databases risolvendo diversi problemi. Abbiamo comunque problemi per l'**integrazione** di moduli/applicazioni.

- **Point to point:** ogni messaggio costruito per specifiche interfacce ha il suo destinatario.
  - message oriented middleware: c'è un intermediario

- **Hub and Spoke:** un hub, la cui logica sarà complessa, centralizzerà tutta la comunicazione.

### 5.3 Architettura fisica

Diversi modi per far collaborare differenti moduli.

#### 5.3.1 Layer e tier

Dividiamo i layer applicativi in 3 principali parti:

- **presentazione** (front end)
- **logica applicativa** (back end)
- **gestione dei dati** (back end)

Posso dividere i diversi tiers in diversi modi a seconda dell'architettura distribuita che scelgo:

- **Single Tiered**
- **Double tiered** = client-server :
  - **thin client** = client si preoccupa solo della parte di presentazione
  - **thick client** = client si preoccupa anche della parte applicativa.
- **Three Tiered** = teoricamente un tier per ciascun layer, anche se poi non è per forza così. Avrò quindi un middle tier.
- **N-tiered** = ulteriore distribuzione dell'architettura. Si suddividono i layer logici in un numero arbitrario di tier fisici (eventualmente implementati come server farm invece che come singole macchine), al fine di aumentare la scalabilità e la sicurezza del sistema. Una tipica configurazione in ambito web è quella a cinque tier: client – web server – script engine – application server – database server.

#### 5.3.2 Scalabilità

- Scalabilità verticale = potenziare la mia macchina, anche se spesso non è il migliore trade off e soprattutto prima o poi avrò limiti fisici. Il vantaggio è che non ho sbatti nella distribuzione delle applicazioni.
- Scalabilità orizzontale/scale out = sfrutto il principio di **downsizing** cioè a parità di potenza di calcolo, tanti server di fascia bassa sono più convenienti di un singolo server di fascia alta. <- necessito però un sistema di load balancing per gestire tutti i servers.

#### 5.3.3 Server Farm

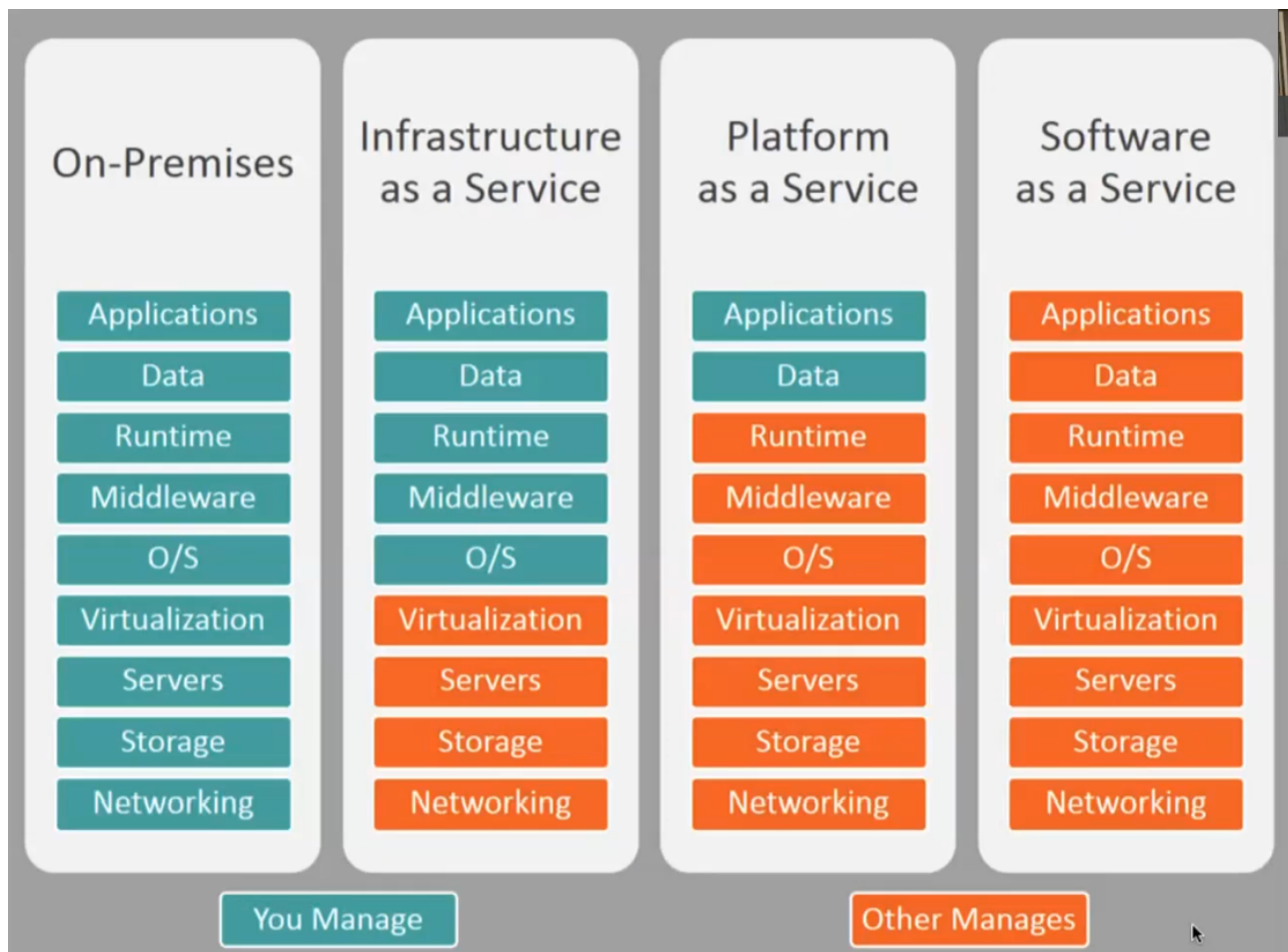
insieme di macchine fisiche che condividono il carico lavorativo. Le server farm hanno un alto livello di downsizing proprio per loro natura. Possibili progettazioni di una Server Farm:

- **RACS (Reliable Array of Cloned Services)**, cloning : a sua volta diviso
  - Shared - nothing = dati tutti replicati, ottimo per read-only intensive applications.
  - Cluster = shared - disk = le macchine condividono la memoria.
- **RAPS (Reliable Array of Partitioned Service)**, partitioning : cloning of the partitions: l'ibrido delle altre due, di base clono le singole partizioni per avere ridondanza.

#### 5.3.4 Cloud Computing

Architetture applicative accessibili da ovunque nel mondo attraverso la rete. Si basa sul concetto di **virtualizzazione**. Con il cloud computing abbiamo quindi un servizio on demand che ci permette di allocare - deallocare risorse in base alle nostre esigenze. E soprattutto pay as you go.

- **Infrastructure as a Service** = completo controllo delle VM.
- **Platform as a Service** = controllo delle proprie applicazioni
- **Software as a Service** = solo applicazione, posso quindi configurare solo i parametri di tale applicazione. (es: i vari cloud storage Dropbox, Onedrive, GDrive ecc. ecc.).



Il cloud potrà quindi essere:

- **Private Cloud** = solo una organizzazione
- **Community Cloud** = poche organizzazioni con interessi comuni
- **Public Cloud** = servizio fornito a tutto il pubblico. Costi molto limitati.
- **Hybrid Cloud** = una combo delle precedenti. Ogni azienda/attività potrà scegliere se gestire l'architettura fisica del SI internamente, parzialmente fuori azienda, esternamente affidandosi a un'offerta cloud che potrà essere IaaS, PaaS, SaaS.

### 5.3.5 Insourcing e Outsourcing

Si può anche possedere direttamente i server fisici o appoggiarsi a enti esterni (outsourcing) sempre facendo riferimento ai tipi di cloud e servizi possibili. Possiamo distinguere 6 livelli, dall'insourcing al outsourcing completo.

Tipologia	Livello	Caratteristiche
In house	0	Servizi sono gestiti internamente
<b>Servizi condivisi (Insourcing)</b>	<b>1</b>	<b>Un compagnia (unità) fornisce servizi IT a tutte le unità</b>
Supporto esterno	2	I servizi sono gestiti internamente, ma con il supporto di compagnie specializzate e società di consulenza
Consorzio	3	Un gruppo di compagnie con bisogni simili creano un consorzio specializzato per alcuni servizi (e.g., call center, data center)
<b>Outsourcing selettivo</b>	<b>4</b>	<b>Alcuni servizi, solitamente non core, sono gestiti da aziende esterne</b>
<b>Outsourcing completo</b>	<b>5</b>	<b>Tutto l' IT è gestito da una ditta esterna. Se l'infrastruttura appartiene al cliente il servizio è chiamato <i>facility management</i></b>

Ricordati che l'insourcing/outsourcing è diverso da make/buy

## 5.4 Sicurezza dei sistemi informativi

Sicurezza per i dati, la nostra risorsa più preziosa. Dobbiamo proteggere i dati da minacce fisiche, logiche e accidentali (magari bug di softwares o errori umani).

### 5.4.1 Proprietà della sicurezza

La sicurezza ha 4 proprietà fondamentali:

- **Integralità:** il messaggio non viene modificato nel suo tragitto mittente-destinatario.
- **Autenticità:** il messaggio deve dimostrarmi la sua autenticità.
- **Riservatezza:** il messaggio deve arrivare solo al destinatario.
- **Disponibilità:** il sistema deve sempre garantire disponibilità all'utente.

### 5.4.2 Attacchi di rete

- **Hijacking:** man in the middle (attacca l'integrità)
- **Spoofing:** ci si finge qualcun altro (attacca l'autenticità)
- **Sniffing:** si ascolta in modalità passiva i messaggi (attacca riservatezza)
- **Flooding:** attacchi DOS (attacca la disponibilità)

### 5.4.3 Attacchi Applicativi

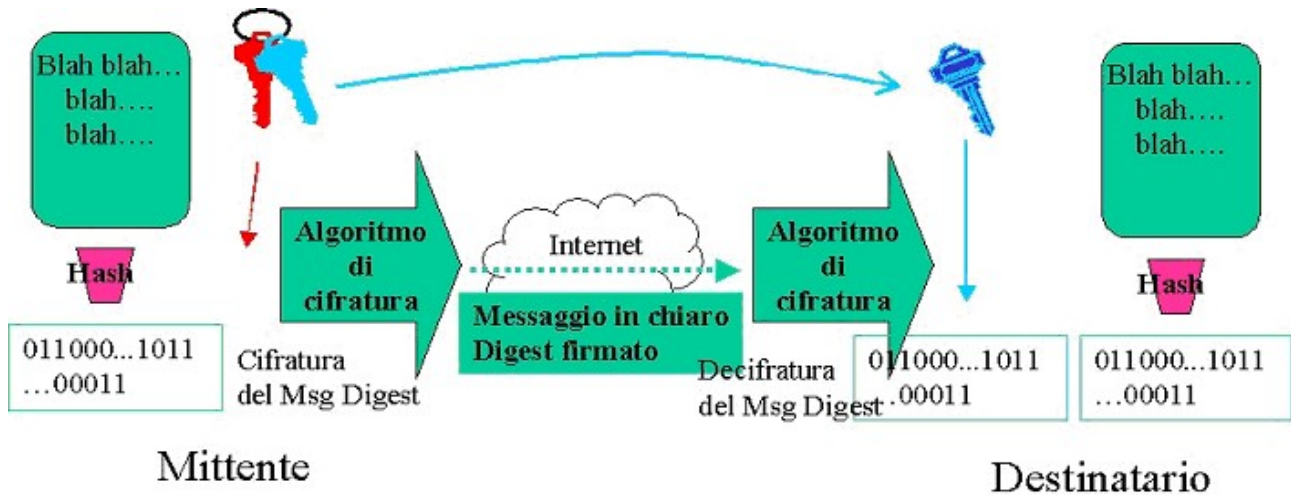
- trojan
- backdoor
- ransomware
- spyware

#### 5.4.4 Crittografia simmetrica e asimmetrica

- **Chiave simmetrica:** si usa una chiave che ha sia A e B per cifrare i messaggi
- **Chiave asimmetrica:** la chiave non viene mai scambiata ma i messaggi si cifrano comunque, usando due chiavi:
  - chiave privata
  - chiave pubblica Solo la chiave pubblica può decifrare un messaggio generato dalla privata.

#### 5.4.5 Impronta e firma digitale

La firma digitale é utilizzata per verificare l'integralità e consiste in una funzione di hash. > La firma digitale é la cifratura da parte del mittente (con la sua chiave privata) del digest del messaggio.



La firma digitale la verifico con la chiave pubblica. Il senso è che una volta verificato il messaggio con la chiave pubblica, solo l'ente che possiede la chiave privata può aver scritto quel messaggio (l'hash del messaggio è unico).

#### 5.4.6 Certificati digitale

Garantisce l'autenticità della chiave pubblica. I certificati vengono gestiti da sistemi detti **Public Key Infrastructure** (PKI), che si occupano di emettere e revocare i **PKC** (Certificato a Chiave Pubblica) attraverso certificate authority **CA** e registration authority **RA**. Tali enti hanno strutture gerarchiche ad albero per garantire solidità. La Certification Authority rilascia certificati non la Registration Authority.

Un fattore critico dunque é come conservare tali chiavi private:

- hardware
- software ma anche come scambiarsele, prima di utilizzarle:
- out-of-bound: si usa un canale esterno al canale utilizzato per comunicare
- ente esterno
- crittografia asimmetrica: si usa infatti l'asimmetrica (basata sulle chiavi pubbliche) per scambiarsi il messaggio contenente le chiavi private in modo sicuro.

#### 5.4.7 Regole e politiche d'accesso

Due possibili classificazioni dei sistemi in base a come viene gestita la sicurezza:

- **sistema aperto:** tutto ciò che non é esplicitato si può fare
- **sistema chiuso:** tutto ciò che non é esplicitato non si può fare

Oltre a questa macro classificazione vengono anche considerati altri livelli di sicurezza. Vengono effettuati diversi controlli d'accesso basati su **regole**. Le regole d'accesso da parte di autorità seguono tutte lo schema: (subject,object,right,constraint) . Oltre a queste regole d'accesso possiamo poi definire delle politiche di accesso:

#### 5.4.8 Politiche d'accesso

- 1) **DAC** - Discrezionale, cioè il proprietario di un oggetto può dare il permesso di modificare tale oggetto
- 2) **MAC** - Mandatorio, tutto è classificato automaticamente da un ente centrale. I proprietari non possono autorizzare altri proprietari.

Le regole d'accesso coinvolgono:

- soggetti:
  - persone,gruppi,app
- oggetti:
  - databases, tabelle varie
- diritti:
  - tipi di azioni consentiti
- vincoli:
  - ulteriori regole e condizioni da applicare ai diritti

**5.4.8.1 MAC** Mentre il DAC si basa sul concetto di proprietà, il mandatorio si basa tutto sulla classificazione di oggetti e soggetti. Idea generale: Tipi di classificazione degli oggetti:

- unclassified
- confidential
- secret
- top secret

2 regole:

- 1) **NO READ UP**: Ogni soggetto può vedere solo gli oggetti di livello uguale o inferiore.
- 2) **NO WRITE DOWN**: Ogni soggetto può scrivere solo nelle classi maggiori o uguali alla sua.

Viene quindi mantenuto un database multi livello, che mostra i dati in base alla classe di appartenenza.

#### 5.4.9 Firewall

Fondamentale per filtrare l'unico punto di contatto con il mondo esterno. Il Firewall però deve essere configurato correttamente:

- Packet inspection/filtering
- Application Gateway

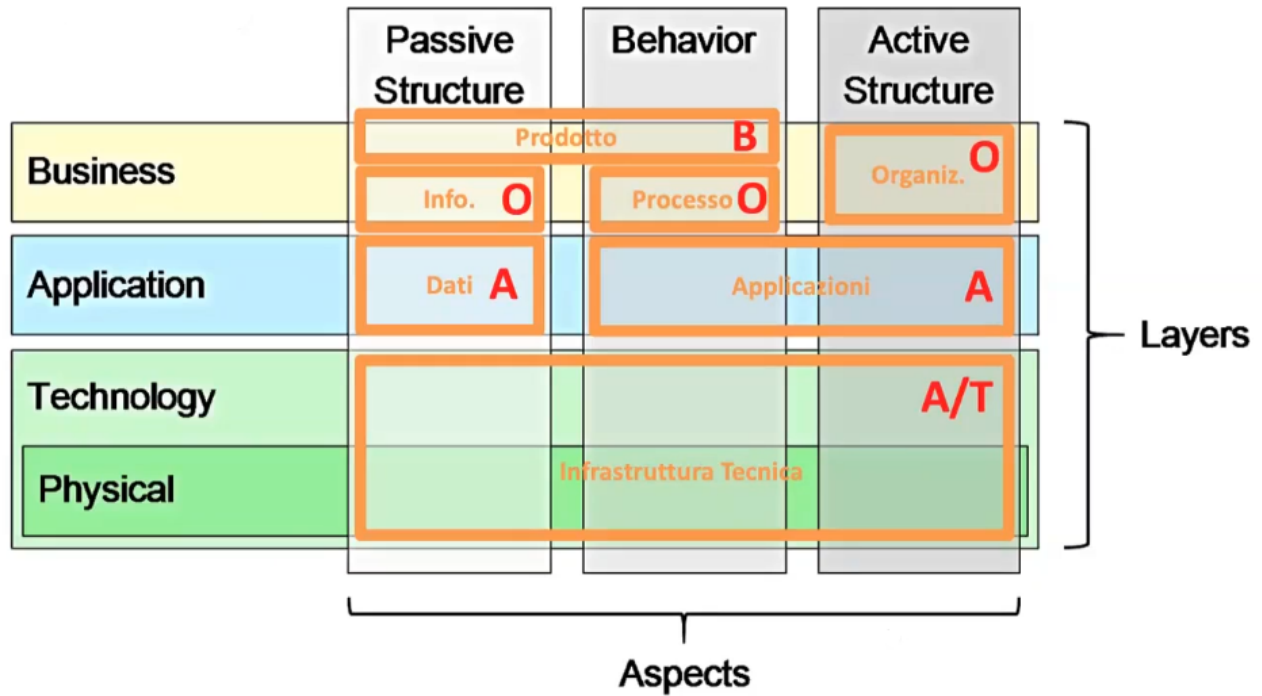
**5.4.9.1 IDS - Intrusion Detection System** sensori che continuano a monitorare attraverso log. E ulteriori analisi sui vari dati raccolti. Tutti gli IDS fanno funzione di **alerting and response** ma si distinguono se hanno una risposta attiva o semplicemente avvisano e basta.



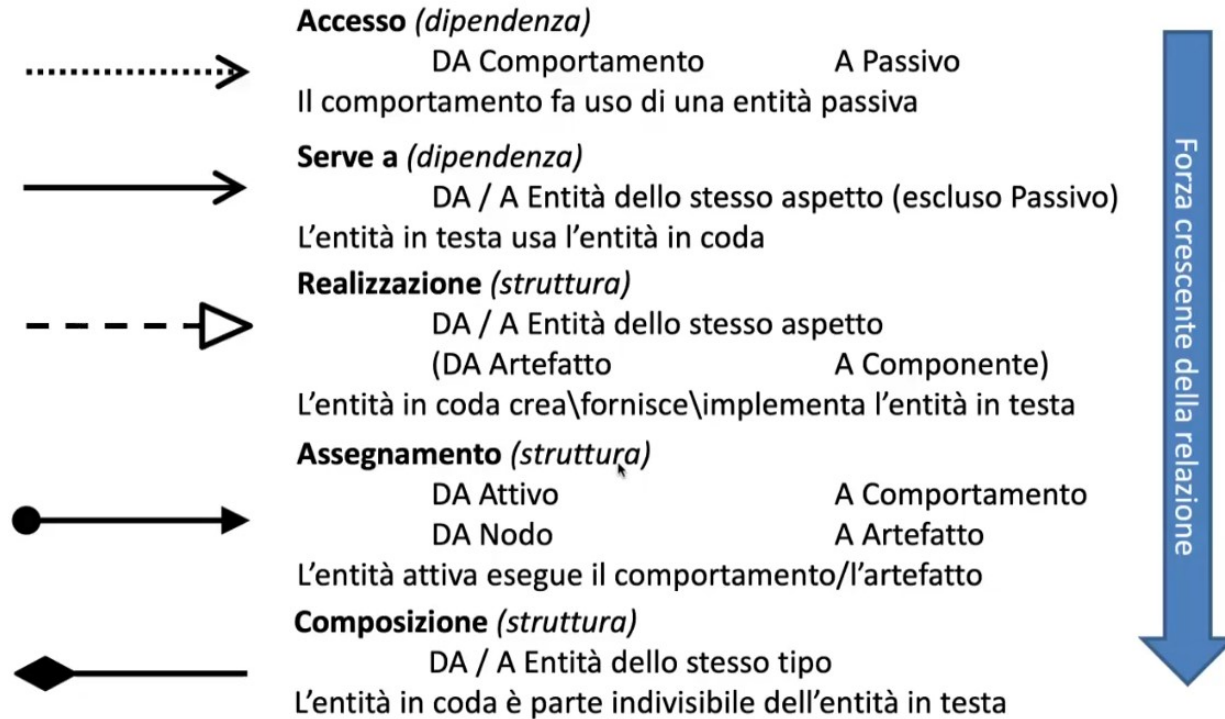
## 5.5 Archimate

# Archimate Core – Domini

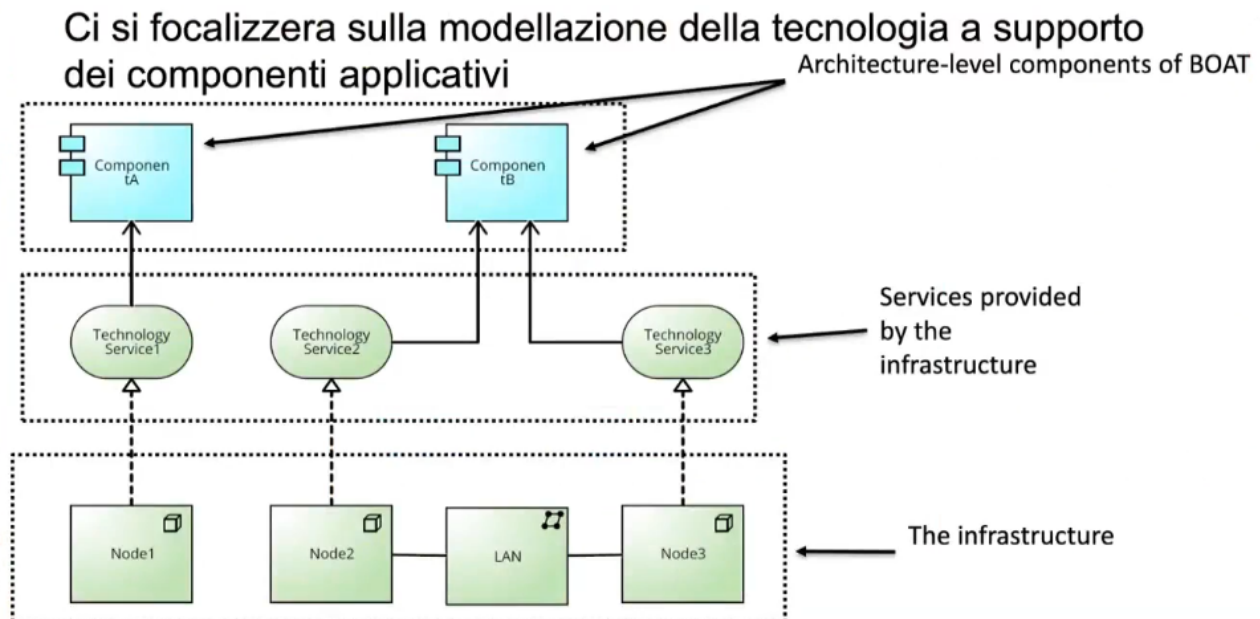
## *Confronto con BOAT*



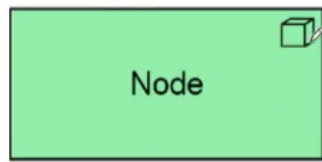
# Relazioni



Punto di contatto tra Archimate e BOAT:



Il device è **fisico** .



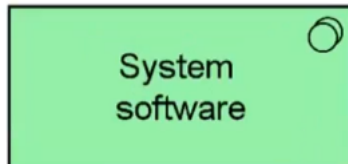
- Risorsa computazionale o fisica che ospita, manipola o interagisce con altre risorse fisiche o computazionali



- Servizio offerto dalla tecnologia



- Risorsa IT fisica dove è possibile memorizzare ed eseguire software applicativo



- Software che permette la memorizzazione di dati e/o l'esecuzione di altro software applicativo



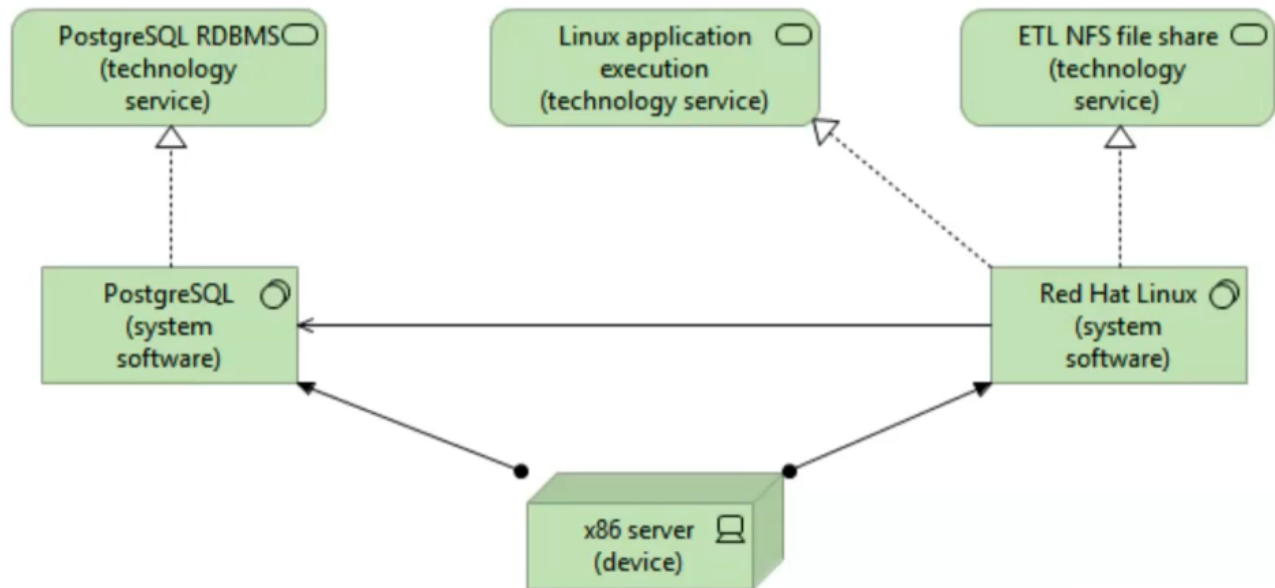
- Collegamento fisico o virtuale diretto (senza intermediari) tra due nodi, tramite il quale è possibile scambiare informazioni e accedere a risorse. Si usa per comunicazioni point-to-point.



- Insieme di apparati e protocolli che permette il collegamento tra computer o altri dispositivi elettronici, consentendo trasmissione, instradamento e ricezione di dati. Si usa per comunicazioni con topologia di rete complessa.

Attenzione i collegamenti (path) non sono linee! Stiamo ragionando a livello fisico .. la fibra è diversa dal ethernet o dallo spago!

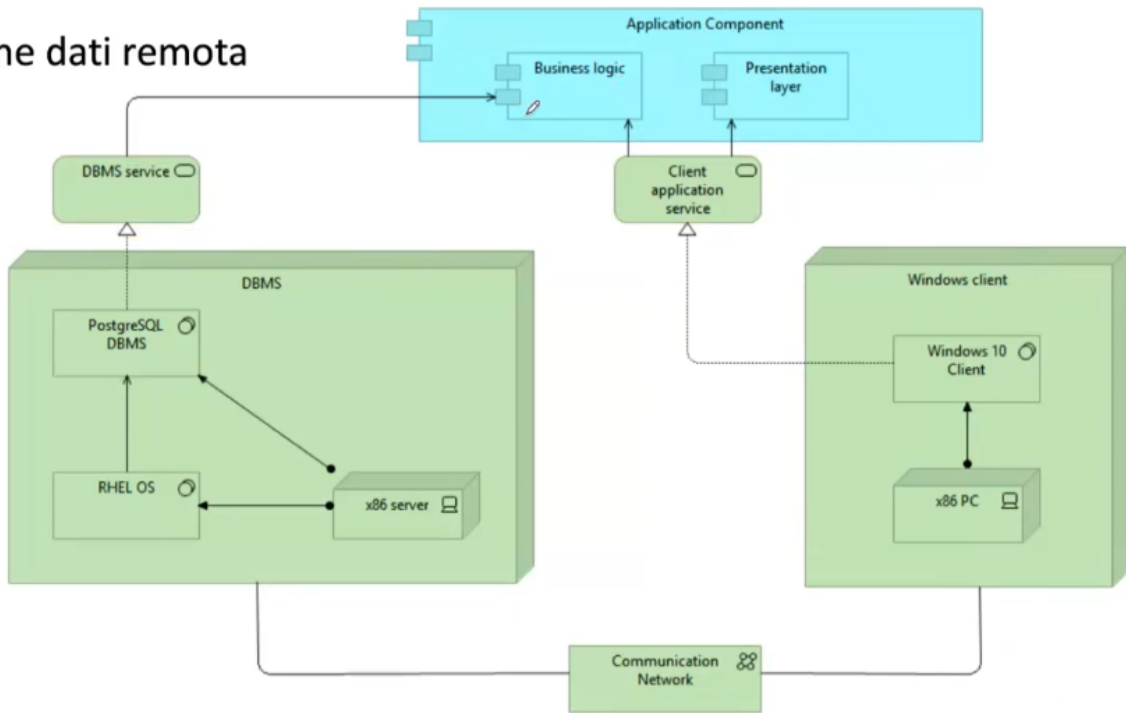
Esempio di DW scritto in Archimate:



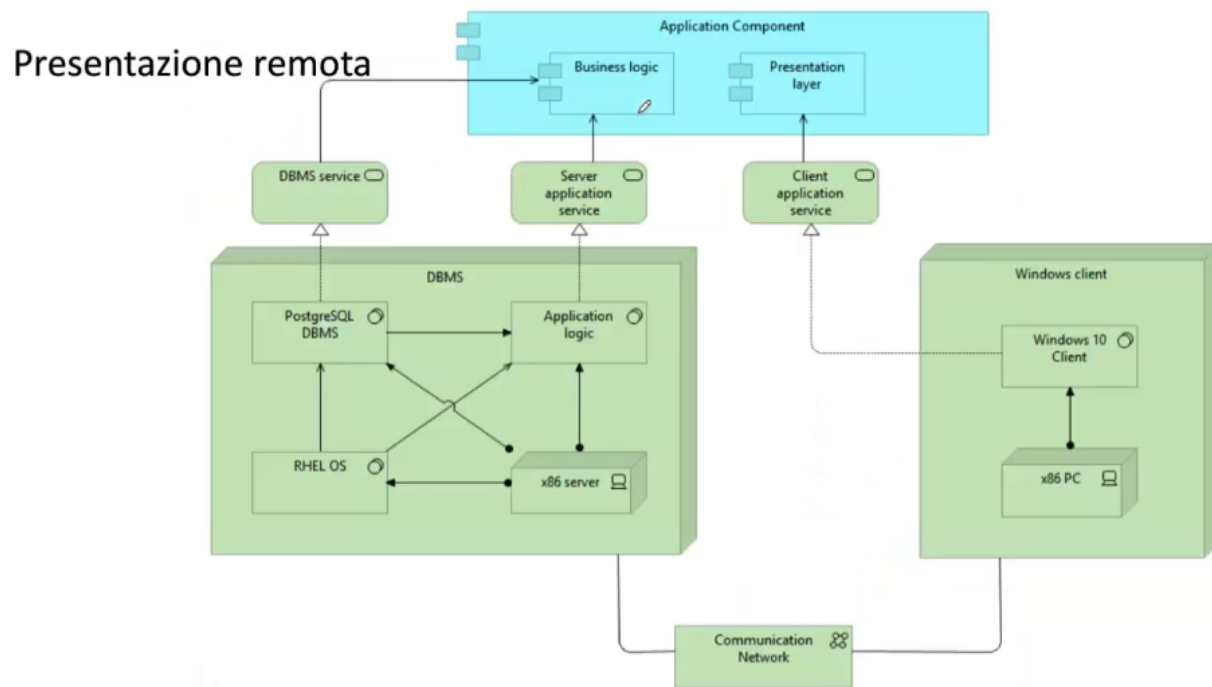
(normalmente all'esame specifichiamo bene la struttura interna del nodo server). Tipi di Double-Tiered:

# 2-tier thick client

Gestione dati remota



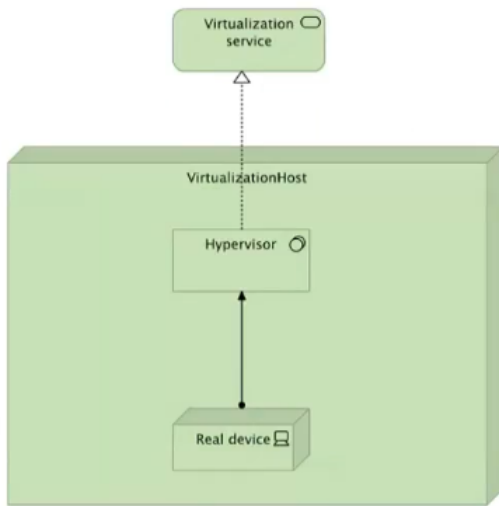
# 2-tier thin client



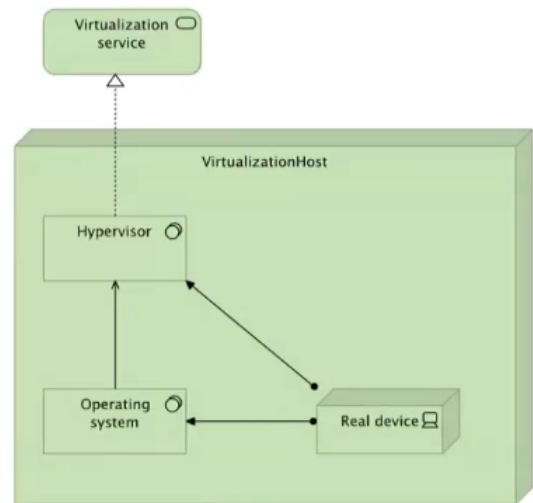
Tipi di virtualizzazione:

# Virtualizzazione

- Bare metal

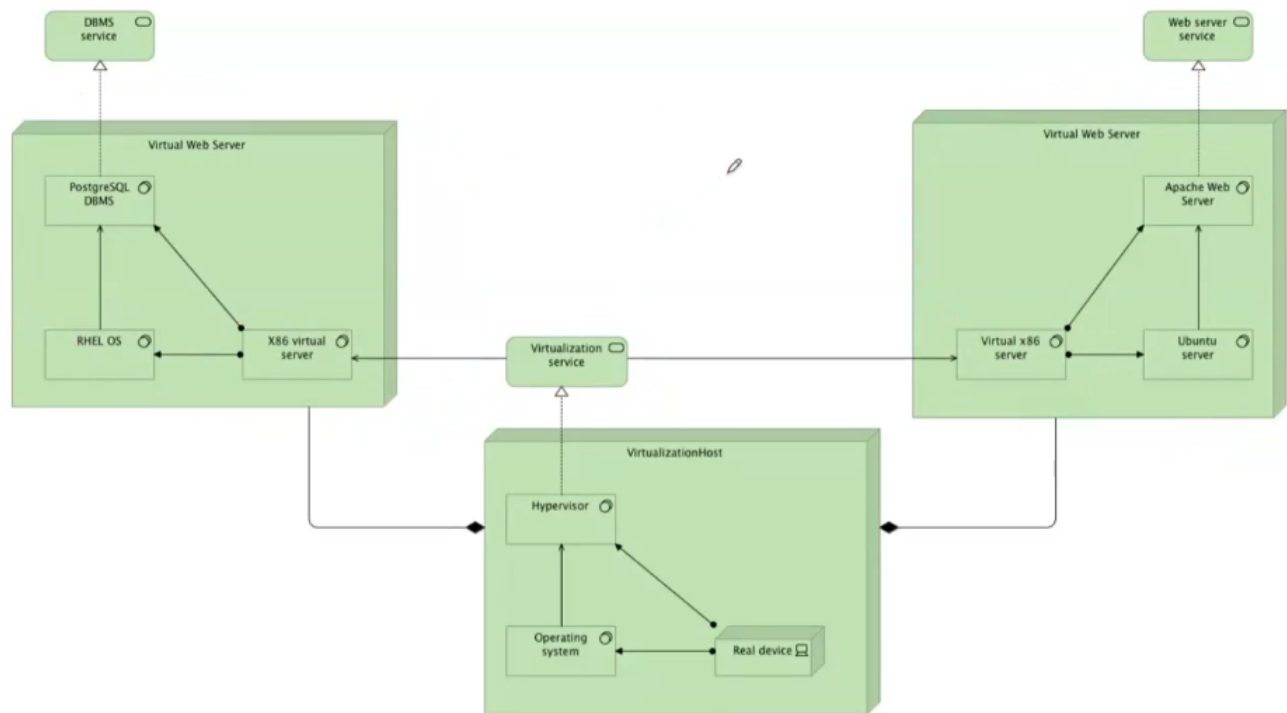


- Hosted



Classico server virtuale:

# Server virtuale



Struttura standard per punto di contatto internet/rete aziendale



