



Argomenti e strumenti dell'insegnamento di "Informatica" del Corso di Studi in "Scienze e Tecnologie per i Beni Culturali" - Anno Accademico 2014-2015

Prof. Giorgio Poletti

Dipartimento di Studi Umanistici - Via Paradiso 12, Ferrara

mail: giorgio.poletti@unife.it

homePage docente: <http://docente.unife.it/giorgio.poletti>

Sommario

Obiettivi dell'insegnamento	3
Contenuti dell'insegnamento	3
Testi di riferimento	3
Tools di sviluppo	4
Modalità d'esame	4
Appunti di teoria dei Grafi	5
Teoria dei Grafi e Rappresentazione dei Problemi.....	5
Premesse generali	5
<i>Definizioni</i>	8
Metodi di memorizzazione di un grafo	14
Grafi e rappresentazione dei problemi	16
Le reti di Petri	25
Introduzione ed elementi costitutivi	25
Proprietà e tipologie delle P-reti	29
<i>Raggiungibilità e grafo di raggiungibilità</i>	30
<i>Limitatezze e reti safe</i>	30
<i>Altre proprietà e caratteristiche delle reti di petri e proprietà decidibili</i>	32
<i>Estensioni delle reti di Petri e reti temporizzate</i>	34
<i>Tipologie notevoli di P-reti</i>	37
Problemi ed esempi per le reti di Petri.....	39
Il WEB 2.0: filosofia e strumenti	41
Premessa	41
Ipertesto, storia e definizioni.....	42
Evoluzione del WEB: verso il Web Semantico	47
WEB 2.0: strumenti e filosofia di utilizzo.....	50
Indice delle Figure	55
APPENDICE - Slide del corso	57
Lezione 1.....	57
Lezione 2.....	65
Lezione 3.....	73
Lezione 4.....	83
Lezione 5.....	98
Lezione 6.....	111
Lezione 7.....	127

Obiettivi dell'insegnamento

- i. **Comprendere** e **utilizzare** in modo appropriato il linguaggio e i lemmi dell'informatica
- ii. **Utilizzare** i fondamenti della logica proposizionale per la formalizzazione di strutture comunicative.
- iii. **Utilizzare** la teoria dei grafi nell'ambito della progettazione di strumenti di comunicazione interattiva.
- iv. **Individuare** e **gestire** gli strumenti software di supporto alla propria attività di ricerca e di lavoro.
- v. **Identificare** gli ambienti e i linguaggi di programmazione idonei a risolvere le diverse problematiche di gestione delle informazioni.
- vi. **Apprendere** e **applicare** metodologie ed elementi di linguaggio per creare prototipi di supporti alla comunicazione, con una speciale attenzione al Web e alle specifiche applicazioni ai beni culturali.
- vii. **Imparare** a progettare e realizzare strumenti efficaci per la comunicazione, in funzione dell'utenza a cui ci si rivolgono e al tipo di contenuto che si intende veicolare.

Contenuti dell'insegnamento

1. Grafi e logica della strutturazione

- 1.1. Teoria dei grafi
- 1.2. Grafi e rappresentazione dei problemi
- 1.3. Mappe concettuali e mappe cognitive
- 1.4. Soluzione vs Risoluzione
- 1.5. I quattro problemi fondamentali
- 1.6. Elementi di teoria delle reti di Petri
- 1.7. Rappresentazione di processi non deterministici
- 1.8. Dalle reti aleatorie alle reti scale free

2. Linguaggi e strumenti del Web 2.0

- 2.1. Il Web e l'ipertesto: storia e struttura
- 2.2. Verso il Web semantico
- 2.3. Elementi e caratteristiche del Web 2.0
- 2.4. Strumenti di sviluppo e comunicazione nel Web 2.0
- 2.5. La rete da medium a format
- 2.6. Wiki, Blog e social software
- 2.7. Mobile App e beni culturali

3. Applicazioni di realtà aumentata

Testi di riferimento

- i. Padula M., Reggiori A., *Fondamenti di informatica per la progettazione multimediale - Dai linguaggi formali all'inclusione digitale*, Collana di Informatica, 2013, Franco Angeli Editore, Milano
- ii. Appunti scaricabili dal mini-sito del corso
(<http://www.unife.it/scienze/beni.culturali/insegnamenti/informatica>)
- iii. Strumenti e approfondimenti: **piattaforma FAD (Frequenza A Distanza)**: bcq.unife.it
con **iscrizione**: fad.unife.it/index.php/component/chronoforms/?chronoform=home_tuttofad

Tools di sviluppo

- **PBwiki** (<https://my.pbworks.com/>)- tool on line per creare wiki (login page):
 - Username: tecnostudenti@gmail.com
 - Password: studentiUniFe
- **COMO** (<http://www.como.com/>)- APP editor on-line, dalla voce di menu <SIGN IN> (Sign in with Email), è possibile crearsi un account free anche accreditandosi utilizzando un account Facebook
 - Email: tecnostudenti@gmail.com
 - Password: studentiunife
- **AURASMA** - piattaforma di realtà aumentata (login ad Aurasma studio - <https://studio.aurasma.com/login>)
 - Username: studentiUniFe
 - Password: UniFestudenti
- **Thinglink** - piattaforma per immagini interattive (<https://www.thinglink.com/>)
 - ci si può registrare free, anche utilizzando l'account Facebook o Google+, oppure inviare una mail a giorgio.poletti@unife.it per essere inseriti in una classe della versione "teach" e ricevere le credenziali
- **Padlet** - Bacheche on line (<https://it.padlet.com/>)
 - Username: tecnostudenti@gmail.com
 - Password: studentiunife
- **Gmail** - account di posta da cui accedere a Google Site, Blogger e Googel Drive
 - Username: tecnostudenti@gmail.com
 - Password: studentiunife
- **Dipity**- editor di TimeLine - (<http://www.dipity.com/>)
 - Username: studenti_unife
 - Password: studentiunife

Modalità d'esame

L'esame è strutturato in 2 parti

- prova pratica**: produzione di un Prodotto multimediale
- test scritto**: 30 items, 27 chiusi (a scelta multipla, una risposta esatta su 4 possibili) + 3 quesiti a risposta aperta.

La prova pratica consiste nello strutturare e implementare un prodotto o strumento multimediale con i tools analizzati e utilizzati durante il corso, che sono software per la produzione e la distribuzione on line di wiki, App, e-book, siti e blog.

Il contenuto di tali applicazioni è concordato con il docente e la consegna del materiale, ovvero dei link di pubblicazione del prodotto, deve avvenire entro la data scelta per il test scritto.

La valutazione finale dell'esame è definita nel modo seguente:

- prova pratica**: 15% del voto finale (max. 4,5 punti)
- test scritto**: 85% del voto finale (max. 25,5 punti), per la valutazione del test scritto si attribuisce 1 punto per ogni risposta corretta e 0 punti per ogni risposta errata o non data.

La prova pratica, se valutata, è utilizzabile per accedere al test scritto fino alla sessione straordinari di esami dell'anno accademico di riferimento del corso.

Appunti di teoria dei Grafi

Teoria dei Grafi e Rappresentazione dei Problemi

Premesse generali

Un **grafo** è una struttura relazionale composta da un insieme finito di oggetti, sul piano più strettamente matematico un insieme finito di punti, detti “**nodi**” o “**vertici**” e da un insieme di relazioni, geometricamente definibili **segmenti di retta** o **di curva** detti “**archi**” o “**spigoli**”, che uniscono coppie di oggetti

La definizione di grafo ha, da un punto di vista strettamente matematico, diverse notazioni e tra quelle possibili utilizziamo la seguente:

$$G(N,A)$$

N indica l'insieme dei nodi e A indica l'insieme degli archi che compongono il grafo G

L'importanza che assume questa struttura logica è da attribuire alla constatazione che più di una “**struttura reale**” o “**struttura concreta**” può essere schematizzata utilizzando i grafi.

Tra le strutture che ben sono rappresentabili con una struttura a grafo ci sono le reti stradali dove i nodi rappresentano luoghi e gli archi strade che li congiungono, un programma di calcolo dove i nodi rappresentano le istruzioni ed gli archi le relazioni che esistono tra l'esecuzione di due istruzioni oppure una struttura dati dove i nodi rappresentano i dati semplici e gli archi i legami tra i diversi dati; i legami tra i dati possono poi essere realizzati tramite puntatori o formalizzabili attraverso relazioni semanticamente definite¹.

L'origine storica della teoria è generalmente fatta risalire ad un lavoro sviluppato da Eulero² nel 1736 in cui veniva data una risposta ad un famoso quesito matematico noto come **problema dei ponti di Königsberg**³.

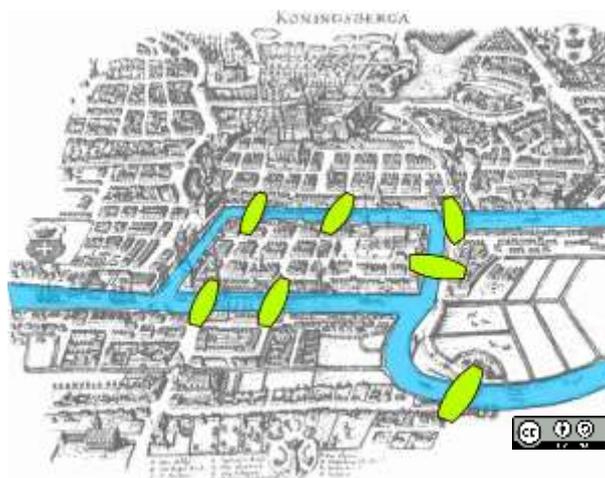


Figura 1 - Mappa di Königsberg ai tempi di Eulero, in evidenza la disposizione dei ponti sul fiume Pregel e la disposizione dei diversi quartieri della città (immagine tratta da Wikipedia - CC BY-SA 3.0)

¹ Relazioni di cui sono non solo noti i termini e le regole, e per le quali è dichiarato anche il significato che tale relazione esplicita.

² Eulero è considerato il matematico che più di ogni altro ha caratterizzato il movimento culturale dell'Illuminismo. Il suo nome è **Leonhard Euler**, nato a Basilea, 15 aprile 1707 e morto a San Pietroburgo, 18 settembre 1783. Allievo di Johann Bernulli, Eulero è autore di un grande numero di scritti e trattati.

³ Il “**problema dei ponti di Königsberg**” viene definito e risolto da Eulero in un suo scritto del 1734, “*Solutio problematis ad geometriam situs pertinentis*”, che viene per questo considerato il primo trattato scientifico sulla teoria dei grafi.

Königsberg era una ricca e popolosa città della Prussia e dal 1245 fù capoluogo della Prussia Orientale e tra i principali centri culturali e politici della Germania. Oggi la città si chiama Kalinigrad ed è in Russia, capitale dell'Oblast di Kalinigrad, un territorio situato tra Polonia e Lituani (cfr. Figura 2); la città si sviluppa sulle rive del fiume Pregel e sulle due isole formate dal fiume in quell'area (cfr. Figura 1), e che nel 1700 era attraversato da sette ponti. I ponti della città erano disposti in modo tale che la prima isola (indicata convenzionalmente A) era collegata con due ponti a ciascuna delle due rive (indicate rispettivamente in maniera convenzionale C e D), mentre la seconda isola (indicata convenzionalmente B) era collegata con un solo ponte a ciascuna riva ed inoltre vi era un ponte che collegava A e B.

Questa astrazione geometrica genera uno schema in cui le isole della città, così come le definisce il fiume Pregel, sono rappresentate da punti e i ponti da archi che le congiungono (cfr. Figura 3).



Figura 2 – Posizione geografica di Kalinigrad (immagine trattata dal sito www.aebr.eu)

Il problema dei sette ponti di Königsberg, alla luce delle definizioni e delle convenzioni descritte può essere enunciato nel modo seguente:

*Come in ogni cittadina tedesca, era uso anche a Königsberg, che alla domenica gli abitanti facessero una passeggiata per le vie della città.
La domanda che ci si poneva era: "è possibile progettare un percorso che permetta ad ogni abitante della città prussiana di partire da casa e tornarvi dopo aver attraversato ciascun ponte una ed una sola volta?"*

Eulero, come nello schema rappresentato in Figura 3, per dimostrare che il problema non aveva soluzioni partì dal rappresentare ogni quartiere, sia quelli situati sulle rive del fiume che quelli situati sulle isole, con un nodo ed ogni ponte con un arco; in tal modo si trasforma il processo della soluzione del problema nell'analisi e nello studio *topologico*⁴ del grafo così costruito.

⁴ La topologia o studio dei luoghi, dal greco **τοπος** (*topos*: luogo) e **λογος** (*logos*: studio) è la branca della matematica che si occupa dello studio delle proprietà delle figure e delle forme; in particolare si occupa dello studio dell'invarianza di tali proprietà quando le figure e le forme vengono deformate senza 'strappi', 'sovrapposizioni' o 'incollature'. Il

Reiterando il procedimento di schematizzazione così definito, Eulero ha prima analizzato le diverse configurazioni possibili osservando che il problema aveva soluzione nei casi in cui ogni nodo ha grado pari, cioè è pari il numero di archi che incidono sul nodo stesso, e non ammetteva soluzioni nel caso in cui almeno un nodo avesse grado dispari. In virtù di queste considerazioni Eulero enunciò il suo teorema che assumeva in questa prospettiva una valenza universale.

Teorema di Eulero

Condizione necessaria e sufficiente affinché un grafo sia percorribile completamente partendo da un nodo e ritornandovi passando una volta solamente per ciascun arco è che esista un percorso fra ogni coppia di nodi e che ogni nodo sia toccato da un numero pari di archi (su ogni nodo incidano un numero pari di archi).

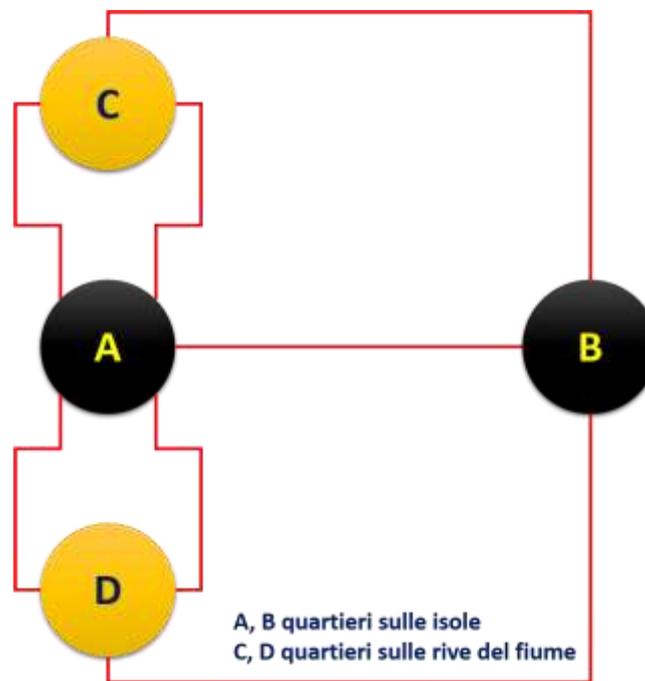


Figura 3 - Rappresentazione del problema attraverso l'astrazione con enti geometrici della mappa reale della città (i nodi del grafo) e della disposizione dei ponti (gli archi del grafo).

Gli studi di Eulero hanno portato a dimostrare che un tipo di cammino con le caratteristiche descritte nel *problema dei ponti di Königsberg* è possibile *se e solo se* il grafo non ha nodi di grado dispari. Un cammino con le caratteristiche enunciate nel problema è detto **ciclo euleriano**; si parla invece di cammino euleriano nel caso in cui non si richieda che il punto di inizio e di fine coincidano e in quel caso si possono avere o nessuno o due nodi di grado dispari. Un grafo che ammette almeno un ciclo euleriano è detto un grafo euleriano.

In modo rigoroso si afferma che:

Dato un grafo $G = (N,A)$ se esiste un cammino di G che percorre tutti gli archi di G una e una sola volta si dice che G è un **grafo euleriano**; se esiste almeno un cammino euleriano di G , si dice che G è un grafo "**semieuleriano**" e per questo è conseguente affermare che **ogni grafo euleriano è anche semieuleriano**.

teorema dei ponti di Königsberg e il relativo approccio alla soluzione di Eulero è considerato uno dei primi esempi topologici, poiché il risultato non dipende da alcun tipo di misurazione.

Nella prima metà dell'Ottocento **Kirchhoff**⁵ e **Cayley**⁶ svilupparono importanti applicazioni della teoria dei grafi, il primo per quanto concerne lo studio delle reti elettriche, il secondo nell'ambito degli isomeri chimici.

In sintesi la "teoria dei grafici" di Cayley permette il calcolo del numero degli isomeri possibili quando si tratta di isomeri strutturali, mentre Kirchhoff dopo studi ed esperimenti empirici enunciò le leggi che portano il suo nome, le leggi di Kirchhoff, che permettono di sostituire completamente, nei circuiti a parametri concentrati⁷, le leggi dell'elettromagnetismo riducendole essenzialmente in un insieme di pure relazioni topologiche, cioè lo studio delle proprietà di figure e forme.

L'informatica, infine, pone particolare interesse a tale teoria: è evidente se si pensa ai grafi come strumenti indispensabili per una progettazione ed una rappresentazione efficiente di reti di computer, le strutture di data base e moltissime altre strutture fisiche e logiche, come le importantissime mappe concettuali e la relativa rappresentazione della conoscenza.

L'informatica, infine, pone particolare interesse a tale teoria: è evidente se si pensa ai grafi come strumenti indispensabili per una progettazione ed una rappresentazione efficiente di reti di computer, le strutture di data base e moltissime altre strutture fisiche e logiche, come le importantissime **mappe concettuali**⁸ e la relativa rappresentazione della conoscenza.

Definizioni

Diviene fondamentale, per affrontare i problemi relativi la teoria dei grafi e le relative applicazioni, disporre di alcune definizioni di base che permettano di descrivere le diverse tipologie di grafo e le proprietà che caratterizzano questa struttura relazionale.

Le definizioni che verranno enunciate si riferiscono essenzialmente a due tipologie di approccio al contesto della teoria dei grafi:

- **descrizione** di un grafo e dei suoi elementi;
- **classificazione** di un grafo per tipologia.

In generale, gli archi rappresentano una relazione fra una coppia di nodi; se tale coppia è ordinata, cioè se gli archi hanno una testa o nodo di arrivo ed una coda o nodo di partenza, il grafo si dice orientato, dove **(i,j)** indica *un arco diretto dal nodo i al nodo j*.

In un grafo orientato, presa la coppia di nodi **(i,j)**, **i** è detto **predecessore** di **j** e **j** è detto **successore** di **i** (vedi Figura 4).

⁵ **Gustav Robert Georg Kirchhoff**, è un fisico e matematico tedesco, nato Königsberg il 12 marzo 1824 e morto a Berlino il 17 ottobre 1887. Sviluppò la sua ricerca scientifica nei campi delle termodinamica, dell'elettrotecnica e delle spettrografia. Ai suoi studi e alle sue intuizioni si devono, tra le altre, due celebri leggi per il calcolo della distribuzione delle correnti elettriche in una rete di conduttori, la prima delle quali stabilisce che in un nodo di un circuito elettrico (schematizzabile come grafo) la somma algebrica delle intensità di corrente è nulla. Con il chimico **Robert Bunsen**, fu il primo scienziato a ricevere, nel 1877, il riconoscimento scientifico che annualmente viene conferito dall'Accademia delle Scienze britannica, la Royal Society, la medaglia Devy.

⁶ **Arthur Cayley** è un importante matematico inglese, nato a Richmond upon Thames il 16 agosto 1821 e morto a Cambridge il 26 gennaio 1895, che diede contributi significativi nel campo della matematica pura.

⁷ Il "circuito a parametri concentrati" è un circuito elettrico "abbastanza piccolo" per cui sia possibile trascurare il tempo di propagazione del segnale e la tensione e la corrente risultano ben definiti in ogni punto del circuito ed in ogni istante. In questo contesto le **leggi di Kirchhoff** approssimano in maniera più che soddisfacente le leggi dell'elettromagnetismo di James Clerk Maxwell.

⁸ Le **mappe concettuali** sono uno strumento che permettono la rappresentazione grafica sia delle informazioni che della conoscenza. Questa metodologia è stata teorizzata da **Joseph Novak**, oggi professore emerito alla Cornell University e ricercatore senior al **IHMC** (Institute for Human and Machine Cognition), che negli anni settanta le immaginò con una struttura di tipo *connessionista* e come strumento di *rappresentazione della propria conoscenza* con un taglio cognitivo di tipo *costruttivista*.

Nel caso in cui tutte le coppie di nodi presenti nel grafo non siano ordinate il grafo si dice “non orientato”.



Figura 4 - Schema e definizioni di un arco orientato

La classificazione dei grafi in relazione agli archi che insistono su una coppia di nodi porta alla classificazione di **grafi semplici** e **multigrafi**.

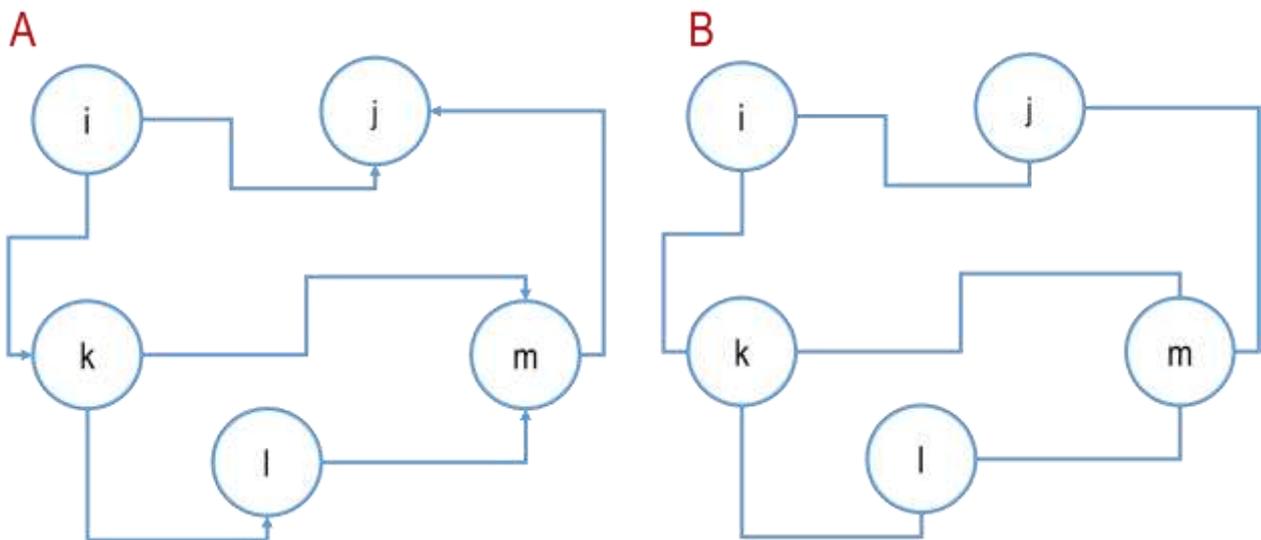


Figura 5- Grafo semplice orientato (A) e grafo semplice non orientato (B)

In particolare si parla di **grafo semplice** quando per ogni coppia di nodi del grafo esiste al massimo un arco che li unisce (vedi Figura 5), se invece esiste più di un arco tra almeno una coppia di nodi del grafo si parla di **multigrafo** (vedi Figura 6).

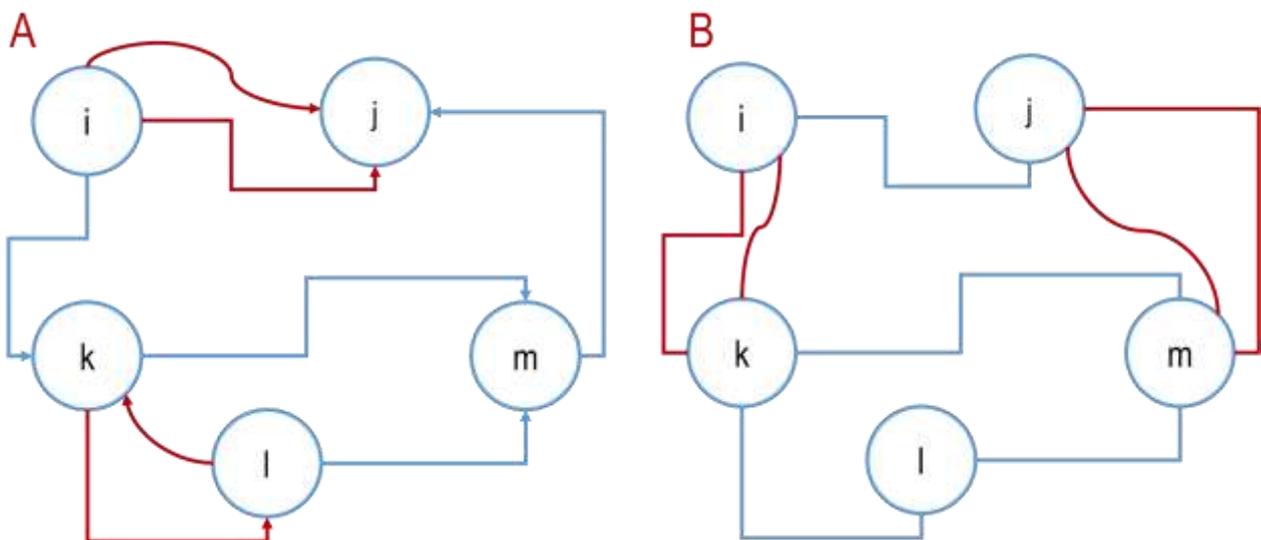


Figura 6 - Multigrafo orientato (A) e multigrafo non orientato (B)

Prendendo ora in considerazione i nodi e gli archi che li uniscono, in un **grafo non orientato**, un **nodo** si dice **adiacente** ad un altro nodo del grafo se esiste almeno un arco che li congiunge; se un nodo non ha nodi adiacenti è detto **nodo isolato**.

Sulla base di queste definizioni si definisce un grafo, sia orientato che non orientato, **completo** se per ogni coppia di nodi esiste un arco che li congiunge.

Prendendo poi in considerazione, in un grafo non orientato, il numero di **archi incidenti⁹ su di un nodo** è detto **grado del nodo o cardinalità**.

Se tutti i nodi di un grafo hanno lo stesso grado il **grafo** è detto **regolare**, in particolare un grafo regolare di grado **3** è detto **grafo cubico**.

Poiché nodi e archi di un grafo rappresentano insiemi di entità si può definire il concetto di **sottografo** e **grafo parziale**.

Dato un **grafo non orientato** $G(N,A)$ ed un **grafo non orientato** $H(V,E)$ con V sottoinsieme di N ($V \subseteq N$) ed E sottoinsieme di A ($E \subseteq A$), tale che gli archi di E sono tutti e solo quelli aventi entrambi i nodi terminali in V , H è detto **sottografo** di G ; se, invece, E è strettamente un sottoinsieme degli archi di A ($E \subset A$) aventi entrambi i nodi terminali in V , H è detto **grafo parziale** di G . Con tali premesse si definisce che un grafo G **copre** un grafo H se H è **grafo parziale** di G (vedi Figura 7).

Nella teoria dei grafi, in relazione al loro utilizzo per la schematizzazione dei problemi e ricerca di soluzioni, è importante approcciare i concetti relativi alla percorribilità di un grafo, cioè l'identificazione di una sequenza di archi che uniscono due nodi del grafo.

Definito che due archi si dicono **adiacenti** se esiste un nodo che è estremo per entrambi, in un **grafo non orientato** è detto **cammino** un insieme di archi (a_1, a_2, \dots, a_n) tale per cui a_i e a_{i+1} sono adiacenti, per ogni i .

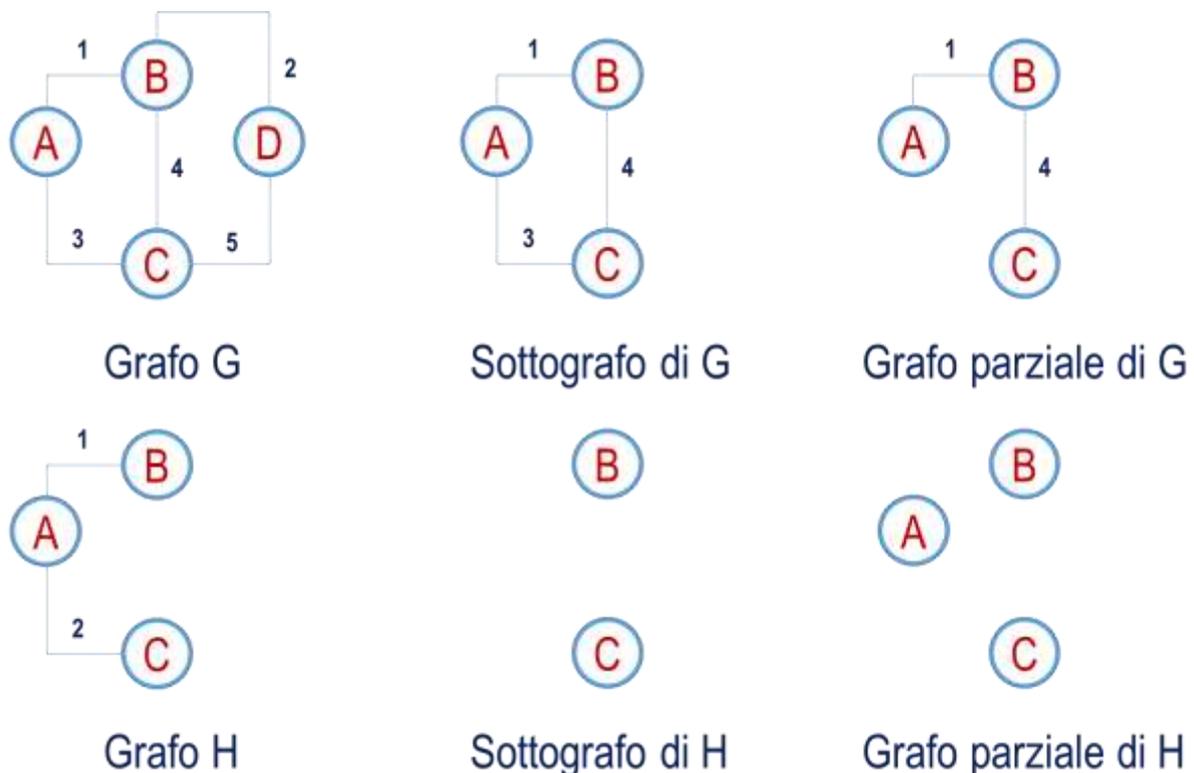


Figura 7 - Esempi di sottografo e grafo parziale

⁹ Un arco si dice incidente su un nodo se il nodo ne è un estremo.

In un **grafo orientato** si hanno due tipi di cammino:

- **cammino non orientato** o **catena**: non pone vincoli all'orientamento degli archi;
- **cammino**: richiede che la sequenza di archi sia tale che la testa di un arco coincida con la coda del successivo.

Se un cammino non passa mai due volte per lo stesso nodo è detto **cammino elementare**, se invece un cammino non passa mai due volte per lo stesso arco è detto **cammino semplice**.

Con questi concetti base si dice che un grafo tale per cui per ogni coppia di nodi esiste un cammino che li unisce è detto **grafo connesso**; nel caso di grafi orientati, se esiste un cammino per ogni coppia di nodi si dice che il grafo è **fortemente connesso**, mentre se esiste solo un cammino non orientato, o catena, fra ogni coppia di nodi il grafo si dice **debolmente connesso** (vedi Figura 8).

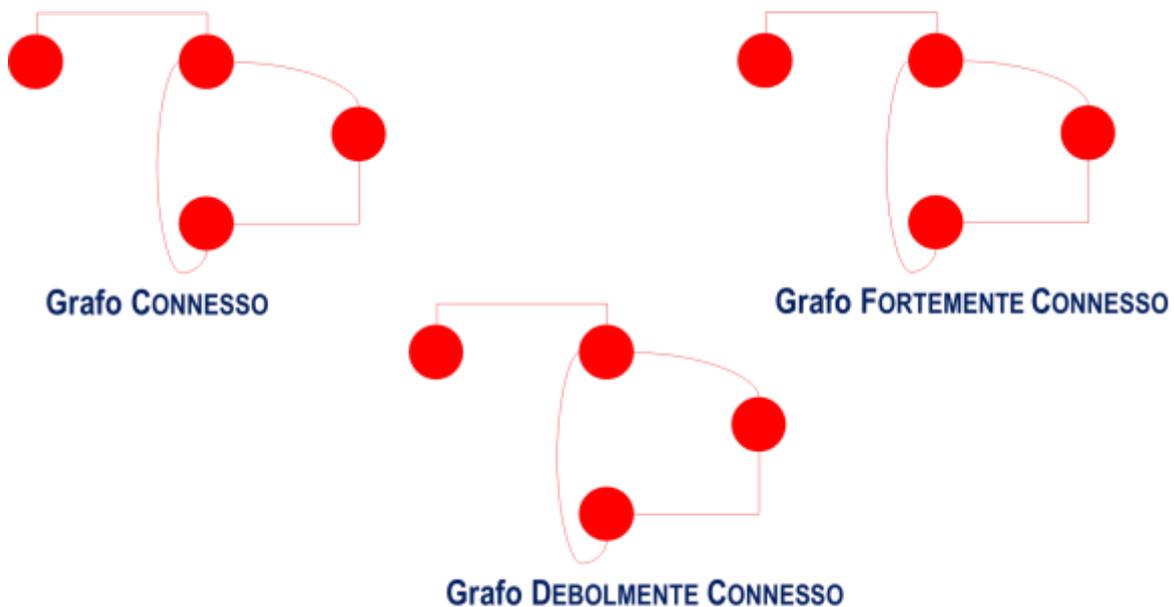


Figura 8 - Grafi e connessione

La caratterizzazione di un grafo a partire dai nodi e dagli archi che formano cammini tra due nodi, porta a definire **diametro**, **centro** e **raggio** di un grafo.

Dato un grafo $G(N,A)$, non orientato, e sia $d(i,j)$ il minimo numero di archi necessario per costruire un cammino tra i nodi i e j , si definisce **diametro** del grafo G , il valore massimo di tutte le distanze $d(i,j)$:

$$D = \max (i,j) d(i,j)$$

Si dice **centro** del grafo il nodo, o i nodi, per cui è minima la quantità $R(i)$, che è la distanza massima di ogni nodo j del grafo, fissato il nodo i :

$$R(i) = \max d(i,j)$$

Preso in considerazione k , il nodo centro o uno dei nodi centro del grafo, si definisce **$R(k)$** come **raggio** del grafo.

Una ulteriore definizione, nell'insieme delle strutture notevoli che coinvolgono i cammini in un grafo, è quella di **ciclo**, un cammino tale che il nodo di partenza e il nodo di arrivo coincidono; così come per i cammini, in generale, un ciclo che non passa mai due volte per lo stesso nodo è detto **ciclo elementare**.

Analizzando le caratteristiche dei cicli in un grafo o un multigrafo, un **ciclo** che passa esattamente una volta per ogni arco è detto **ciclo euleriano**; un grafo che ammette un ciclo euleriano è un **grafo euleriano**, grafo che soddisfa il citato Teorema di Eulero.

In maniera duale¹⁰, un ciclo che passa esattamente una volta sola per ogni nodo è detto ciclo hamiltoniano e, analogamente a ciò che è stato affermato in precedenza, un grafo che ammette un **ciclo hamiltoniano** è detto **grafo hamiltoniano** (Vedi Figura 9).

I cicli hamiltoniano devono il loro nome a William Rowland Hamilton¹¹ che inventò un gioco da tavolo, noto come **icosian game** o **puzzle di Hamilton** (vedi Figura 10), che consisteva nel trovare un ciclo sul grafo composto dai 20 vertici e 30 spigoli del bordo di un **dodecaedro**¹².



Figura 9- Icosian game o Puzzle di Hamilton, gioco inventato nel 1857

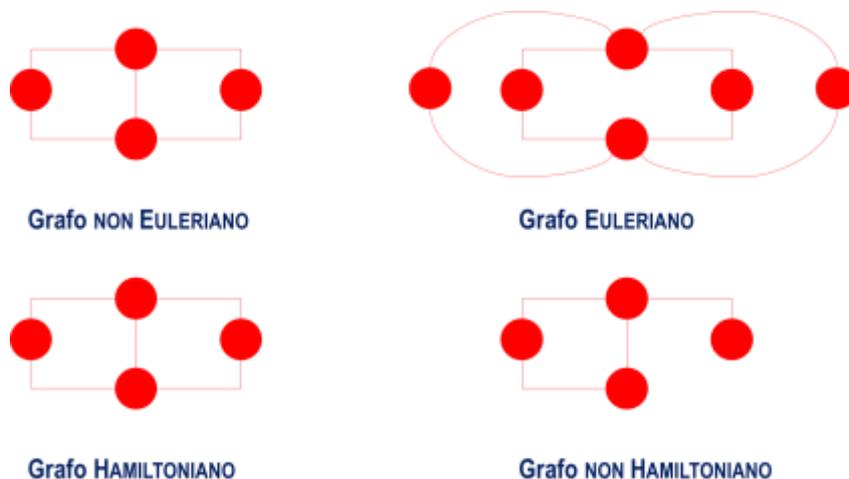


Figura 10 -Esempi di grafi euleriani e hamiltoniani

Considerando la distribuzione spaziale di un grafo e analizzando da un punto di vista degli insiemi i nodi si possono enunciare due ulteriori definizioni:

¹⁰ In matematica il concetto di dualità indica la proprietà di due proposizioni ottenibili l'una dall'altra scambiando fra loro determinati enti o operazioni.

¹¹ **William Rowland Hamilton** (1805-1865), matematico e fisico irlandese che si pose il problema di affrontare una variazione del teorema di Eulero e cioè: "quali sono le condizioni affinché sia possibile determinare che un grafo ammette un ciclo che passa, una ed una sola volta, per ogni nodo?".

¹² Il **dodecaedro**, in geometria solida, è un poliedro con 12 facce. Il dodecaedro regolare, le cui facce sono pentagoni regolari, è uno dei 5 solidi platonici (tetraedro, cubo o esaedro, ottaedro, dodecaedro e icosaedro).

1. **grafo planare**: un grafo tale che sia possibile disegnarlo su di un piano senza che gli archi si intersechino (vedi Figura 11);
2. **grafo bipartito**: grafo, orientato o non orientato, tale che l'insieme **N** dei suoi nodi può essere scomposto nei due sottoinsiemi **S** e **T** con $(S \cup T) = N$, per cui non esistono archi con entrambi i nodi terminali in **S** o in **T**; se poi per ogni coppia di nodi (i,j) con **i** appartenente a **S** ($i \in S$) e **j** appartenente a **T** ($j \in T$) esiste un arco che li congiunge il grafo è detto **bipartito completo** (vedi Figura 12);

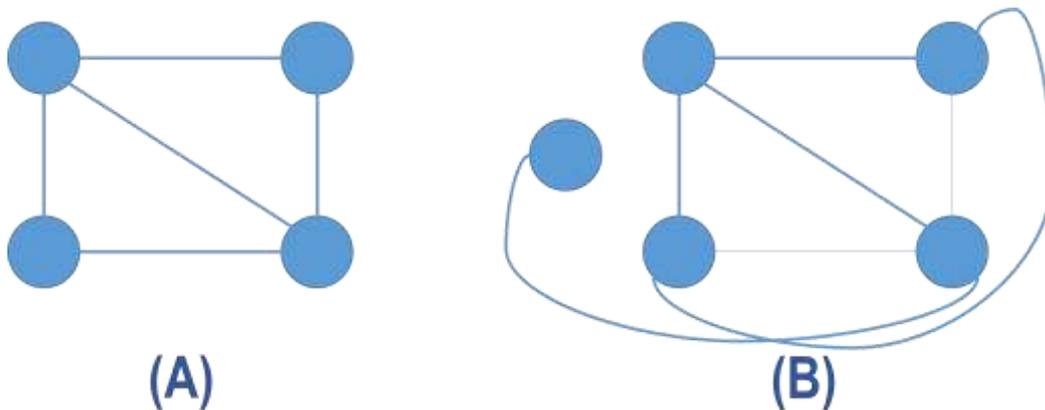


Figura 11 - Esempio di grafo planare (A) e grafo non planare (B)

Nell'analisi delle diverse configurazioni e tipologie di grafo si definisce **foresta** un grafo non orientato e privo di cicli; se il grafo privo di cicli è connesso allora si parla di **albero**.

Il concetto di **ALBERO** è molto importante, in particolare perché è un ottima rappresentazione di importanti strutture applicative come la struttura delle directory in un hard disk.

Per un **albero** è possibile dare sei definizioni, tra loro equivalenti:

1. grafo connesso e senza cicli;
2. grafo connesso con n nodi e $n-1$ archi;
3. grafo senza cicli con n nodi e $n-1$ archi;
4. grafo senza cicli tale che comunque si aggiunga un arco si crea uno ed un solo ciclo;
5. grafo connesso tale che comunque si elimini un arco si ottiene un grafo non connesso;
6. grafo tale che per ogni coppia di nodi esiste esattamente un cammino che li collega.

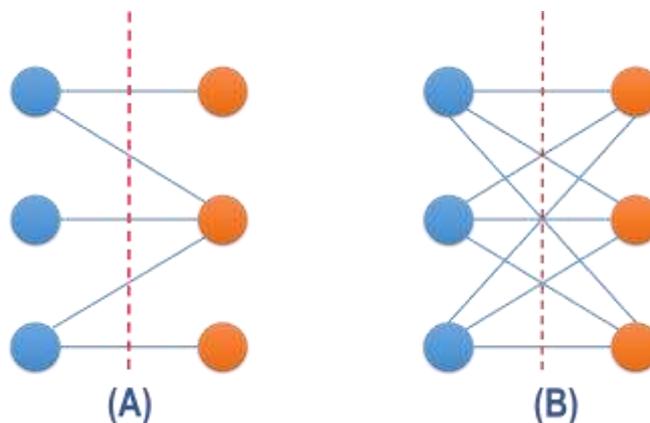


Figura 12 - - Esempio di grafo bipartito (A) e grafo bipartito completo (B)

In un albero i nodi di **grado 1** sono detti **foglie** e scelto un nodo e detto tale nodo **radice**, ottenendo così un **albero radicato** (vedi Figura 13), si introducono alcuni concetti ulteriori:

- **livello o profondità** di un nodo: numero di archi che è necessario percorrere per giungere dal nodo alla radice;
- **altezza dell'albero**: il livello o profondità massima di un nodo nell'albero;
- **padre e figlio**: due nodi adiacenti come i nodi 1 (padre) e 4 (figlio) in Figura 13;
- **nodo X antenato di Y**: se X è nel cammino tra R e Y come il nodo 1, antenato del nodo 4 in Figura 13;

Casi notevoli per un albero sono quelli in cui tutti i nodi hanno un grado minore o uguale a 3 e in questo caso l'albero è detto **albero binario** e, se tutte le foglie sono al medesimo livello, cioè se ogni foglia dell'albero ha la stessa distanza dalla radice, l'albero si dice **bilanciato**, dove se k sono i livelli, i nodi sono $(2^{k+1}-1)$.

Infine se un albero bilanciato ha ogni nodo di grado 2 è detto **albero completamente bilanciato** o **albero pieno**; se tutte le foglie, inoltre, sono allo stesso livello si dice essere un albero bilanciato completo; in questo ultimo tipo di albero, se k sono i livelli, i nodi sono $(2^{k+1}-1)$.

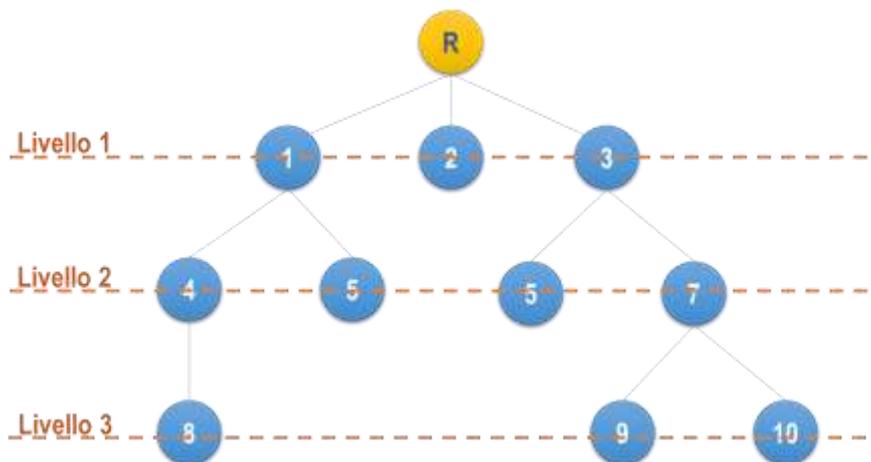


Figura 13 - Esempio di albero radicato (nodo radice R, livello 0) di profondità 3

Metodi di memorizzazione di un grafo

L'utilizzo dei grafi come struttura relazionale è strettamente legato alla possibilità di memorizzarne struttura e relazioni in maniera semplice ed efficace.

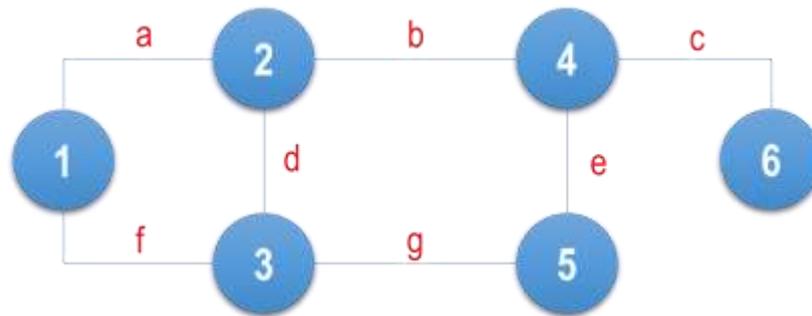
Un grafo $G(N,A)$ può essere memorizzato su di un calcolatore in molteplici modi e la scelta della modalità è fortemente dipendente dalle caratteristiche che si vogliono evidenziare, dal tipo di utilizzo che si intende fare, dalle dimensioni del grafo da memorizzare, in particolare il numero di nodi e di archi e dalla sua densità, cioè il numero di archi rispetto al numero di nodi.

Le modalità di memorizzazione che è possibile utilizzare (vedi Figura 14) sono:

- matrice di connessione;**
- matrice di incidenza;**
- lista di adiacenza.**

Di queste, le più utilizzate sono matrice di connessione e lista di adiacenza; la matrice di connessione è preferibile nel caso di grafi molto densi, cioè dove sono molti i nodi connessi tra loro, rispetto alla lista di adiacenza; per grafi molto densi la matrice di connessione presenta una maggiore efficienza ed immediatezza dei calcoli.

La lista di adiacenza è preferibile nel caso di grafi sparsi, cioè dove il numero di archi è notevolmente inferiore rispetto al numero di nodi.



6x7	a	b	c	d	e	f	g
1	1	0	0	0	0	1	0
2	1	1	0	1	0	0	0
3	0	0	0	1	0	1	1
4	0	1	1	0	1	0	0
5	0	0	0	0	1	0	1
6	0	0	1	0	0	0	0

Matrice di INCIDENZA

Nodo	
1	2,3
2	1,3,4
3	1,2,5
4	2,5,6
5	3
6	4

Lista di INCIDENZA

6x6	1	2	3	4	5	6
1	0	1	1	0	0	0
2	1	0	1	1	0	0
3	1	1	0	0	1	0
4	0	1	0	0	1	1
5	0	0	1	1	0	0
6	0	0	0	1	0	0

Matrice di CONNESSIONE

Figura 14 - Esempi di memorizzazione di un grafo NON orientato

La matrice di incidenza, seppur meno efficace degli altri due metodi dal punto di vista computazionale, è preferibile in alcuni problemi di ricerca operativa¹³, in cui corrisponde molto bene ad alcuni modelli di ottimizzazione, per cui ne facilita la rappresentazione.

La matrice di connessione come metodo di memorizzazione, si basa sull'utilizzo di una **matrice quadrata $n \times n$** , con n numero dei nodi del grafo, in cui il generico elemento **(i,j)** assumerà il valore 1 se esiste un arco che connette il nodo i con il nodo j e 0 negli altri casi; questo permette di memorizzare sia grafi orientati che grafi non orientati, notando che nel caso di grafi non orientati la matrice sarà simmetrica, cioè il valore contenuto nell'elemento **(i,j)** è uguale al valore contenuto nell'elemento **(j,i)**.

La memorizzazione di un grafo con una **matrice di incidenza** si basa sull'utilizzo di una matrice **$n \times m$** , con n numero dei nodi ed m numero degli archi del grafo; in questa matrice il generico elemento **(i,j)**, con nodi ed archi numerati in modo arbitrario, assumerà il valore 1 se l'arco j incide sul nodo i e 0 se non incide sul nodo i .

Se il grafo memorizzato utilizzando la matrice di incidenza è un grafo orientato, il generico elemento **(i,j)** della matrice conterrà il valore **1** se l'arco j esce dal nodo i (**i coda dell'arco**), **-1** se l'arco j entra nel nodo i (**i testa dell'arco**) e **0** se l'arco j **non incide il nodo** i , cioè il nodo i non è un estremo dell'arco j .

¹³ La "**ricerca operativa**" è una scienza che fornisce strumenti matematici a supporto delle decisioni relative alla gestione di attività che prevedano l'utilizzo di risorse limitate per poter massimizzare o minimizzare funzioni obiettivo. La ricerca operativa è anche definita teoria delle decisioni o scienza della gestione e in sintesi si occupa di formalizzare problemi, in particolare approssimando i problemi complessi, con l'utilizzo di modelli matematici al fine di giungere a una soluzione "**ottima**" o "**approssimata**" definita anche "**subottima**".

Infine la lista di adiacenza è basata sulla memorizzazione di una lista che contiene, per ogni nodo del grafo, l'elenco dei relativi nodi adiacenti; tale lista richiede un massimo di $(n + m)$ elementi (n numero di nodi ed m numero degli archi del grafo). Un esempio dei metodi ora descritti è visibile in Figura 8.

Grafi e rappresentazione dei problemi

La teoria dei grafi è un passo fondamentale per correggere quel modo di agire che Arthur Schopenhauer raccoglieva nell'aforisma «L'errore nasce sempre dalla tendenza dell'uomo a dedurre la causa della conseguenza». I grafi devono molto in termini di importanza e di applicazione alle possibilità che offrono in relazione alla rappresentazione in modo semplice ed efficace dei problemi¹⁴, in termini di dati e relazioni.

I grafi permettono di modellare diverse tipologie di problemi come, ad esempio, per i grafi non orientati, problemi connessi alle reti come possono essere problemi di flusso, traffico, trasmissione, raggiungibilità o distanza minima in reti stradali, telefoniche, di computer o ferroviarie; analogamente, prendendo in considerazione grafi orientati, si possono modellare problemi legati a strategie progettuali con diversi obiettivi collegati da procedure oppure si possono modellare sistemi fatti di **stati**, i **nodi**, e **transizioni**, gli **archi**; ad esempio si possono analizzare problemi legati al gioco degli scacchi, dove le mosse sono le transizioni che portano da una configurazione ad un'altra della disposizione dei pezzi in scacchiera.

Di questo tipo è il **problema** dello **scambio dei cavalli**¹⁵ che si può enunciare nel seguente modo: “data una scacchiera 3x3 e numerate le caselle da 1 a 9, dall'alto in basso e da sinistra verso destra, e posti i cavalli bianchi degli scacchi nelle caselle 1 e 3 e i cavalli neri nelle caselle 6 e 9: è possibile spostare i cavalli bianchi nelle caselle 6 e 9 i cavalli neri nelle caselle 1 e 3, spostando un cavallo alla volta secondo la modalità degli scacchi e senza mai avere due cavalli nella medesima casella?”.

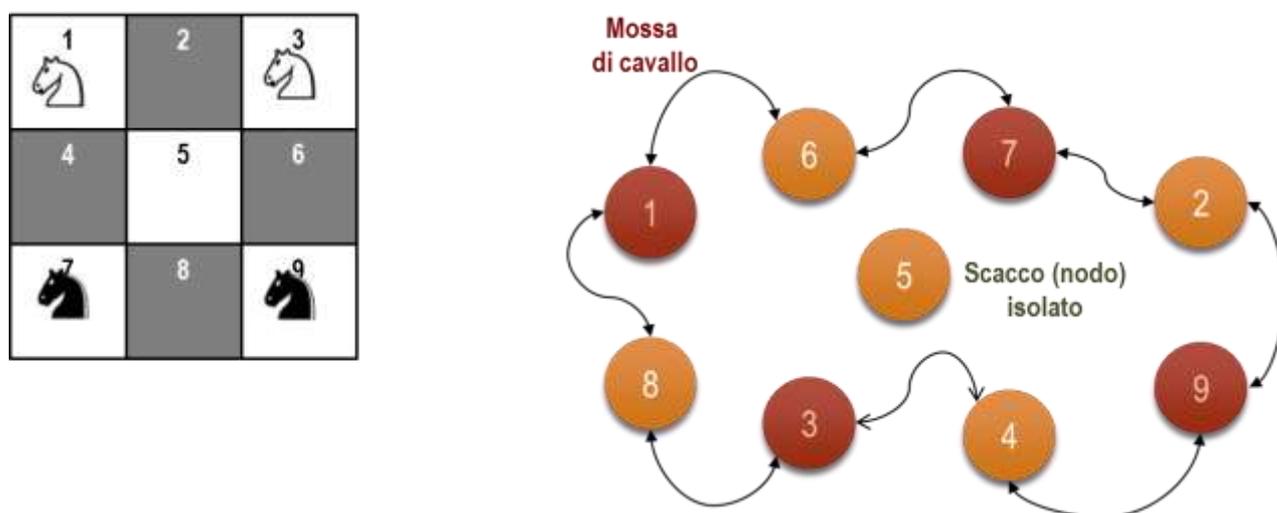


Figura 15 - Grafo schema del problema dello “scambio dei cavalli” e immagine della relativa scacchiera di riferimento

¹⁴ Il termine “**problema**” deriva dal greco **πρόβλημα** (**problēma**) che significa “*sporgenza, promontorio, impedimento, ostacolo*”, ma nell’accezione **matematica** ci si occupa solo di situazioni risolvibili, si può dire che c’è problema solo se c’è soluzione. In questo contesto l’uso dei grafi permette di schematizzare come nodi tutte le possibili situazioni generabili dai dati di partenza e come archi tutte le regole che mettono in relazione una situazione con una derivante da essa. Con queste premesse risolvere un problema significa identificare un cammino che congiunga il nodo della condizione iniziale al nodo della configurazione soluzione del problema.

¹⁵ Il problema è stato proposto da **Henry Ernest Dudeney** (1857-1930) un matematico britannico, conosciuto come un maestro della matematica divertente e autore di molti enigmi che permettono di approfondire i legami tra la matematica, la logica e il gioco. I suoi lavori hanno dato un contributo fondamentale alla divulgazione della **matematica creativa**.

Rappresentare gli stati e le regole del problema dello scambio dei cavalli per evincere lo schema di risoluzione del problema utilizzando un grafo significa associare ad ogni casella un nodo, e definire che due nodi, e quindi gli scacchi che rappresentano, sono collegati da un arco se solo se è possibile passare da uno all'altro con una mossa di cavallo (vedi Figura 15); da questo schema si osserva, ad esempio, che lo scacco 5, a partire dalla configurazione iniziale del problema, non è raggiungibile e che il problema ha soluzione poiché è possibile muovere i cavalli, in maniera circolare, secondo lo schema del grafo poiché è possibile non sovrapporli in nessuna delle configurazioni possibili.

Un altro problema, che può essere definito "classico", e che nel quale ci si può avvalere della teoria dei grafi per un'analisi dello schema di soluzione è un enigma noto a tutti i bambini fin dai primi anni di scuola noto anche come il **problema di "salvare capra e cavolo"**, che è modo di dire popolare ma che trae origine da un rompicapo vecchio di secoli e la cui nunciazione in modo rigoroso e relativa soluzione si deve a Tartaglia¹⁶ (*libro 16, N. 141*) che scrive, a proposito di un problema che enuncia, "e da questo è nasciuto un certo proverbio fra gli huomini, dicendo in qualche proposito, egli ha salvato la capra e i verzi".

Il problema si può enunciare nel modo seguente:

"Un uomo vuole traghettare da una sponda all'altra di un fiume un lupo, una capra ed una verva utilizzando una barca capace solo di ospitare l'uomo e la verza o una delle due bestie; è possibile traghettare verza e bestie in modo tale che su una sponda non rimangano mai una delle coppie capra-lupo o capra cavolo?"

Ovviamente non è possibile lasciare soli il lupo e la capra oppure la capra e il cavolo altrimenti il lupo mangia la capra o la capra mangia il cavolo; in pratica si tratta di "interrompere" la catena alimentare che da un punto di vista della programmazione si può rappresentare con un semplice vettore di 3 elementi in cui memorizzare la presenza di un elemento sulla sponda, lo schema successivo mostra la situazione iniziale sulla sponda di partenza dove ci sono l'uomo che ha verza (V), capra (C) e lupo (L) con sé:

V	C	L
1	1	1

Osservando che ogni elemento della catena alimentare così rappresentata è "preda dell'elemento alla sua sinistra saranno accettabili solo le configurazione in cui non ci sono due 1 in elementi contigui.

Ricorrendo alla teoria dei grafi è possibile schematizzare le mosse necessarie per la soluzione dell'enigma (vedi Figura 16) e cioè come è possibile traghettare le tre entità senza perderne alcuna; usando un metodo descrittivo si può descrivere il processo risolutivo nel seguente modo algoritmico:

1. sulla sponda A ci sono, con il traghettatore, la verza la capra e il lupo;
2. con le regole del problema il traghettatore ha una sola possibilità e cioè trasportare sulla sponda B la capra, lasciando sulla sponda A la verza e il lupo, che è una combinazione ammessa e in questo caso anche l'unica;
3. lasciata la capra sulla sponda B da sola, il traghettatore torna sulla sponda A e a questo punto può scegliere se caricare la verza o il lupo:
 - a. se sceglie il lupo, sulla sponda A rimane il cavolo; quando giunge sulla sponda B non potendo lasciare lupo e capra insieme lasciato il lupo sulla sponda B riporta

¹⁶ Niccolò Tartaglia (1499 ca - 1577), soprannome del matematico italiano Niccolò Fontana noto per il triangolo numerico, detto *triangolo di Tartaglia*, metodo per lo sviluppo dei coefficienti di un binomio elevato a potenza n, $(a+b)^n$, e per la formula di *Cardano-Tartaglia* per la risoluzione di equazioni cubiche.

- la capra sulla sponda A, dove si comporrà la combinazione di verza, capra e traghettatore;
- b. se sceglie la verza, sulla sponda A rimane il lupo; quando giunge sulla sponda B non potendo lasciare cavolo e capra insieme lasciato il cavolo sulla sponda B riporta la capra sulla sponda A, dove si comporrà la combinazione di capra, lupo e traghettatore;
4. qualsiasi sia stata la scelta al passo precedente, sia la combinazione verza, capra e traghettatore, che la combinazione capra, lupo e traghettatore, obbliga il traghettatore a portare il cavolo sulla sponda B, sia perché è l'unico elemento che compone sulla sponda B l'unica combinazione ammessa di due elementi, lupo e cavolo, sia perché altre scelte plausibili farebbero ripercorrere a ritroso il cammino che il grafo schematizza;
 5. il traghettatore torna sulla sponda A, carica la capra e ricomponne sulla sponda B la combinazione traghettatore, verza, capra, e lupo.

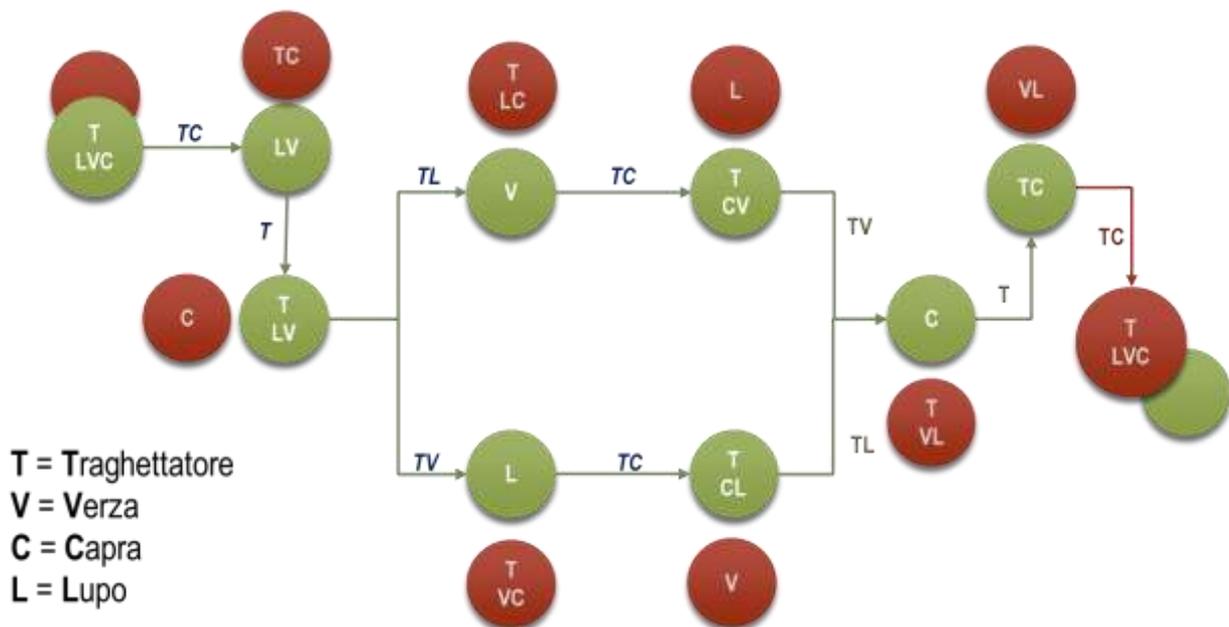


Figura 16 - Grafo dello schema di risoluzione del problema "salvare capra e cavolo"; in questo schema i nodi contengono le configurazioni presenti sulla riva di partenza (Riva A) e le relazioni (gli archi) rappresentano gli stati che si ottengono dopo un passaggio.

In questa esemplificazione della risoluzione di un problema con l'ausilio della teoria dei grafi si può notare anche come alcuni enigmi, proprio come quello "salvare capra e cavoli", siano risolti direttamente in funzione delle regole che vengono enunciate; chiunque affronti l'enigma "salvare capra e cavoli" deve solo applicare le regole senza compiere nessuna scelta, e anche dove questa scelta fosse possibile, come nel passo 3 del processo risolutivo, in realtà il cammino logico ha un numero di passi sempre uguale (vedi gli archi del grafo in Figura 16).

Le applicazioni affrontabili con la teoria dei grafi sono schematizzabili attraverso l'analisi dei problemi classici legati alla teoria dei grafi stessa e si possono coniugare con diverse applicazioni come ad esempio:

- problemi di trasporto e viabilità, definizione di reti stradali e percorsi ottimizzati;
- problemi di definizione dello schema di circuiti elettrici, elettronici, schede stampate;
- problemi di ottimizzazione di reti di calcolatori per la distribuzione e l'immagazzinamento di informazioni;
- problemi di assegnazioni di aree di controllo, distribuzione di dati nella memoria fisica di un computer, aree di competenza per un responsabile di vendita.

In particolare è interessante elencare i quattro **problemi classici** le cui soluzioni implicano e sviluppano la teoria dei grafi sono:

1. **Problema dei ponti di Königsberg**
2. **Problema dei quattro colori**
3. **Problema del commesso viaggiatore**
4. **Problema delle tre case e delle tre forniture**

Il **problema dei ponti di Königsberg** è stato analizzato ed è specificamente legato a problemi di ottimizzazione come ad esempio il **problema del postino cinese**¹⁷ (**Chinese Postman's Problem**) e in questo contesto si analizzano i problemi di **Vehicle Routing**.

I problemi di **vehicle routing** possono essere descritti in modo formale come il problema di determinare, dato un numero v di veicoli e un insieme C predeterminato di clienti da servire, un insieme di n percorsi a costo minimo, dove per costo si possono considerare distanze, tempi o risorse, in modo tale da servire tutti i clienti; questa tipologia di problemi prevede l'utilizzo di un grafo $G(N,V)$, con archi sia orientati che non orientati e tale per cui ad ogni **arco** (i,j) che raccorda il nodo i con il nodo j , è associato un valore non negativo, il **costo**, indicato con c_{ij} e usando un formalismo matematico questo vincolo si scrive:

$$c_{ij} > 0 \text{ per ogni coppia di nodi } i, j$$

Nei problemi di **vehicle routing** i clienti sono rappresentati sia da nodi che da archi; se tutti i clienti sono rappresentati da nodi si parla di problemi di **NODE ROUTING**, mentre se tutti i clienti sono rappresentati da archi si parla di problemi di **ARC ROUTING**; i problemi di tipo **vehicle routing** presentano un alto grado di complessità dovuto al numero e alla variabilità dei vincoli che da un punto di vista operativo presentano.

Il **problema dei quattro colori**¹⁸, contesto in cui si utilizza la teoria dei grafi per una schematizzazione e rappresentazione semantica del problema, enuncia:

“Data una superficie piana divisa in regioni connesse, come ad esempio una carta geografica, sono sufficienti quattro colori per colorare ogni regione facendo in modo che regioni adiacenti non abbiano lo stesso colore, dove per regioni adiacenti si intendano aree che abbiano un segmento di confine in comune.”

Il **“problema dei quattro colori”** è probabilmente uno tra i più noti, nella storia della teoria dei grafi, e nato da una congettura¹⁹ di Francis Guthrie, che si rese conto che bastavano quattro colori per colorare una mappa della Gran Bretagna in maniera tale che due stati confinanti non avessero mai lo stesso colore; un esempio (vedi Figura 17) è la colorazione della mappa degli Stati Uniti.

¹⁷ Il problema del postino cinese ((Chinese Postman's Problem) è un problema di teoria dei grafi la cui formulazione si deve al matematico cinese Mei-Ku Kwan nel 1962, che in un articolo, *Graphic Programming Using Odd or Even Points, Chinese Math.*, 1, pp. 273-277, 1962, si è occupato dell'ottimizzazione del percorso di un postino. Il nome **“problema del postino cinese”** si deve a Alan J. Goldman quando lavorava al **NBS (National Bureau of Standards)**, che ora è il **NIST (National Institute of Standards and Technology)** agenzia degli Stati Uniti che si occupa della gestione delle tecnologie.

¹⁸ Il **“problema dei quattro colori”** è stato enunciato per la prima volta nel 1852 da Francis Guthrie, allievo del matematico e logico inglese Augustus De Morgan (1806-1871); il primo studioso a pubblicare, relativamente a questo problema, è stato il matematico inglese Arthur Cayley (1821-1895), nell'articolo *On the colourings of maps* pubblicato in *“Proceedings of the Geological Society”*, 1, 1879, pp. 259-261.

¹⁹ Per **“congettura”**, dal latino *coniectūram* (verbo conicere, che significa **“concludere”**, **“interpretare”**), si intende una affermazione basata sull'intuito, che si crede vera ma è non ancora dimostrata.

Anche questo problema può essere schematizzato in un grafo dove i punti sono le aree geografiche e gli archi collegano due punti solo se rappresentano aree geografiche con un segmento di confine in comune (vedi Figura 18).



Figura 17 - Mappa degli Stati Uniti colorata secondo i principi del problema dei quattro colori

La dimostrazione del **problema o teorema dei quattro colori** è notevolmente complessa, la storia di questa dimostrazione è articolata e lunga nel tempo (vedi Figura 19).

Per averne una completa si deve giungere al 1976, ad opera di Kenneth Appel e Wolfgang Haken, matematici dell'Università dell'Illinois, che poterono contare sull'ausilio fondamentale di calcolatori veloci e potenti.

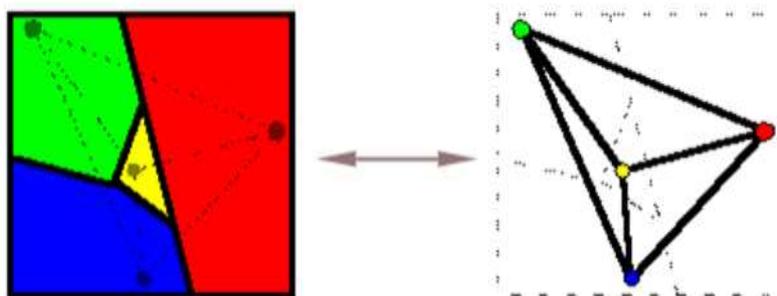


Figura 18- Rappresentazione che associa ad ogni regione della mappa un nodo del grafo

Questo problema da una parte riveste un'importanza metodologica poiché evidenzia come si possa affrontare un problema con l'ausilio fondamentale degli strumenti della logica, e dall'altra riveste un'importanza operativa dimostrando come sia possibile trasformare il problema in un sistema che permetta un approccio topologico alla soluzione, così come aveva fatto Kirchhoff in relazione a problemi di elettromagnetismo.



Figura 19 - Linea temporale dello sviluppo della dimostrazione del teorema dei quattro colori

Il problema del commesso viaggiatore (**TSP - Travelling Salesman Problem**), di tipo “*hamiltoniano*”, è un caso particolare dei problemi di tipo “*node routing*” in cui si ha un solo veicolo su un grafo fortemente connesso se si opera su grafi orientati, o connesso se si opera su grafi non orientati.

Se il grafo è orientato si parla del “**problema del commesso viaggiatore asimmetrico**”, mentre nel caso di grafo non orientato si parla del “**problema del commesso viaggiatore simmetrico**”.

Il problema del commesso viaggiatore è prototipo per problemi che si occupano, ad esempio, di determinare:

- flussi delle merci tra i magazzini di fornitori e di distributori;
- percorso più breve per connettere due o più città;
- percorso minimo di un trapano per creare i fori nella piastra di un circuito stampato.

La formulazione classica del **problema del commesso viaggiatore** è:

“Data una rete di città connesse ognuna a ciascuna delle altre con strade dirette, trovare il percorso di minore lunghezza che un commesso viaggiatore deve seguire per visitare tutte le città.”

Lo stesso problema enunciato in termini di grafi è definibile come “dato un grafo completo pesato²⁰, trovare il ciclo hamiltoniano con peso minore”.

Il problema del commesso viaggiatore presenta diversi algoritmi risolutivi che sono tecniche di tipo euristico²¹, come ad esempio quelli detti “**algoritmi dell’albero**” (*Tree Algorithms*), per la visita delle strutture ad albero, per i problemi di tipo simmetrico e l’algoritmo “**vicino più prossimo**” (*Nearest Neighbour Algorithm – NNA*) per i problemi di tipo asimmetrico.

In particolare il “Nearest Neighbour Algorithm” (vedi Figura) è stato uno dei primi utilizzati per la risoluzione dei problemi di tipo **TSP** (**T**ravelling **S**alesman **P**roblem).

Questo algoritmo seppur facile da implementare e rapido da eseguire, tempi di esecuzione minimi, per la sua natura di algoritmo “**greedy**”²² può non determinare i percorsi più brevi, percorsi per altro identificabili attraverso processi di intuizione da parte dell’uomo.

²⁰ Un **grafo completo pesato** è un grafo tale per cui ogni nodo è collegato a tutti i rimanenti nodi con un arco e ogni arco è associato a un valore, ad esempio per le strade che collegano due città il peso possono essere i chilometri che dividono le due città stesse.

²¹ Per **procedimento euristico** (*euristica* deriva termine greco **εὐρίσκω, heurisko** che significa scopro o trovo) si intende, in contrapposizione con il metodo algoritmico, un approccio alla soluzione dei problemi che non segue un cammino preciso e non ha obiettivi precisi prestabiliti, ma si affida a intuizioni e all’analisi delle circostanze per generare nuova conoscenza.

²² Un algoritmo “**greedy**” (avido, goloso) è un algoritmo che opera per ottenere una soluzione che risulti ottima globalmente operando scelta della soluzione “**più golosa**”, “**guidata dall’avidità**”, come ad esempio la distanza più breve o il costo minore, ad ogni passo locale.

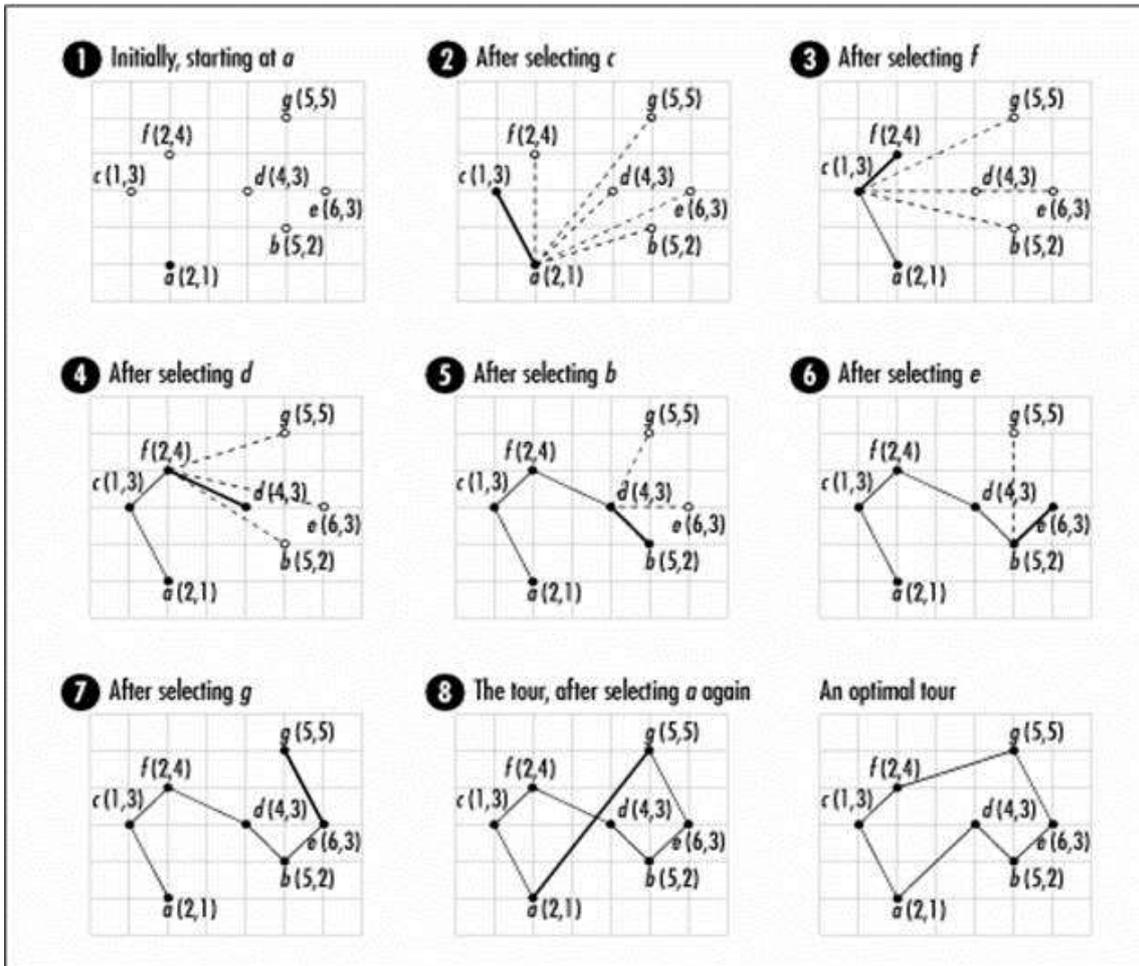


Figura 20 – Schema grafico dei passi per la soluzione del “problema del commesso viaggiatore” utilizzando che usando l’algoritmo euristico “Nearest Neighbour Algorithm”

Una descrizione formale dell’algoritmo NNA (vedi Figura 20) può essere fatta attraverso la descrizione dei semplici passi previsti:

1. **Definire** in maniera arbitraria un nodo N come nodo corrente (città in cui si è)
2. **Identificare** l’arco più breve che parte da N e lo collega ad M, nodo non ancora visitato
3. **Definire** M come nuovo nodo corrente
4. **Marcare** M come visitato
5. **Se** tutti i nodi dell’insieme considerato sono marcati come visitati l’algoritmo ha **termine**
6. *Tornare al punto 2*

La sequenza dei nodi così identificata è l’output dell’algoritmo e rappresenta il percorso soluzione del problema TSP.

Questo algoritmo, nel peggiore dei casi, potrebbe produrre un percorso notevolmente più lungo di un percorso ottimo; inoltre, per ogni numero di città e c’è un combinazione di assegnazioni dei valore delle distanze tra ogni coppia di città per cui l’euristica del NNA produce l’**unico peggior giro possibile**, così come definito da Gregory Gutin, Anders Yeo e Alexey Zverovic²³.

²³ Gutin G., Yeo, Zverovic A., Traveling salesman should not be greedy: domination analysis of greedy-type heuristics for the TSP in Discrete Applied Mathematics Volume 117, Issues 1–3, 15 March 2002, Pages 81–86, Elsevier Editorial System

Il **problema delle tre case e delle tre forniture** è un ulteriore esempio di classe di problemi in cui trova applicazione la teoria dei grafi, che interessa particolarmente i grafi planari; spesso questo problema è interpretato come gioco dai bambini nel problema “*le tre case e i tre pozzi*” che si può enunciare:

“*Si possono collegare tre case a tre fornitori senza che le strade che le connettono si incrocino? o, in modo equivalente, si possono collegare tre case a tre forniture senza che le tubature che le uniscono ai pozzi si incrocino? e, eventualmente questo non fosse possibile, qual è il numero di incroci necessari?*”

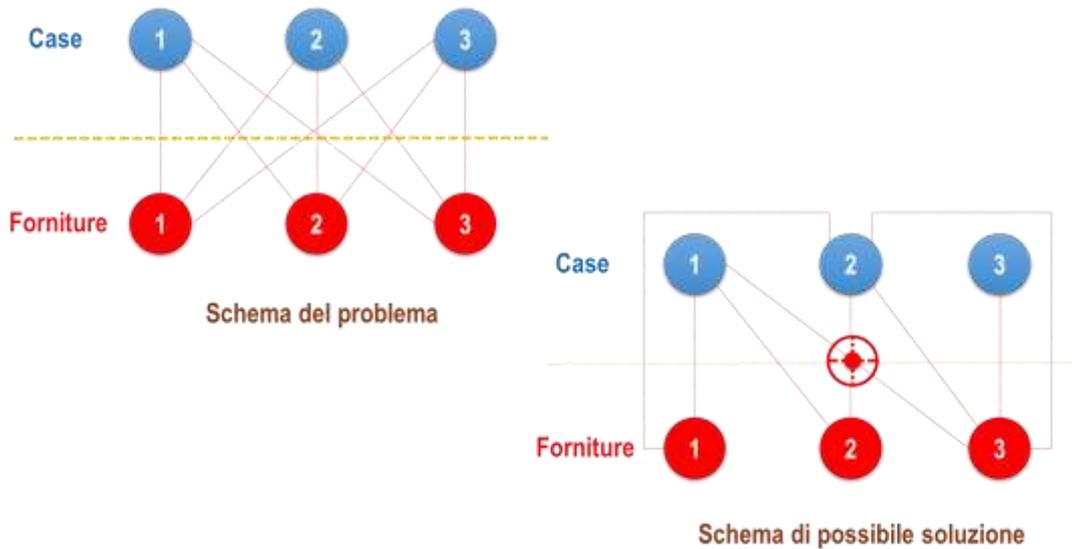


Figura 21 - Rappresentazione del problema “tre case a tre fornitori”

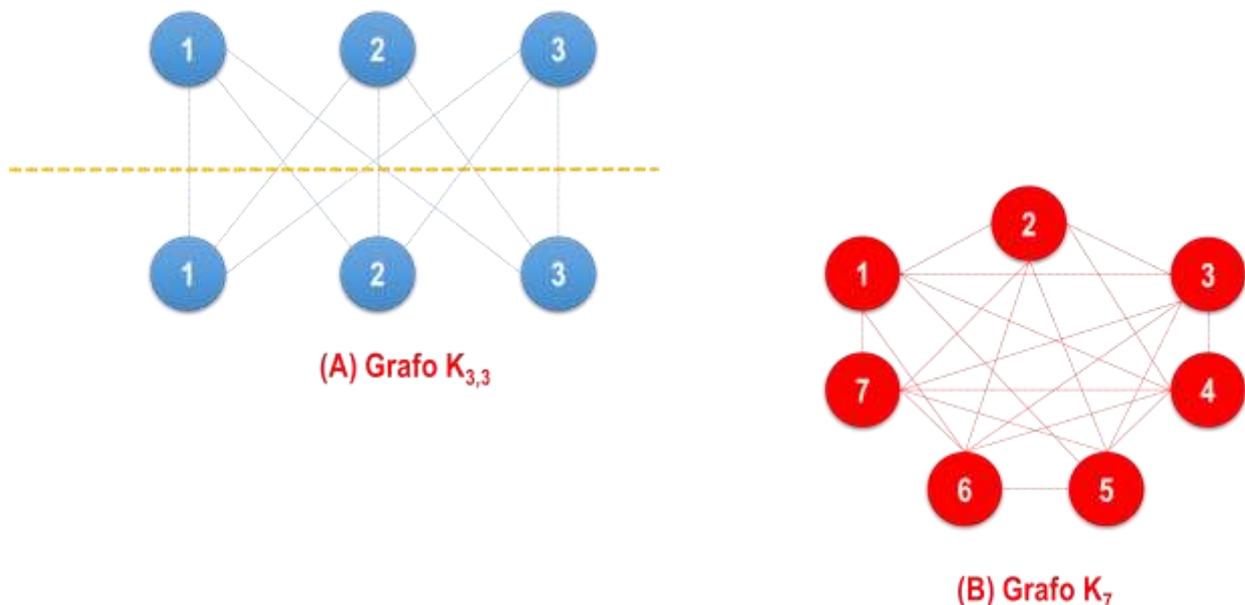


Figura 22- (A): esempio di grafo $K_{3,3}$, grafo bipartito completo 3+3 nodi; (B): esempio di grafo K_7 , grafo completo con 7 nodi.

A partire da questo quesito e lavorando sulle proprietà e le caratteristiche dei grafi planari si dimostra che questo problema non è risolvibile ed è **1** il numero minimo di incroci necessari

nel contesto del problema enunciato (vedi Figura 22), soluzione individuata con l'applicazione del **teorema di Kuratowski**²⁴ che postula:

“Un grafo è planare se e solo se non contiene alcun sottografo che sia un'espansione di K_5 o un'espansione di $K_{3,3}$.”

Nel formalismo usato nell'enunciazione del teorema si indica con \mathbf{K}_n un *grafo completo con n nodi* e con $\mathbf{K}_{n,m}$ un *grafo bipartito completo con $n+m$ nodi* (vedi Figura 21).

Il “*problema delle tre case e delle tre forniture*” è prototipo per problemi che si occupano, tra l'altro, di determinare:

- **Layout** di reti elettriche, telefoniche o idriche;
- **Connessioni ottimizzate** tra i componenti di circuiti stampati e di circuiti integrati;
- **Connessioni ottimizzate** di calcolatori in una rete telematica.

²⁴ **Kazimierz Kuratowski** (1896–1980) è stato un matematico polacco, la cui ricerca si è focalizzata, e ha dato numerosi ed importanti contributi, allo studio delle strutture metriche e topologiche astratte.

Le reti di Petri

Introduzione ed elementi costitutivi

Le reti di Petri²⁵ sono una rappresentazione matematica di un **sistema distribuito discreto**²⁶; in particolare con **sistema distribuito**, in questo contesto, si intende quello che in informatica è descrivibile come un sistema di elaborazione dati, composto da hardware e software, costituito da processi interconnessi che comunicano solo attraverso messaggi opportunamente strutturati.

Per meglio comprendere l'importanza delle reti di Petri nello studio dei sistemi distribuiti si elencano alcune delle caratteristiche principali che definiscono tale tipo di sistema:

- **Eterogeneità** dei processi, possono essere scritti e implementati in linguaggi diversi e su sistemi operativi diversi, nonché avvalersi di hardware diversi;
- **Scalabilità**, all'aumentare del carico operativo il sistema è in grado di mantenere prestazioni equivalenti in termini di *latenza*²⁷ e *throughput*²⁸;
- **Nessun clock globale**, impossibilità di sincronizzare i timer di tutti i processi, generando l'impossibilità di definire in modo univoco un ordine temporale degli eventi che accadono nel sistema. L'impossibilità di definire la sequenza temporale degli eventi in modo certo e univoco è principalmente dovuta all'eterogeneità degli hardware utilizzati e delle relative frequenze di clock;
- **Possibili fallimenti** dei processi sia per improvvisi crash o per cause non deducibili.

I sistemi distribuiti, e le reti di Petri ne tengono conto, possono essere sincrono o asincroni; un **sistema** si dice **sincrono** se è possibile calcolare in modo preciso:

1. **Tempo massimo e minimo** per l'esecuzione di un processo
2. **Intervallo** di tempo massimo di trasferimento di un messaggio dalla sorgente al destinatario
3. **Deviazione massima** dal tempo reale di ciascun orologio (*clock drift rate*) delle diverse componenti del sistema.

I sistemi per cui non è possibile calcolare tali valori si dicono **asincroni**.

Per dare una definizione formale di una rete di Petri, date queste premesse, si può affermare che una rete di Petri, o **P-rete**, in prima approssimazione, può essere definita come un **grafo orientato e bipartito** in grado di descrivere processi in termini di comportamenti ed interazioni.

Con un approccio formale si può anche affermare che una **P-rete**, detta anche **rete Posto/Transizione** o **rete P/T**, è una rappresentazione matematica descrittiva di un sistema distribuito discreto, utilizzando uno schema a grafo che è composto da **nodi posto** e **nodi transizioni** e archi che connettono posti a transizioni e transizioni a posti. (vedi Figura 23.A).

Il **nodo posto** da cui *esce un arco* è detto **posto di input** e con analogo formalismo il **nodo posto** sul quale *incide un arco* è detto **posto di output**.

²⁵ **Carl Adam Petri** (1926 – 2010) è stato un matematico e informatico tedesco la cui ricerca a partire dalla comunicazione con gli automi, soggetto della sua tesi di dottorato a Bonn nel 1962, introduce e sviluppa una teoria sulle reti, chiamate poi reti di Petri. La teoria delle reti di Petri è fondamentale per lo sviluppo nel campo della computazione parallela e lo studio di sistemi complessi e **workflow**, inteso come automatizzazione di un processo.

²⁶ Con il termine **discreto** si definisce, in modo intuitivo e non rigoroso, un sistema costituito da elementi isolati.

²⁷ **Latenza** o **tempo di latenza** è l'intervallo di tempo che intercorre tra l'istante t_0 in cui arriva l'input al sistema e l'istante t_1 in cui il sistema rende disponibile l'output.

²⁸ **Throughput (THR)** o **capacità di trasmissione** di un canale di comunicazione si intende la capacità di trasmissione utilizzata effettivamente. $THR = \eta \times C$ dove η indica l'efficienza del protocollo usato e C la capacità di trasmissione. Il **THR** si misura in bit al secondo (*bit/s*).

Nelle rappresentazione di sistemi distribuiti discreti con reti di Petri, ogni **nodo posto**, può contenere anche *uno o più marche*, i **token**, (vedi Figura 23.B) per caratterizzare la configurazione di ogni posto e la distribuzione di *token* nei diversi posti di una P-rete è detta **marcatura**, in un linguaggio si può dire che una *marcatura* è definita come una *assegnazione di token a una P-rete*.

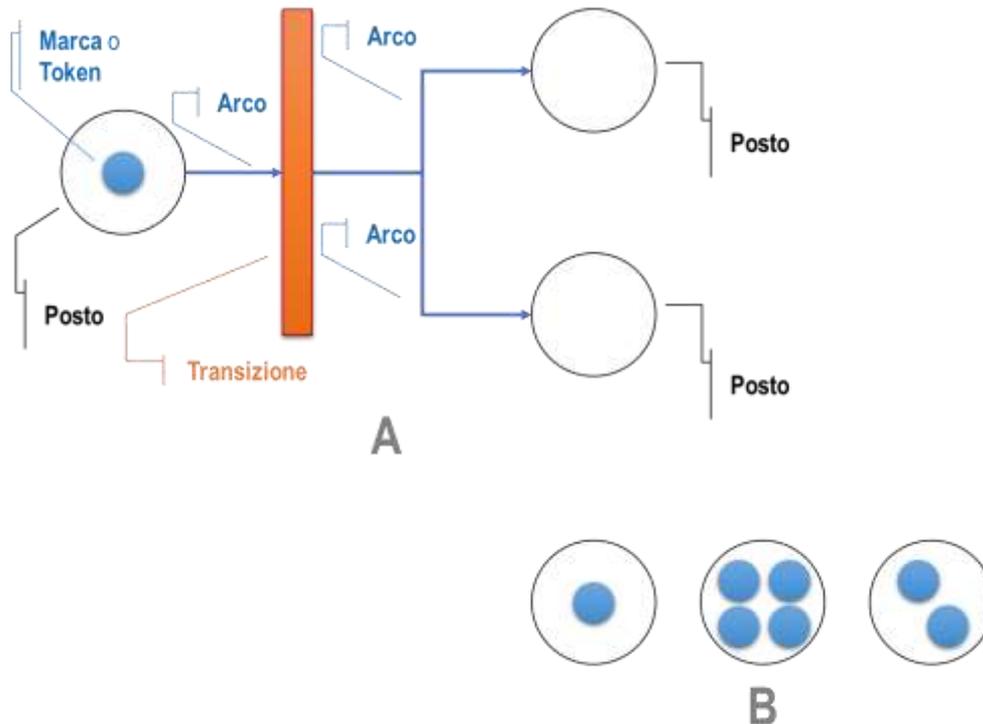


Figura 23- Rappresentativo di una struttura posto-transizione-posto i una P-rete (A); nodi posto che contengono token (B)

I token presenti nei singoli posti della rete possono variare per numero e posizione durante l'esecuzione dei processi schematizzati.

A partire dalle considerazioni fatte e dai concetti analizzati è possibile comprendere una definizione formale²⁹, da un punto di vista prettamente matematico, di una rete di Petri come la 6-tupla³⁰:

$$(S, T, F, M_0, W, K)$$

- **S**: insieme dei posti
- **T**: insieme delle transizioni
- **F**: insieme di archi detto "relazioni di flusso", insieme che vincola un arco a non connettere due posti o due transizioni; ogni arco ha sempre come estremi un posto e una transizione: $F \subseteq (S \times T) \cup (T \times S)$
- **M₀**: una possibile marcatura iniziale dove in ogni posto $s \in S$ ci sono $n \in \mathbb{N}^{31}$ [$S \rightarrow \mathbb{N}$]
- **W**: insieme dei pesi associati agli archi, un insieme che ad ogni arco $f \in F$ associa un valore non negativo $n \in \mathbb{N}^{+32}$ [$F \rightarrow \mathbb{N}^+$]; il valore n indica:
 - i token che "consuma" una transizione nei suoi posti di input
 - i token che sono "prodotti" da una transizione nei relativi posti di output

²⁹ Desel J., Juhas G., *What Is a Petri Net? Informal Answers for the Informed Reader*, Hartmut Ehrig et al. (Eds.): Unifying Petri Nets, LNCS 2128, pp. 1-25, 2001.

³⁰ Una **tupla**, in informatica e in particolare nella teoria dei data base è l'elemento generico di una relazione; usando una definizione formale "una tupla è una funzione su un insieme di attributi X, che ad ogni elemento di X associa un valore appartenente al dominio dell'attributo".

³¹ \mathbb{N} è l'insieme dei numeri naturali compreso lo 0: $\mathbb{N} = \{0, 1, 2, 3, 4, \dots\}$; altre notazioni equivalenti \mathbb{N}_0

³² \mathbb{N}^+ è l'insieme dei numeri maggiori di 0: $\mathbb{N}^+ = \{1, 2, 3, 4, \dots\}$; altre notazioni equivalenti \mathbb{N}^* , \mathbb{N}_+ e \mathbb{N}^+

- **K**: insieme, detto “*restrizioni sulla capacità*”, che assegna ad ogni posto $s \in S$ un valore non negativo $n \in \mathbb{N}^+$ che definisce il numero massimo di token che possono essere presenti

Nelle P-reti le azioni delle transizioni sono rappresentate dalle azioni che compiono sui token presenti nel relativo posto di input secondo **regole di scatto o regole di sparo, firing rules**. Le regole di scatto, la *transizione può scattare*, sono attive quando una transizione è **abilitata**; una transizione si considera abilitata se in ogni suo posto di input sono presenti tutti i token necessari.

Lo scatto di una transizione opera, con processi automatici, posizionando i token secondo le regole di scatto da i posti di input ai posti di output; la sequenza di scatti avviene secondo le regole predefinite e fino a quando la transizione è abilitata (vedi Figura 24).

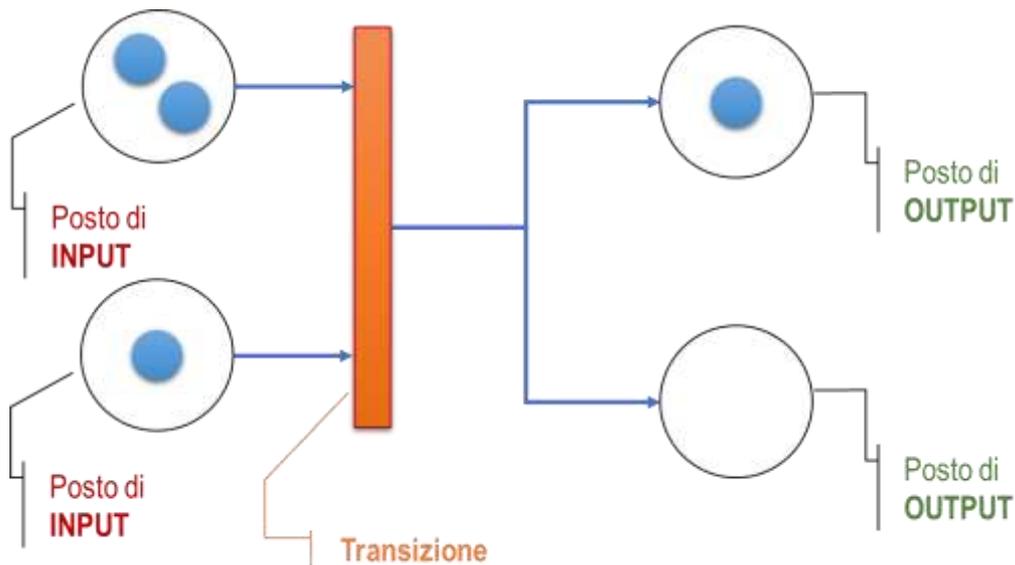


Figura 24 - Schema di transizione attiva

La definizione del concetto di **transizione** e **regole di scatto** inquadra le P-reti come **reti non deterministiche** e permette di comprendere meglio come le reti di Petri siano un ottimo strumento logico per la schematizzazione e la modellizzazione delle dinamiche dei **sistemi concorrenti**³³.

Le P-reti, come reti non deterministiche, e in funzione del concetto di transizione attiva si può descrivere attraverso le seguenti regole:

se in un determinato istante t_n ci sono più transizioni abilitate, allora ne può scattare una qualsiasi

- in nessun istante t_n è certo che una transizione attiva scatti
- una transizione attiva può scattare:
 - **immediatamente**, la transizione scatta senza consumare tempo
 - **dopo un intervallo di tempo** ΔT qualsiasi (tempo di attesa o **firing delay**), se è ancora abilitata
 - **mai**, la transizione non scatta in nessun istante t_n , anche se abilitata

³³ La **concorrenza**, come termine informatico, identifica un aspetto caratterizzante i sistemi di elaborazione dati; tali sistemi presentano configurazioni in cui è possibile che più processi computazionali, **thread**, siano in esecuzione simultanea. Un sistema in cui siano presenti concorrenze è chiamato **sistema concorrente** o **sistema a concorrenza**. Le concorrenze possono dare origine a interazione tra processi che necessitano delle stesse risorse. Esempio notevole dei sistemi concorrenti, sistemi di elaborazione dati, sono i **sistemi operativi**.

Categorizzando le configurazioni delle configurazioni che possono assumere le transizioni e i relativi scatti, nelle reti di Petri, si hanno **sequenze**, **concorrenze**, **conflitti** e **confusioni**. Si ha una **sequenza** in una P-rete quando è necessario che debba scattare una transizione t_i prima che sia possibile che scatti una transizione t_j (vedi Figura 25).

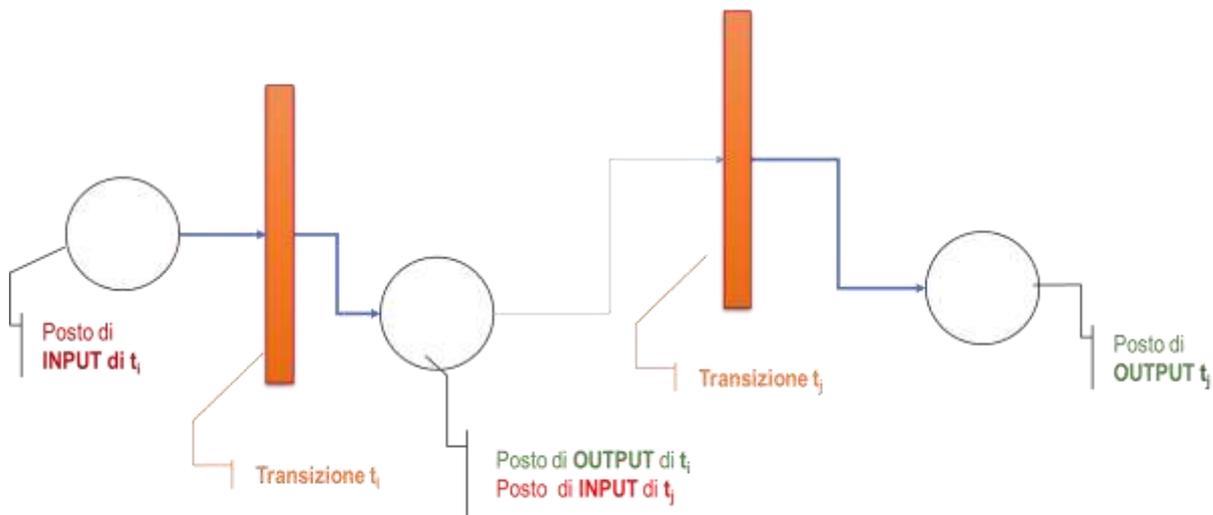


Figura 25- Schema semplificato di **sequenza** di 2 transizioni t_i e t_j

La **concorrenza** in una P-rete si ha quando una transizione ha più posti di input; si parla anche, da un punto di vista delle transizioni di:

- **concorrenza strutturale** (vedi Figura 26.A), 2 transizioni, t_i e t_j non hanno posti di input in comune, per cui gli scatti dell'una non interferiscono sugli scatti dell'altra
- **concorrenza effettiva** (vedi Figura 26.B), 2 transizioni, t_i e t_j hanno 1 o più posti di input in comune, ma esiste almeno una marcatura della rete cui gli scatti dell'una non rende disattiva l'altra.

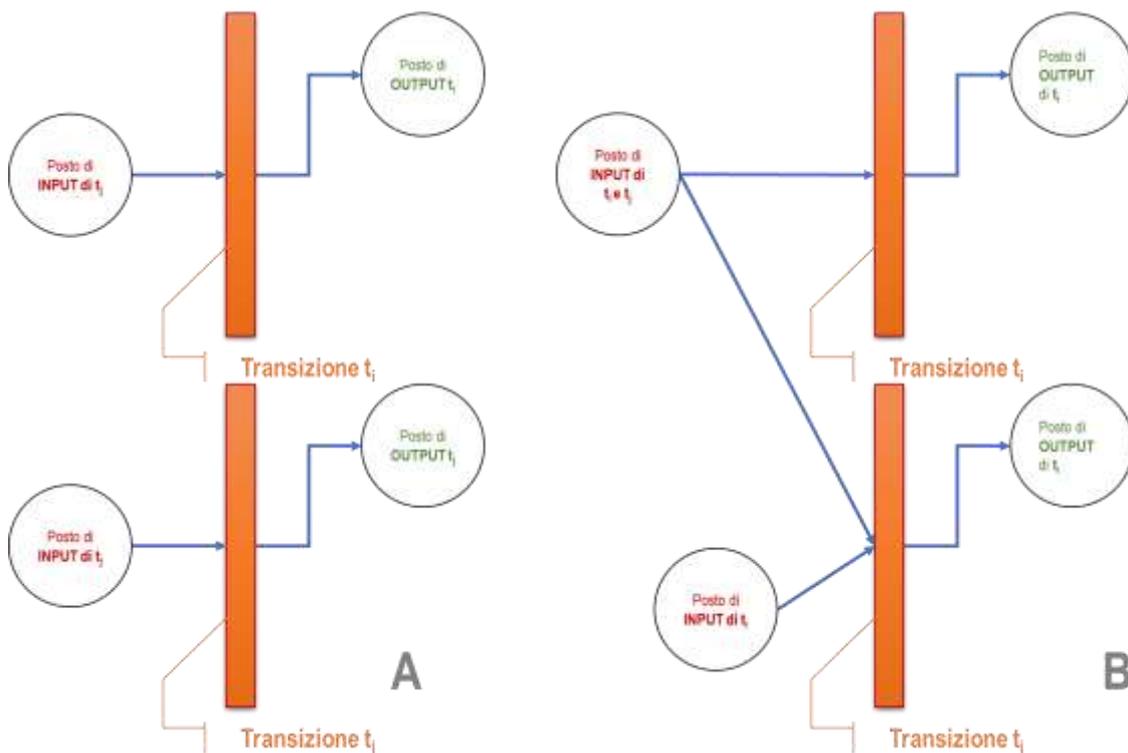


Figura 26 - Esempi di concorrenza strutturale (A) e concorrenza effettiva (B)

Si definisce **conflitto** (vedi Figura 27.A), in una P-rete, una struttura in cui un posto è *posto di input* per più transizioni, mentre una configurazione in cui un posto è *posto di input* per più transizioni delle quali almeno una ha più *posti di input* si definisce **confusione** (vedi Figura 26.B).

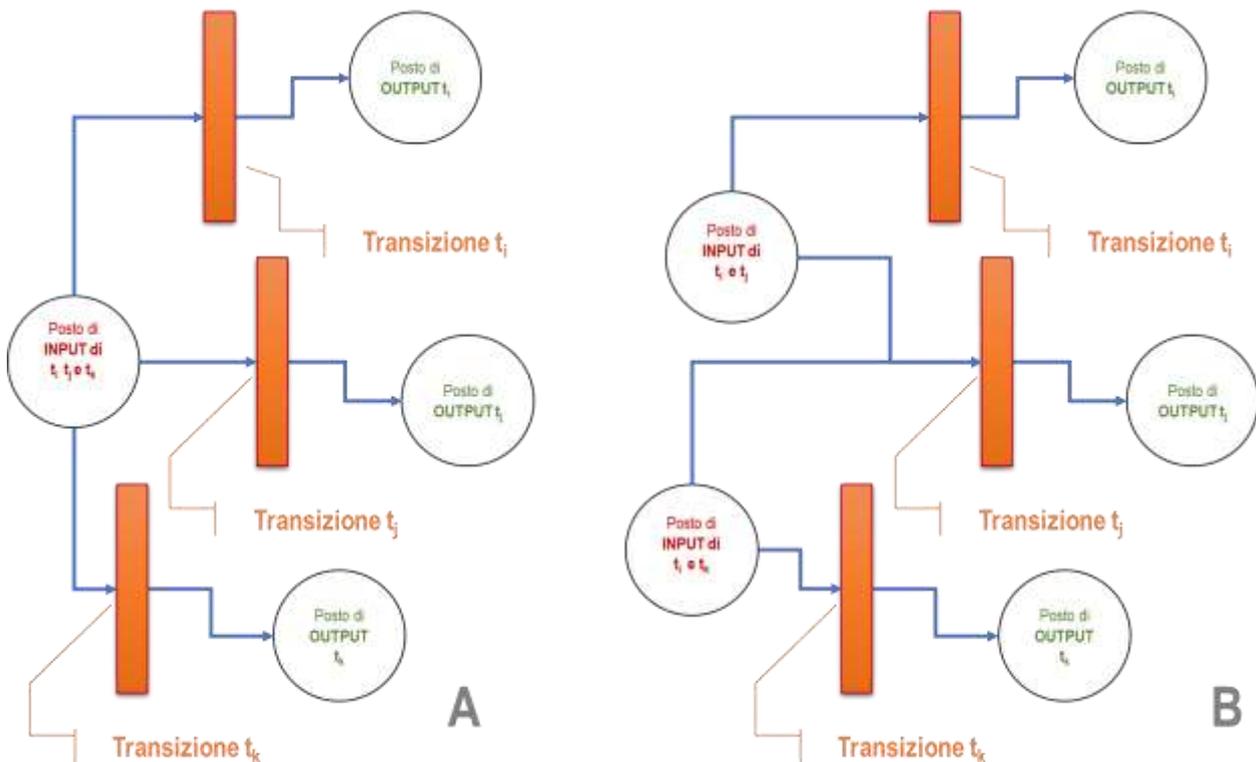


Figura 27 - Esempi di conflitto (A) e confusione (B)

Proprietà e tipologie delle P-reti

Le reti di Petri sono classicamente caratterizzate da quattro proprietà:

- **raggiungibilità** (*reachability*): una marcatura M_q è raggiungibile se esistono scatti che la rendono raggiungibile a partire da M_0 ; questa proprietà si avvale dello strumento del grafo di raggiungibilità che schematizza tutti i possibili stati e le relative correlazioni, mostrando quali stati sono raggiungibili e a partire da quali stati iniziali, avvalendosi delle proprietà dei grafo e del concetto di cammino in un grafo;
- **limitatezza** (*boundedness*): definito che un posto è detto k -limitato quando si può definire un numero naturale k come massimo numero di token che possono essere presenti in quel posto per una qualsiasi marcatura possibile, una rete si dice limitata se tutti i suoi posti sono limitati;
 - **sicurezza** (*safe*) : una rete di Petri **k -limitata** con $k = 1$ (1-limitata) si dice sicura o safe;
- **vitalità** (*liveness*): è una proprietà delle P-reti relativa alle transizioni; il grado di vitalità è definito attraverso l'analisi dell'attivabilità di una transizione all'interno di una marcatura raggiungibile; la definizione di vitalità permette di affermare che "una P-rete è viva se, qualunque sia la marcatura M_q raggiunta a partire da M_0 , da M_q è sempre possibile far scattare qualunque transizione $t \in T$ della rete a seguito di una ulteriore sequenza di scatti"; si possono definire i gradi di vitalità di una transizione $t \in T$ utilizzando la seguente scala:
 - **0 (Lo-live, transizione morta)**: t non può scattare in nessuna marcatura raggiungibile;

- **1 (L1-live)**: esiste almeno una marcatura raggiungibile in cui t può scattare
- **2 (L2-live)**: per ogni numero intero $k \in \mathbb{N}^+$ esiste almeno una marcatura raggiungibile tale per cui t può scattare k
- **3 (L3-live)**: esiste almeno una marcatura raggiungibile tale per cui t può scattare infinite volte
- **4 (L4-live, transizione viva)**: t può scattare in tutte le marcature raggiungibili

In stretta correlazione alla vitalità di una rete e per riconoscerne la vitalità stessa, si parla anche di due strutture particolari:

- **sifone**: insieme di posti di una rete nei quali, durante l'esecuzione del processo, tende a perdere token e non ha la capacità di acquisire nuovi token
- **trappole**: insieme di posti di una rete nei quali, durante l'esecuzione del processo, tende ad acquisire token e non ha la capacità di perdere token ; struttura duale del sifone

Una rappresentazione formale di *sifoni* e *trappole* è la seguente:

Sifone	$\bullet S \subseteq S \bullet$
Trappola	$S \bullet \subseteq \bullet S$

Raggiungibilità e grafo di raggiungibilità

Il concetto di raggiungibilità in una rete di Petri ha come strumento di analisi e verifica il **grafo di raggiungibilità** (vedi Figura 28), un grafo orientato in cui i nodi rappresentano le possibili marcature e gli archi le transizioni che mettono in relazione due marcature, cioè modificano la marcatura del grafo e in particolare permettono di passare da una marcatura M_i ad una marcatura M_j .

Il grafo di raggiungibilità è, in pratica, uno strumento dei processi schematizzabili e rappresentati dalle reti di Petri, per individuare:

- raggiungibilità di marcature, stati, indesiderati
- raggiungibilità di marcature, stati, desiderati
- raggiungibilità di marcature *deadlock*, cioè una marcature che blocca il processo, lo termina, da cui non si può proseguire

La costruzione dei grafi di raggiungibilità viene fatta attraverso algoritmi di ricerca in ampiezza, *breadth-first*³⁴, potendo essere tali strutture di larghezza illimitata.

Nei numerosi casi in cui venga generato un grafo di raggiungibilità infinito, per operare con l'approssimazione di un grafo finito si introduce la variabile ω che rappresenta il numero *illimitato* di token che possono essere associati ad un singolo posto; con questo artificio matematico si ottiene di collassare in un unico nodo infinite marcature.

Le indicazioni più importanti deducono dalla generazione dei grafi di raggiungibilità, utilizzando il concetto di cammino, sono quelle relative alla possibilità ovvero l'impossibilità di avere come possibile una marcatura "indesiderata"; per questo se obiettivo principale della ricerca è la determinazione dell'impossibilità di raggiungere una marcatura, stato, indesiderato si utilizzano algoritmi per la ricerca di "*stati indesiderati*".

Limitatezze e reti safe

Considerando una generica P-rete (S, T, F, M_0, W, K) e analizzando gli stati che si generano durante la sua evoluzione si potrà notare che, in funzione delle transizioni che scattano, si possono generare e distruggere token nei posti di input e output.

³⁴ Gli algoritmi BFS (*Breadth First Search*) sono metodi di ricerca **non informati**, senza informazioni sul dominio utilizzati per grafi con l'obiettivo di esaminare tutti i nodi del grafo in maniera sistematica per individuare il nodo cercato. Questi algoritmi sono di tipo *euristico*, quindi non porta a soluzioni ottime.

Se in una rete di Petri il numero di token presente in un posto, in ogni marcatura possibile, è minore o al massimo uguale a k , con k numero naturale e positivo ($k \in \mathbb{N}^+$) il **posto** si dice **k -limitato** e per estensione se ogni posto è k -limitato la **rete** è definita **k -limitata**; la limitatezza di una rete la fa definire **safe** se $k=1$.

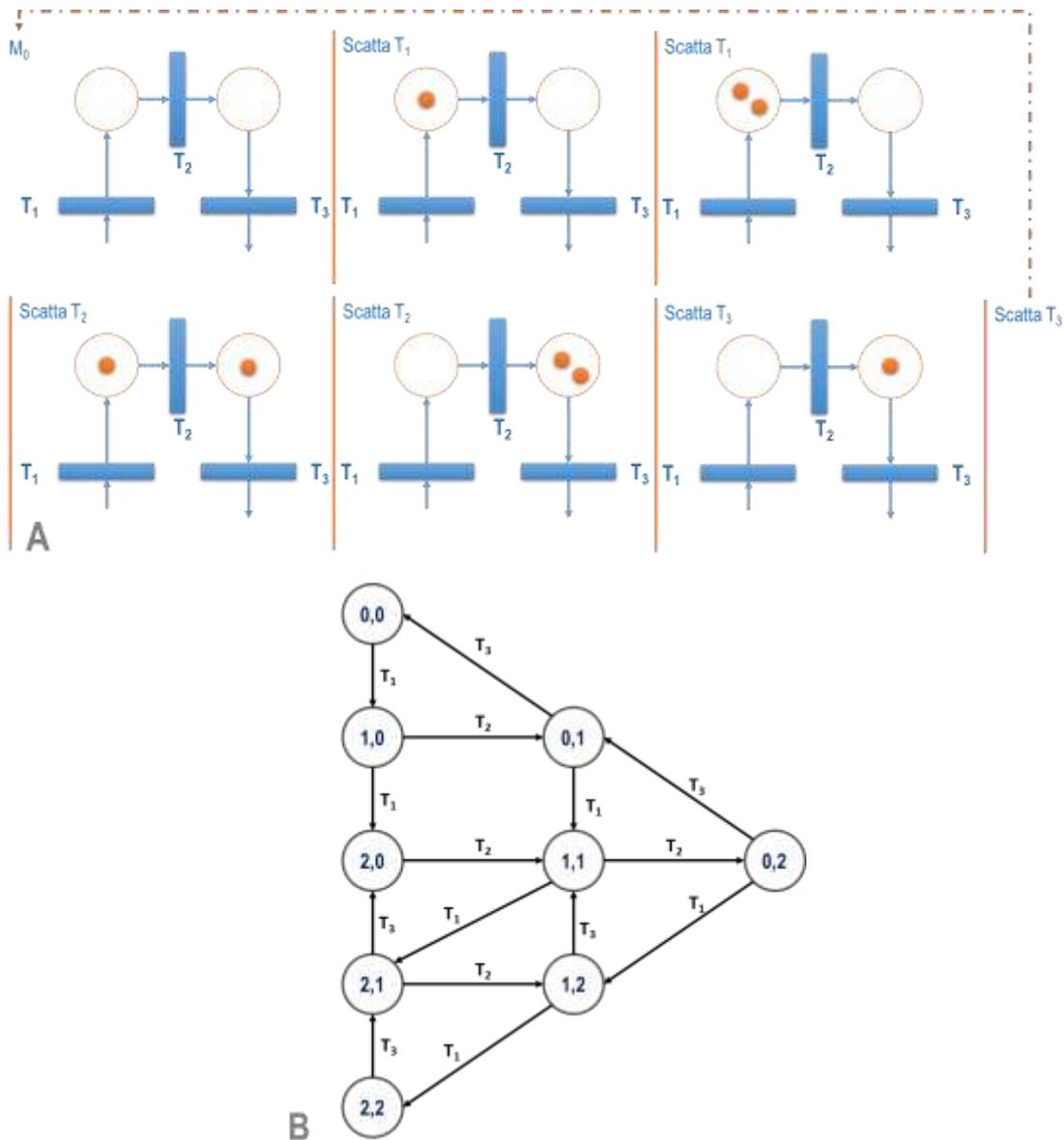


Figura 28 - Grafo di raggiungibilità (B) relativo a una rete di Petri descritta in termini di transizione (A)

Il concetto di limitatezza dei posti di una rete di Petri, e della rete stessa, è una schematizzazione formale del concetto di risorse presenti nel sistema; la limitatezza di una rete è definita dalla marcatura considerata come iniziale, M_0 , cioè posizionare 6 token in un posto nella marcatura M_0 , rende naturalmente impossibile considerare la rete 3-limitata.

Se in una generica rete di rete di Petri un posto non è limitato per nessun $k \in \mathbb{N}^+$ si dice illimitato e una P-rete che abbia almeno un posto illimitato si dice **rete illimitata**.

È sempre possibile rendere limitata una rete (vedi Figura 29) ad un valore k prefissato utilizzando la tecnica dei **posti complementari**; il metodo dei *posti complementari* consiste nell'operare sulla rete di Petri secondo i seguenti passi, fissato un valore $k \in \mathbb{N}^+$:

- **definizione** del posto complementare P' per ogni posto della rete

- **collegamento** del posti P' alle stesse transizioni del posto P , con le medesime caratteristiche
- **definizione** della marcatura iniziale M_0 , affinché in numero di token in P' , $M_0(P')$, dato k e il numero di token in P in M_0 , $M_0(P)$, sia definito dall'uguaglianza $M_0(P') = k - M_0(P)$

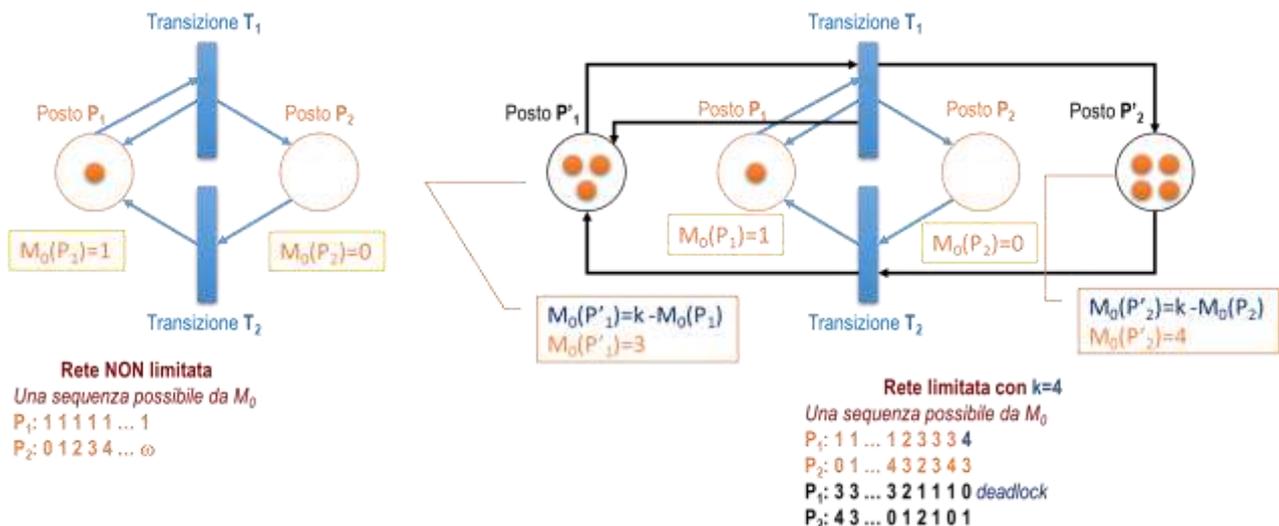


Figura 29 - Esempio di metodo dei posti complementari con $k=4$

in questo modo la somma dei token in P e P' è il valore costante k , rendendo così nel complesso la rete k -limitata; questo permette di estendere le considerazioni e le analisi delle reti limitate alle reti non-limitate.

Altre proprietà e caratteristiche delle reti di Petri e proprietà decidibili

Le reti di Petri presentano ulteriori strutture e configurazioni notevoli che sottolineano la versatilità e la potenza rappresentativa di problemi dinamici e indeterminati.

Se, ad esempio, si analizzano insieme di posti di una rete di Petri possiamo parlare di:

- **P-invariante**: un insieme di posti di una P-rete in cui è costante la somma dei token contenuti in ogni marcatura M_i appartenente all'insieme delle marcature raggiungibili
- **T-invariante**: un insieme di sequenze di scatti possibili che riportano la P-rete alla marcatura iniziale M_0 ; le T-invarianti si possono definire come duale delle P-invarianti

Esaminando l'evoluzione della quantità di token presenti nel sistema, le caratteristiche delle marcature e l'evoluzione delle transizioni è possibile definire ulteriori caratteristiche che descrivano le reti di Petri:

- **conservazione (proprietà dei token)**: il numero totale di token presenti nel sistema ad ogni marcatura è costante
- **persistenza (proprietà delle transizioni)**: una transizione abilitata rimane abilitata fino a che non scatta, ovvero lo scatto di una qualsiasi transizione non disabilita nessuna altra transizione abilitata nel sistema
- **reversibilità (proprietà delle marcature)**: esistono sequenze di transizioni tali per cui la marcatura M_0 , marcatura iniziale, è raggiungibile a partire ogni altra marcatura M_i appartenente all'insieme delle marcature raggiungibili
- **stato di accettazione (proprietà delle marcature)**: una marcatura M_j tale per cui esistono sequenze di transizioni che la rendono raggiungibile a partire ogni altra marcatura M_i appartenente all'insieme delle marcature raggiungibili

Quando si analizzano le proprietà nel contesto delle reti di Petri è significativo porre la domanda: **quali proprietà sono decidibili**³⁵?

Ci sono **problemi** importanti di **logica matematica** che sono decisionali e **indecidibili** come ad esempio i “**teoremi di incompletezza**” di Göedel formalmente enunciati:

Primo Teorema di incompletezza

“In ogni teoria matematica *T* sufficientemente espressiva da contenere l'aritmetica, esiste una formula φ tale che, se *T* è coerente, allora né φ né la sua negazione $\neg\varphi$ sono dimostrabili in *T*.”

Secondo Teorema di incompletezza

“Sia *T* una teoria matematica sufficientemente espressiva da contenere l'aritmetica: se *T* è coerente, non è possibile provare la coerenza di *T* all'interno di *T*.”

Da un punto di vista descrittivo i teoremi di completezza postulano che:

- è possibile costruire una proposizione sintatticamente corretta che non può essere né **dimostrata** né confutata all'interno dello stesso sistema (*primo teorema*)
- nessun sistema coerente può essere utilizzato per dimostrare la sua stessa coerenza (*secondo teorema*)

Con queste premesse e definizioni generali è interessante elencare le **proprietà decidibili**:

- *raggiungibilità*
- *vitalità*
- *limitatezza*

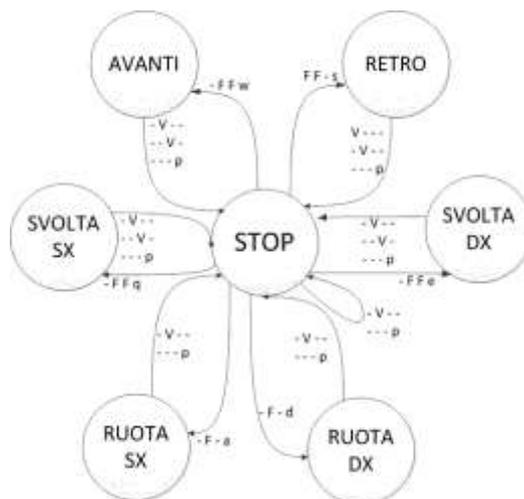


Figura 30 - Esempio di schema di Macchina a Stati Finiti (fonte iralab - URL: http://irawiki.disco.unimib.it/irawiki/index.php/INFIND2011/12_Rover_Robot)

Il formalismo definito dalle reti di Petri, in relazione alla decidibilità delle proprietà, le situa tra **ASF**³⁶ (Automi a Stati Finiti, vedi Figura 30), dove tutte le proprietà sono decidibili, e le **macchine di Turing**³⁷, (vedi Figura 31) dove pressoché nessuna proprietà è decidibile; il

³⁵ Si parla di **problemi decisionali** quelli in cui si ha un algoritmo che decide se una proprietà è verificata o meno, algoritmi per cui l'output finale è di tipo logico Vero/Falso. Se un problema algoritmico non ha soluzione si dice che non è computabile, mentre nel caso di problemi decisionali si dice, se non ha soluzione, **indecidibile**.

³⁶ Un **ASF** (Automa a Stati Finiti) o macchina a stati finiti, **FSM** (Finite-State Automaton o il plurale Automata) è un modello matematico adatto alla descrizione formale e precisa di sistemi e utilizzato in maniera specifica in informatica e ricerca operativa; un automa a stati finiti è rappresentato da un *grafo*. Si può affermare che è un caso particolare di una macchina di Turing.

³⁷ Una **Macchina di Turing** è uno strumento teorico, introdotto da **Turing** nel 1936, molto potente che trova vasta applicazione *nella teoria della calcolabilità* nello studio della *complessità degli algoritmi*. Lo scopo iniziale che ha portato Turing a definire questo strumento era dare risposta al *problema della decisione* (*Entscheidungsproblem*) enunciato da **Hilbert** nel 1928. La macchina di Turing è una macchina ideale che opera

formalismo delle reti di Petri ha come punto di forza il fatto che coniuga **espressività**³⁸ e **analizzabilità**.

Come limite evidenziabile, in questo contesto, si evidenzia che, in generale, le reti di Petri non risolvono il **Test a Zero**³⁹, possibile solo per reti limitate applicando la tecnica dei posti complementari; l'applicabilità del *Test a Zero* per reti limitate è altresì deducibile dal fatto che tali tipologie di reti è equivalente a un *automa a stati finiti*.



Figura 31- Esempio di Macchina di Turing (fonte RAI Cultura Filosofia - URL:<http://www.filosofia.rai.it/articoli/la-macchina-di-turing/13882/default.aspx>)

Le estensioni di reti di Petri che sono in grado di strutturare e formalizzare *Test a Zero* sono equivalenti a *macchine di Turing*, per le quali diventano indeducibili praticamente la totalità delle proprietà.

Estensioni delle reti di Petri e reti temporizzate

A partire dalle Reti di Petri “classiche” è possibile generare nuove reti come estensione delle P-reti, alcune che introducono nuove proprietà altre che invece non introducono nuove caratteristiche e quindi considerabili come trasformazioni matematiche delle reti esistenti; tali trasformazioni hanno lo scopo di rendere più semplice o evidenti strutture e regole di reti esistenti, mantenendone il significato originale.

Le principali estensioni delle rete di Petri sono:

- **reti con archi inibitori**
- **reti colorate**
- **reti con priorità**

Per definire una **rete con archi inibitori**, va premesso che un arco inibitore (vedi Figura 32) è un arco che entra in una transizione T_x e formalizza la condizione che T_x è abilitata solo se il posto di input P_i è privo di token, vuoto.

La rete con archi inibitori soddisfa l'esigenza di condizionare l'abilitazione di una transizione al fatto che un determinato posto di input per la transizione in esame è vuoto.

su dati utilizzando un nastro di lunghezza infinita e riferendosi ad un insieme predeterminato di leggi formalmente ben definite. La capacità di una macchina di Turing si può dunque riassumere nella possibilità che ha di eseguire algoritmi utilizzando un nastro su cui scrivere simboli.

³⁸ L'espressività in un linguaggio misura la facilità e la rapidità con cui è possibile scrivere un algoritmo utilizzandolo. Un linguaggio presenta buona espressività se permette la scrittura di un algoritmo con poche e chiare istruzioni.

³⁹ Per **Test a Zero** si intende la possibilità di abilitare una *transizione* T_1 , anche quando un determinato posto di ingresso per T_1 non contiene token.

Usando il semplice formalismo di base di una P-rete infatti non è possibile imporre ad una funzione di consumazione di consumare tutti i token prima che il produttore di token sia nuovamente abilitato.

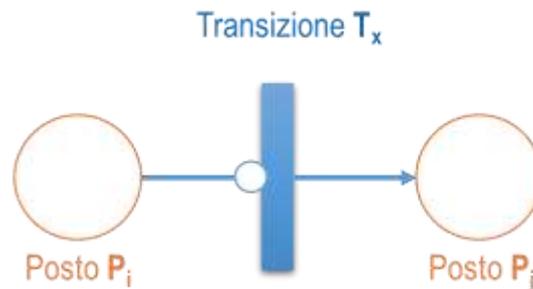


Figura 32- Schema di arco inibitore per la transizione T_x

La rete con un arco inibitore può, invece, imporre la condizione appena descritta e assumere così le potenzialità di una equivalente Macchina di Turing, di cui è possibile simulare il funzionamento tramite una rete con archi inibitori.

Un esempio di utilizzo di una rete ad archi inibitori è un **allocatore** di risorse, un sistema che ha una unica risorsa, inizialmente libera, disponibili e richieste di due tipi: alta priorità e bassa priorità; le possibili condizioni e regole del sistema sono:

- **risorsa libera e una richiesta a bassa priorità:** viene soddisfatta la richiesta e la risorsa diventa occupata
- **risorsa occupata:**
 - non è possibile ricevere richieste a bassa priorità
 - è possibile ricevere richieste ad alta priorità

La rete di Petri in questa situazione tiene conto, tramite token in un posto, delle richieste ad alta priorità togliendo un token ogni volta che una di tali richieste è soddisfatta fino a che non ci sono più token nel posto.

In questo caso l'utilizzo dell'arco inibitore permette alla transizione che allocare le risorse a bassa priorità quando non ci sono risorse ad alta priorità, cioè è vuoto il posto di input che contiene un token per ogni richiesta ad alta priorità.

Le **reti colorate** sono reti di Petri in cui si possono incontrare diversi tipi di token, che per convenzione hanno colori diversi; in particolare i token sono coinvolti nell'abilitazione delle transizioni che prevedono archi etichettati, per cui le transizioni necessitano di particolari tipi di token, token del colore giusto. Se una rete colorata prevede un numero finito di token è riconducibile ad una normale rete di Petri, mentre se il numero di colori è illimitato ci si trova di fronte ad un formalismo equivalente ad una Macchina di Turing.

Nelle reti colorate non è detto che lo scatto di una determinata transizione generi una unica marcatura.

Le **reti con priorità** sono estensione di una rete di Petri in cui si introduce il concetto di priorità tra le diverse transizioni.

In particolare se due o più transizioni sono attive, le transizioni a minor priorità non possono scattare se non sono scattate tutte quelle a priorità superiore; c'è un procedimento di tipo deterministico con cui si scelgono le transizioni da far scattare, in funzione della loro priorità.

In un gruppo di transizioni con la medesima priorità il procedimento è, ovviamente, ancora di tipo non-deterministico.

Anche questa estensione delle reti di Petri è *Turing-equivalente*, per cui le tre proprietà fondamentali della rete: raggiungibilità, limitatezza, vitalità diventano indecidibili.

C'è un teorema che afferma che: "è sempre possibile trasformare una rete con priorità in una rete con archi inibitori, e viceversa".

L'importanza del non-determinismo che è possibile formalizzare con le reti di Petri in diversi casi è legato ad una variabile temporale fondamentale per il sistema stesso.

La necessità indotta della variabile tempo per una buona efficace descrizione di un sistema logico, per superare il limite dovuto al fatto che le P-reti danno solo un ordinamento temporale parziale fra gli eventi ha fatto sì che si introducessero le **reti di Petri temporizzate**. Le reti di Petri temporizzate, come visto in precedenza per altre estensioni, divengono con l'introduzione della variabile tempo *Turing-equivalenti* per cui le tre proprietà fondamentali della rete: raggiungibilità, limitatezza, vitalità, anche in questo caso, diventano indecidibili.

A loro volta le reti temporizzate si dividono in due categorie:

- **rete temporizzata deterministica**: intervalli temporali e valori di archi e posti sono dati in modo deterministico
- **rete temporizzata stocastiche**⁴⁰: intervalli temporali e valori di archi e posti sono dati in modo probabilistico

Le **reti temporizzate deterministiche** sono:

- **reti Time Petri Nets** (Merlin e Farber, 1976): in questo tipo di reti temporizzate ad ogni transizione è associata un'etichetta con due istanti di tempo definiti tempo minimo e tempo massimo (t_{min} e t_{max}); in queste reti per poter scattare una transizione deve rimanere attiva almeno t_{min} e se al tempo t_{max} è ancora attiva **deve** scattare; per capire il significato di t_{min} e t_{max} si consideri che in una rete di Petri classica, non temporizzata, ogni transizione è etichettata con $t_{min} = 0$ e $t_{max} = \infty$; per queste reti l'insieme delle marcature raggiungibili è contenuto al più uguale dell'insieme delle marcature raggiungibili nell'equivalente rete non temporizzata;
- **reti Timed Petri Nets** (Ramchandani, 1980): in questo tipo di reti ogni transizione ha una durata t , e ogni transizione **deve scattare** appena abilitata e lo scatto ha una **durata t** ; la durata t dello scatto comprende, all'inizio, la rimozione di tutti i token dai posti di input, una permanenza in stato di scatto per un tempo t e poi l'emissione di tutti i token nei rispettivi punti di output; questo modello di rete si rileva utile per la valutazione delle prestazioni di sistemi ciclici come ad esempio i protocolli di comunicazione;

Le reti stocastiche o **Stochastic Petri Nets**, introdotte nel 1982, sono caratterizzate dal fatto che ogni transizione ha un tempo di scatto non prefissato ma distribuito in modo esponenziale; in particolare questo significa che ad ogni transizione è associata una variabile aleatoria con valori reali e non negativi, valori in R^+ ; in termini di significato la variabile associata alla transizione rappresenta il tempo che intercorre dall'abilitazione allo scatto della transizione.

Le reti stocastiche così definite sono isomorfe⁴¹ ai *processi di Markov*⁴² con la variabile tempo continuo; in particolare il grafo di raggiungibilità di una rete stocastica limitata è isomorfo a *una catena di Markov finita*⁴³.

⁴⁰ **Stocastico**, dal greco $\sigma\tau\omicron\chi\alpha\sigma\tau\iota\kappa\acute{o}\varsigma$ che significa congetturale, è un termine che nel calcolo delle probabilità significa casuale, aleatorio. Si usa per descrivere e studiare situazioni che variano in base a leggi probabilistiche e non deterministiche.

⁴¹ In formalismo matematico un **isomorfismo** (dal greco $\acute{\iota}\sigma\omicron\varsigma$, *isos* - uguale, e $\mu\omicron\rho\phi\acute{\eta}$, *morphé*, -forma) è "un'applicazione biunivoca fra oggetti matematici tale che l'applicazione e la sua inversa siano omomorfismi". Da un punto di vista più descrittivo si può dire che "Si parla di isomorfismo quando due strutture complesse si possono applicare l'una sull'altra, cioè far corrispondere l'una all'altra, in modo tale che per ogni parte di una delle strutture ci sia una parte corrispondente nell'altra struttura; in questo contesto diciamo che due parti sono corrispondenti se hanno un ruolo simile nelle rispettive strutture" (Douglas Hofstadter - Gödel, Escher, Bach: Un'Eterna Ghirlanda Brillante, p. 54).

Questa tipologia di rete, infine, permette di calcolare alcune proprietà tra e quali:

- probabilità del verificarsi una condizione particolare
- valore atteso del numero di token
- valore medio, nell'unità di tempo, del numero di scatti

Nell'ambito delle reti stocastiche nel 1984 Ajmone Marsan, Balbo e Conte introducono le **Generalized Stochastic Petri Net**, reti in cui le **transizioni** sono o **temporizzate** o **immediate**.

Nelle *Generalized Stochastic Petri Net* le *transizioni temporizzate* agiscono come nelle reti stocastiche semplici mentre le *transizioni immediate* scattano appena attivate; quest'ultimo tipo di transizione approssimano le transizioni i cui tempi di esecuzione possono reputarsi trascurabili rispetto ai tempi di tutte gli altri.

In queste reti, inoltre si postula che nel caso in cui entrambi i tipi di transizione sono abilitate scattano solo quelle immediate e nel caso in cui ci siano due o più transizioni immediate in conflitto la scelta di quali far scattare è affidata alla *distribuzione di selezione*, una distribuzione di probabilità.

Questo tipo di reti ha il vantaggio di consentire una analisi semplificata del sistema, in funzione del fatto che è possibile trascurare le marcature con transizioni immediate, perché sono marcature che "durano poco".

Tipologie notevoli di P-reti

È possibile catalogare alcune tipologie notevoli di rete, che ne rappresentano i tipi principali, tra lo in relazione insiemistica (vedi Figura 33), di seguito elencheremo le più significative:

- **Macchine a Stati Finiti (SM: State Machine)**
- **Grafo Marcato (MG: Marked Graph)**
- **Scelta Libera (FC: Free Choice)** e **Scelta libera estesa (EFC: Extended Free Choice)**
- **Scelta Asimmetrica (AC: Asymmetric Choice)**

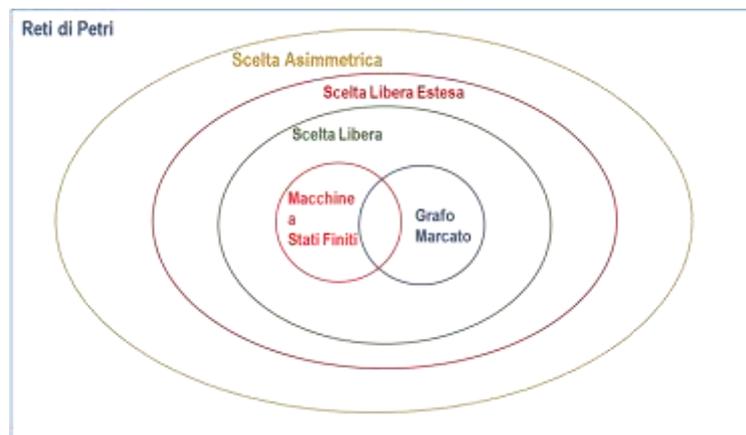


Figura 33 - Relazione tra le diverse tipologie notevoli di Reti di Petri

⁴² Un **processo di Markov**, definito dal matematico russo Andrej Andreevič Markov, è un processo stocastico dove la probabilità di transizione che determina il passaggio a un nuovo stato S_i del sistema dipende esclusivamente dallo stato precedente S_{i-1} e non dalle modalità con cui si è giunti allo stato S_{i-1} .

⁴³ Una **catena di Markov finita** è un sistema dotato di un numero finito n di stati S_1, S_2, \dots, S_n che soddisfi la seguente ipotesi: la probabilità che il sistema passi dallo stato i allo stato j al tempo $k - 1$ è $p_{ij}(k)$. Con un approccio descrittivo si può dire che una catena di Markov è un sistema in cui la probabilità di uno stato i al tempo k è determinata univocamente dallo stato del sistema al tempo $k - 1$ e non dagli avvenimenti precedenti lo stesso.

Una **macchina a stati finiti** è una P-rete tale per cui ogni transizione ha un solo posto di input e un solo stato di output, ossia, descrivendolo secondo posti e archi, uno ed un solo arco in entrata ed uno ed un solo arco in uscita per ogni posto-transizione. Una macchina a stati finiti, in formalismo matematico è definita:

$$\forall t \in T : |\bullet t| = |t \bullet| = 1 \quad (t \text{ transizione e } T \text{ insieme di tutte le transizioni})$$

Le macchine a stati finiti, per come sono caratterizzate le transizioni, non presenta **concorrenze** e non presenta **conflitti**; la marcatura iniziale presenta almeno un token e la rete è fortemente connessa, inoltre è una rete viva con almeno un token, e se il token è solo 1 è anche una rete safe.

Il **grafo marcato** è una P-rete definibili con il principio di dualità rispetto ad una macchina a stati finiti; in particolare un grafo marcato è una P-rete in cui ogni POSTO ha un solo arco in entrata e un solo arco in uscita, per questo si può affermare che la macchina a stati finiti è un grafo fortemente connesso e sempre limitato.

Un grafo marcato è una P-rete per cui è possibile affermare che, non ci sono **conflitti** ma possono esserci **concorrenze**. Un grafo marcato, in formalismo matematico è definito:

$$\forall p \in P : |p \bullet| = |\bullet p| = 1 \quad (p \text{ posto e } P \text{ insieme di tutti i posti})$$

Una P-rete **a scelta libera** è caratterizzata dal fatto che ogni arco a_k del grafo che rappresenta la rete è o l'unico che entra in un posto P_i o l'unico che esce da una transizione T_j . Questa configurazione permette di dire che nelle reti a scelta libera ci possono essere concorrenze e ci possono essere conflitti, ma mai contemporaneamente.

Per le reti a scelta libera si ha il seguente formalismo matematico:

$$\forall p \in P : (|p \bullet| \leq 1) \vee (\bullet p) = \{1\} \quad (p \text{ posto e } P \text{ insieme di tutti i posti})$$

Le P-reti che possono essere trasformate in reti a scelta libera, si dicono **reti a scelta libera estese**.

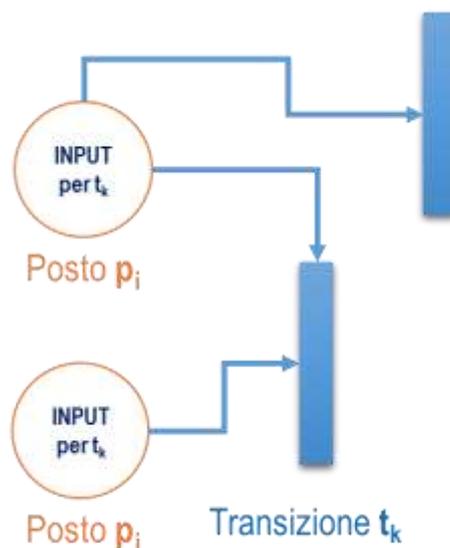


Figura 34 - Esempio di configurazione per rete a scelta simmetrica

Da ultimo, e forse le più complesse, si definiscono le reti a **scelta asimmetrica** (vedi Figura 34) caratterizzate dalla seguente regola: presi due posti nella rete p_i e p_j , posti di input per

una stessa transizione t_k , l'insieme delle transizioni per cui p_i è posto di input contiene l'insieme delle transizioni per cui p_j è posto di input, che da un punto di vista del formalismo matematico si esprime:

$$\forall p_1, p_2 \in P : (p_1 \bullet \cap p_2 \bullet \neq \emptyset) \rightarrow [(p_1 \bullet \subseteq p_2 \bullet) \vee (p_2 \bullet \subseteq p_1 \bullet)] \text{ (} p \text{ posto e } P \text{ insieme di tutti i posti)}$$

Problemi ed esempi per le reti di Petri

L'utilizzo delle reti di Petri è ben esemplificato attraverso due problemi che rappresentano anche tipologie generali di problemi a cui le P-reti possono essere applicate.

I due problemi principali che si prendono in esame sono:

- **problema dei filosofi a cena** o **problema dei 5 filosofi** (*dining philosopher problem*) enunciato da Edsger Dijkstra⁴⁴ nel 1965
- **problema del barbiere che dorme** o **il barbiere sonnolento**

Il **problema dei filosofi a cena** fu utilizzato da Dijkstra per rappresentare i problemi di sincronizzazione, ed in particolare per quanto concerne la concorrenza e la condivisione di risorse.

Il problema è descritto a partire dall'affermazione che cinque filosofi siedono ad una tavola rotonda con un piatto di spaghetti davanti, una forchetta a destra e una forchetta a sinistra o, in alcune descrizioni davanti a un piatto di riso con un bastoncino a destra e uno a sinistra (vedi Figura 35).

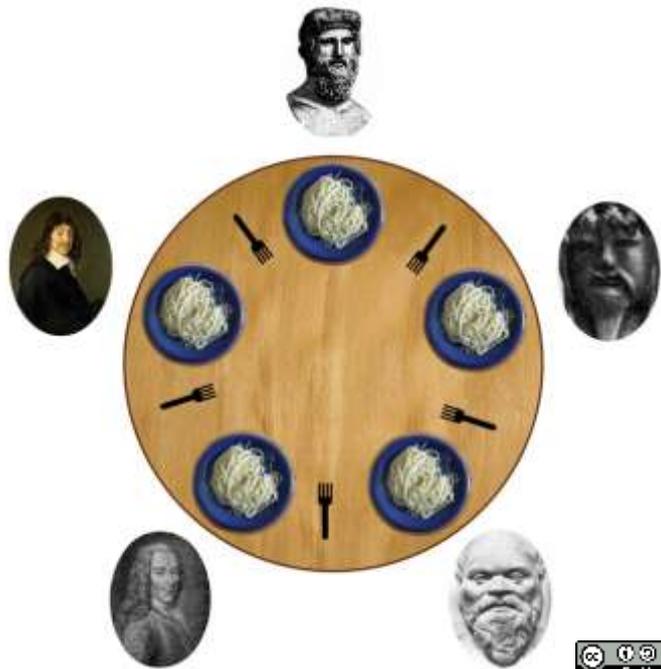


Figura 35 - Immagine che illustra problema dei 5 filosofi (dall'alto e in senso orario Platone, Confucio, Socrate, Voltaire e Cartesio) - Fonte Benjamin D. Esham / Wikimedia Commons

⁴⁴ **Edsger Wybe Dijkstra** (Rotterdam, 11 maggio 1930 – Nuenen, 6 agosto 2002) è stato un informatico olandese, noto soprattutto per i contributi all'informatica dell'algoritmo che porta il suo nome (*algoritmo di Dijkstra*) e per la definizione del concetto *informatico* di *semaforo*. L'algoritmo di Dijkstra si applica ai problemi di "cammino minimo" (problemi **SP** - Shortest Paths); il concetto informatico di **semaforo** è la definizione di un **tipo di dati astratto**, *Abstract Data Type*, che viene utilizzato dai sistemi operativi multitasking per gestire la sincronizzazione dell'accesso a risorse condivise da parte dei diversi processi o task.

Ogni filosofo alternativamente pensa e mangia e per mangiare deve avere contemporaneamente in mano 2 forchette, o 2 bacchette.

Le regole che devono essere rispettate a questo tavolo sono:

- ogni filosofo può prendere solo le due forchette, o bacchette, che stanno alla sua destra e alla sua sinistra, una per volta e solo se sono libere
- ogni filosofo non può sottrarre la forchetta, o bacchetta ad un altro filosofo che l'ha già acquisita ; questa regola è detta **no preemption**, ovvero *non c'è predominanza, nessun diritto di prelazione* di un filosofo rispetto ad un altro
- ogni filosofo attende, seppur affamato, fino a che non riesce a prendere le due forchette, o le due bacchette
- ogni filosofo, quando è sazio, posa nuovamente le forchette, o le bacchette, al loro posto e ritorna a pensare per un certo periodo

Una corretta programmazione, che risolve il problema, deve essere in grado di far mangiare alternativamente tutti i filosofi evitando che qualcuno in particolare soffra di **starvation**⁴⁵ ed evitando che si verifichino stalli in fase di "*acquisizione delle forchette, o delle bacchette*".

Il **problema del barbiere che dorme** o il **barbiere sonnolento** è enunciabile come: "Un barbiere possiede un negozio con una sola sedia da lavoro e un certo numero limitato di posti per attendere. Se non ci sono clienti il barbiere dorme altrimenti, all'arrivo del primo cliente il barbiere si sveglia ed inizia a servirlo. Se dovessero sopraggiungere clienti durante il periodo di attività del barbiere, essi si mettono in attesa sui posti disponibili. Al termine dei posti di attesa, un ulteriore cliente viene scartato."

Una corretta programmazione concorrente, che risolva il problema, deve far dormire il barbiere in assenza di clienti, attivare il barbiere sul primo cliente al suo arrivo e mettere in coda tutti i successivi clienti tenendoli inattivi.

Un problema reale che bene approssima questa situazione Questa problematica è il sistema di funzionamento degli helpdesk informatizzati dove l'operatore serve, uno per volta, tutti i clienti in coda oppure attende l'arrivo di nuove chiamate, senza dover fare alcun altro tipo di operazione.

I problemi notevoli qui descritti per sommi capi permettono di comprendere ulteriormente come le rete di Petri siano uno strumento concettuale raffinato e potente per la rappresentazione di processi dinamici non deterministici e complessi, presentando diversi campi di applicazione dal workflow management al Data Analysis e la programmazione concorrente, concurrent programming.

⁴⁵ **Starvation**, *inedia* in lingua inglese, nel campo dell'informatica identifica lo stato di un processo pronto per essere eseguito ma che non riesce ad ottenere le risorse di cui necessita.

Il WEB 2.0: filosofia e strumenti

Premessa

La nascita di **Internet** ha sicuramente dato forma e successivamente modificato e riempito di nuovi significati la comunicazione, la distribuzione e l'accesso all'informazione.

In particolare nel campo della comunicazione e della connessione tra le persone ha giocato un ruolo esiziale il **WEB**, così ormai chiamato con una familiarità che ne sottolinea la diffusione nel quotidiano di milioni di persone, uno tra i più importanti servizi **WWW** (**World Wide Web**, la ragnatela mondiale) di Internet dedicato alla navigazione attraverso **link**⁴⁶ e alla fruizioni di documenti multimediali, testi e immagini statiche e in movimento, così come di diversi servizi, come ad esempio motori di ricerca o prenotazioni di viaggi.

I link configurano il WEB come un **ipertesto**⁴⁷ e i servizi messi a disposizione del WEB lo hanno caratterizzato in funzione del fatto che gli utenti sono diventati anche possibili editori dei contenuti che il WEB rende accessibili.

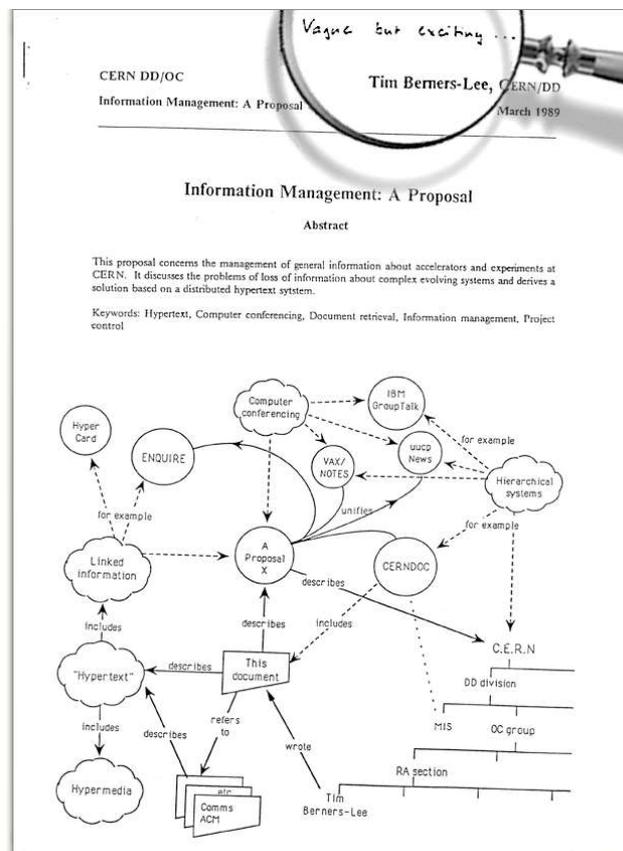


Figura 36 - Documento "Information Management" con la nota Mike Sendall (fonte sito CERN - <http://info.cern.ch/Proposal.html>)

La nascita ufficiale del WEB si data il 6 agosto 1991 quando Tim Berners-Lee⁴⁸ mise on-line su Internet il primo sito Web, sito accessibile alla comunità scientifica fino al 30 aprile 1993 quando il CERN ha reso pubblica la tecnologia su cui il WEB si basa.

⁴⁶ **Link**, collegamento, è l'abbreviazione del termine inglese **iperlink**, che in informatica indica un rinvio da una unità informativa ad un'altra; unità informative memorizzate ed ecdedibili con supporti digitali. Il concetto di link non è nato con il WEB ma è stato strutturato negli anni 60, quando si cominciavano ad avere dati su supporti magnetici e digitale.

⁴⁷ L'**ipertesto** è descrivibile come insieme di contenuti messi in relazione attraverso parole chiave.

⁴⁸ **Tim Berners-Lee** è il padre riconosciuto del Web; nato a Londra l'8 giugno del 1955 è un fisico, laureato al Queen's College alla Oxford University nel 1976; ha lavorato come ricercatore al CERN dove ha sviluppato la maggior parte dei progetti e dei software che sono la struttura portante del WEB.

Da un punto di vista della ricerca e dell'implementazione logica il WEB fu inizialmente implementato da Tim Berners-Lee e Robert Cailliau⁴⁹, entrambi ricercatori a CERN; il progetto base dell'invenzione dello strumento che è istanziato dal WEB, "*WorldWideWeb: Proposal for a HyperText Project*" fu presentato, nella sua versione definitiva, il 12 novembre del 1990, da Berners-Lee e Cailliau.

Lo stesso Berners-Lee aveva cominciato a lavorare negli anni 80 al software "*Enquire*" che diventerà la base concettuale del WEB e finalizzato alla creazione di un sistema documentale, che il CERN definiva "*un sistema ipermediale distribuito, ovvero un oggetto in grado di gestire in maniera organica documenti sotto forma di testo, suono ed altro localizzati in computer geograficamente distanti nel mondo*".

In questo cammino diventò una tappa importante anche il marzo 1989 quando Berners-Lee iniziò la stesura di un documento (vedi Figura 36) per ottenere finanziamenti, hardware e software per far realizzare un sistema ipertestuale in rete; sul documento di presentazione l'allora capo di Berners-Lee scrisse, Mike Sendall, "*Vague, but exciting...*" ("Vago, ma entusiasmante...") dimostrando tutto il potere innovativo che il progetto veicolava.

Quando dopo una serie di vicissitudini il progetto fu approvato, passando tra diversi titoli che ne evidenziassero lo scopo, cosicché dopo le proposte **Information Mesh** (*Rete/Maglia di Informazioni*), **Mine of Information** e **Information Mine** (*Miniera di informazioni e Informazione miniera*), venne scelto il più "evocativo": **World Wide Web**, la ragnatela globale, utilizzando NeXT⁵⁰, Berners-Lee creò le prime versioni dei programmi server e client, il browser, creando un'interfaccia in grado di leggere ed implementare ipertesti.

Iper testo, storia e definizioni

Una semplice ed efficace definizione di ipertesto è quella che si trova nella versione on-line della enciclopedia Treccani "**ipertesto**, in informatica, testo organizzato in un insieme di moduli elementari che ne rende possibile la lettura, integrale o parziale, secondo diversi percorsi logici (ciascuno dotato di autonomia di significato), scelti dal lettore in base a sue personali esigenze" (vedi Figura 37).

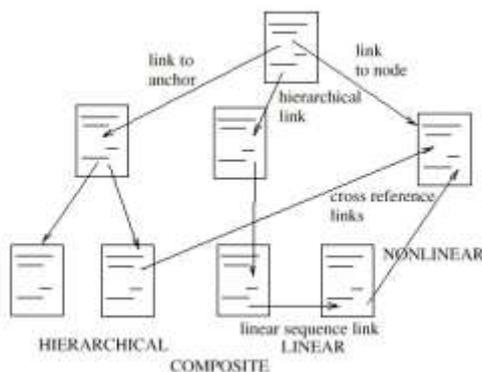


Figura 37- Schema rappresentativo di un ipertesto (fonte Treccani - <http://www.treccani.it/enciclopedia/ipertesto/>).

⁴⁹ **Robert Cailliau** è un informatico belga, nato a Tongeren il 26 gennaio 1947; laureato in ingegneria civile a Gand ha poi conseguito un master in Computer Science all'Università del Michigan. Entrato al CERN nel 1974 fino al 1987 anno in cui iniziò a lavorare alla divisione Elaborazioni Dati, diventando capogruppo dei Sistemi Informatici, si è occupato del sistema di controllo degli acceleratori di particelle. Dopo aver lavorato, al progetto WWW, con Tim Barners-Lee, nel 1994 è l'ideatore del progetto, in collaborazione con la Comunità Europea, "*Web for Schools*" per sviluppare la cultura e l'utilizzo del Web come risorsa educativa.

⁵⁰ **NeXTstation**, una workstation nel 1998, lanciata da Steve Jobs che uscito dalla Apple nel 1985 fonda la Next Computer. **NeXTstation** era una macchina dotata di un sistema operativo, **NeXTSTEP**, orientato agli oggetti, multitasking, con una interfaccia grafica e una potenza di calcolo che in quel periodo non aveva pari.

Si può quindi dire che la caratteristica dell'ipertesto è di poterlo "navigare" secondo logiche "non lineari", e il contenuto successivo non è determinato dalla sequenzialità dei testi o dei diversi media ma dalle parole chiave che determinano i link che gli utenti decidono di seguire. L'inizio della storia dell'ipertesto si può far risalire ad **Agostino Ramelli**⁵¹ che nella sua opera descrive una macchina la "ruota di libri" (vedi Figura 38), un leggio rotante che tramite meccanismi e strutture in legno è possibile consultare diversi libri, con una relazione logica sviluppata attraverso segnalibri meccanici e movimenti della ruota. Per questa sua similitudine con una navigazione tra documenti con strutture paragonabili a **link**, questa macchina è considerata come una prefigurazione dei sistemi ipertestuali.

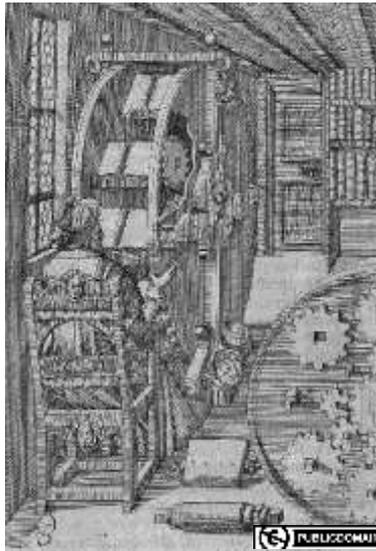


Figura 38 - il leggio rotante di Ramelli (Ignoto - de.wikipedia, uploaded as de:Bild:Ramellis Bücherrad.jpg by Asb)

Il concetto di ipertestualità trae anche la sua forza dal fatto che sia la mente umana che l'approccio delle persone al sapere e all'informazione, così come la ricerca di contenuti non è solo un processo sequenziale ma, la maggior parte delle volte, opera per associazione di idee, per parole chiave.

L'origine del termine **hypertext**, **ipertesto**, è attribuita a Ted Nelson⁵² che per primo lo usò e nel 1965 ebbe l'idea di un sistema software in grado di memorizzare documenti ed informazioni in diversi formati, e di permettere la costruzione di percorsi di fruizione differenziati, cioè di "navigare" in modo rapido ed efficace la rete informativa così costituita da documenti interconnessi.

I concetti di ipertesto e le intuizioni di Ted Nelson, hanno radici nelle teorie sviluppate già negli anni 1950, e nella prima applicazione **oN Line System (NLS)** ideata da Douglas

⁵¹ **Agostino Ramelli** (Ponte Tresa, 1531-1608) era un ingegnere italiano noto per l'opera "*Le diverse et artificiose machine del Capitano Agostino Ramelli Dal Ponte Della Tresia Ingegniero del Christianissimo Re di Francia et di Pollonia: nelle quali si contengono uarij et industriosi Mouimenti, degni digrandissima speculatione, per cauarne beneficio infinito in ogni sorte d' operatione*", che contiene la descrizione di 195 macchine diverse, tra cui la "**ruota dei libri**". L'opera di Ramelli ha avuto anche una pubblicazione moderna "*Le diverse et artificiose Macchine. 1588*" BSI - Banca Della Svizzera Italiana, Lugano 1992.

⁵² **Theodor Holm Nelson**, nato il 17 giugno 1937 è considerato uno dei pionieri dell'informatica ed è un sociologo e filosofo. Oltre al termine **ipertesto** coniato nel 1963, e usato per la prima volta in una pubblicazione del 1965, gli viene attribuita la paternità, tra gli altri, dei termini **ipermedia** e **virtualità**. La sua ricerca ha come focus l'accessibilità ai sistemi informatici per gli tutti gli utenti, ed in particolare quelli meno esperti. È significativa la sua affermazione "*Un'interfaccia utente dovrebbe essere così semplice da poter essere compresa, in caso di emergenza, nel giro di dieci secondi da un principiante*".

Engelbart⁵³ alla fine degli anni 1960. NLS aveva lo scopo di facilitare le attività che implicavano la soluzione cooperativa di problemi complessi, attraverso l'introduzione di diversi meccanismi per la condivisione delle informazioni. NLS è importante perché introduce i primi esempi di mouse e di interfaccia grafica con la struttura a finestre.

Tra il 1962 e il 1968, Engelbart, progetta e realizza in un laboratorio dello *Stanford Research Institute* **Augment**⁵⁴ il primo sistema ipertestuale funzionante e utilizzato fino al 1975.

Un altro punto di riferimento sia per i concetti di ipertestualità sono i concetti introdotti da **Vannevar Bush**⁵⁵, che ha ispirato il lavoro e le ricerche sia di Engelbart che di Nelson.

In particolare Bush in un articolo, considerato una pietra miliare in questo ambito speculativo, pubblicato del 1945 su *The Atlantic Monthly*, *As We May Think (Come potremmo pensare)*, metteva in evidenza le problematiche collegate al reperimento delle informazioni, fondamentale per ogni ricercatore. Nello stesso articolo supporta le sue teorie con il progetto di una macchina con la quale i ricercatori potessero raccogliere e organizzare i saperi e la documentazione utile per le loro ricerche; la macchina teorizzata nell'articolo viene chiamata **MEMEX (MEMory EXpansion)**.

Una frase di questo articolo è significativa per la grande intuizione che l'articolo stesso rappresenta *"Compariranno nuovi tipi di enciclopedie confezionate con una rete di percorsi associativi che le collegano, pronte ad essere inserite in memex e qui ampliate"*.

Il progetto prevedeva anche che ogni documento inserito in **memex** fosse associato ad uno spazio, identificato da un codice, per permettere diverse concatenazioni tra informazioni situate in anche in documenti diversi. *"Quando l'utente costruisce un collegamento gli assegna un nome, inserisce il nome nell'elenco dei codici, e lo compone sulla tastiera. Di fronte all'utente stanno i due elementi da collegare, proiettati su visori adiacenti. In fondo a ciascun elemento ci sono alcuni spazi bianchi per il codice, e un puntatore indica uno di questi spazi per ciascun elemento. [...] Quando numerosi elementi sono stati congiunti in questo modo a formare una pista, possono essere esaminati uno dopo l'altro [...] E' esattamente come se gli elementi fisici fossero stati raccolti separatamente e quindi rilegati assieme a formare un nuovo libro"* (Vannevar Bush, *"Come possiamo pensare"*, in Theodor Holm Nelson, *Literary Machines* 90.1. Il progetto Xanadu, Franco Muzzio Editore, 1992).

A partire dalle riflessioni, iniziata da Bush, sulla dicotomia tra la *logica sequenziale* del testo scritto e la *modalità associativa* del pensiero umano, ben rappresentato dalla struttura di rete/grafico, Nelson lavorò al progetto **Xanadu**⁵⁶, il primo software ipertestuale iniziato nel 1960 e di cui solo nel 1998 venne rilasciato il codice sorgente **Project Udanax**. Per questo, non stante l'importanza metodologica di questo progetto, la rivista americana di tecnologia *Wired* lo definì *"più lunga storia di vaporware⁵⁷ nella storia dell'industria informatica"*.

Con lo sviluppo che il software e l'hardware hanno avuto negli anni a partire dalla fine degli anni 80 sono stati implementati strumenti potenti, flessibili ed efficaci per l'implementazione di ipertesti.

⁵³ **Douglas Carl Engelbart** (Portland, 30 gennaio 1925 – Atherton, 2 luglio 2013) è stato un inventore noto, tra l'altro, per essere stato l'inventore con Bill English, del primo mouse ed essere stato un precursore degli studi dell'interazione uomo-macchina.

⁵⁴ Il progetto **Augment**, deve il suo nome, in italiano **aumento**, al fatto che aveva come concetto ispiratore il *valore aggiunto*, l'*aumento* appunto, che la tecnologia può dare all'uomo. Questo progetto è anche il contesto in cui Ted Nelson *"inventò"* il termine **hypertext**.

⁵⁵ **Vannevar Bush** (Everett, 11 marzo 1890 – Belmont, 30 giugno 1974) è stato uno scienziato e inventore che coordinò le attività di ricerca degli USA durante la seconda guerra mondiale; è considerato il precursore degli ipertesti, ed è stato l'ideologo del supporto delle attività di ricerca ai fini del potenziamento delle democrazie.

⁵⁶ Il nome Xanadu è un omaggio al poema *Kubla Khan* o "Visione in un sogno" di Samuel Taylor Coleridge. Xanadu è la città edificata da Kublai Khan dopo essere diventato imperatore della Cina.

⁵⁷ **Vaporware** è un neologismo inglese, con la radice **vapor** (*vapore*), utilizzato per definire ironicamente software o hardware continuamente annunciati in uscita a breve, ma di cui poi non si hanno più notizie, dileguandosi, appunto, come vapore.

Come software prototipale in questo campo, significativo sia per gli strumenti che ha messo a disposizione di programmatori ma anche di esperti di contenuto, si deve ricordare **Hypercard** (vedi Figura 39), un software progettato da Bill Atkinson, che Apple presenta nell'agosto 1987 e implementa fino al 1992 quando è stata distribuita l'ultima versione **Hypercard 2.4.1**; Hypercard è stato in vendita fino al 2004. Contestualmente ad Hypercard Apple distribuisce con System 6, la versione 6 del sistema operativo di Macintosh (*Mac OS*), che ancora non dava nomi a caratterizzare i propri sistemi operativi, iniziando dalla versione successiva System 7, noto come "Big Bang".

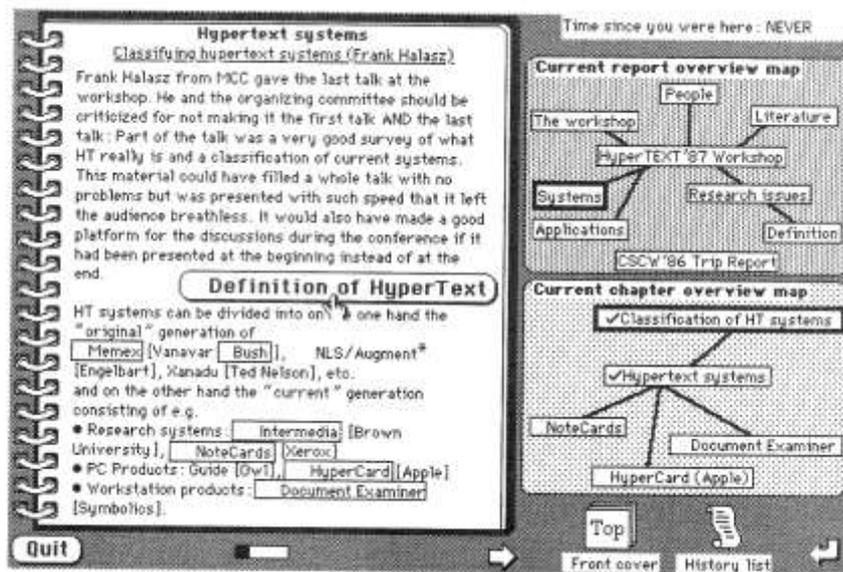


Figura 39 - Esempio di schermata di Hypercard 2 (fonte <http://trabalhohipermedia.do.sapo.pt/hypercard.htm>)

Un'analisi più dettagliata degli ipertesti da un punto di vista strutturale ha portato anche alla categorizzazione delle strutture ipertestuali tra le quali:

- *struttura assiale*
- *struttura gerarchica*
- *struttura di rete*

Un ipertesto a **struttura assiale** (vedi Figura 40) è un primo passo per la lettura non sequenziale del testo, ma di fatto è una rete, un grafo, composta da due tipologie di nodi: principali e secondari.

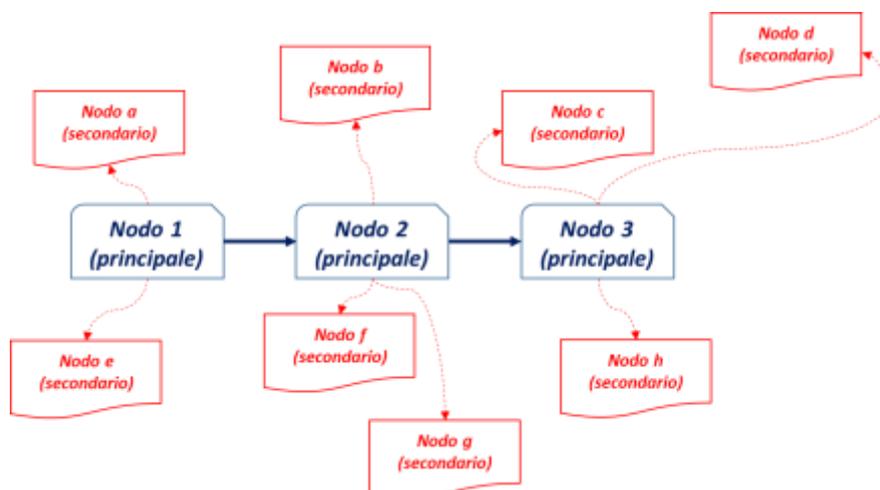


Figura 40 - Schema di Ipertesto a *struttura assiale*

Negli ipertesti a struttura assiale i nodi primari sono connessi tra loro in maniera sequenziale e sono, a loro volta, collegati a uno o più nodi secondari, concettualmente approfondimenti dei contenuti dei nodi principali.

Un ipertesto a **struttura gerarchica** (vedi Figura 41) si basa sul concetto di gerarchizzare i diversi nodi e rispetta le logiche della struttura ad albero; la struttura si basa sul concetto di nodo padre e nodo figlio, il contenuto di un nodo figlio è la continuazione del contenuto del suo nodo padre.

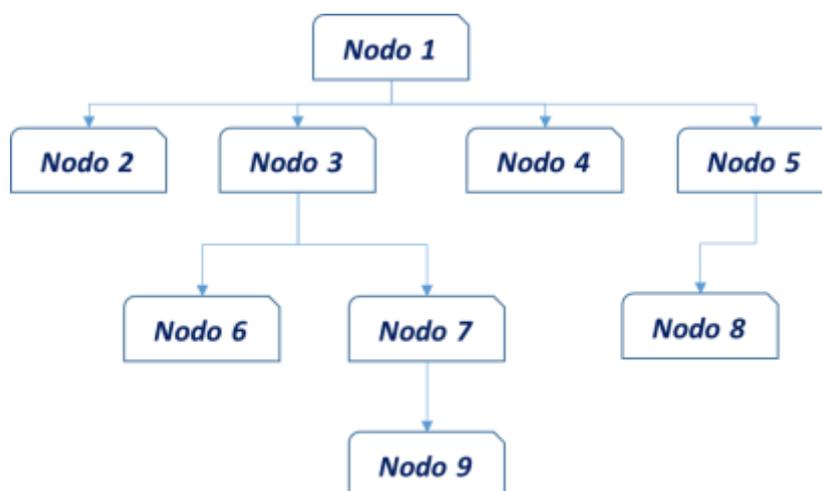


Figura 41 - Schema di Ipertesto a **struttura gerarchica**

L'ipertestualità e la scelta di percorsi diversi è garantita dal fatto che ogni nodo padre può avere più nodi figlio. Un ipertesto a **struttura di rete** (vedi Figura 42) è sicuramente quello che più di altri sfrutta completamente la potenzialità di un approccio per link associativi alla navigazione di contenuti.

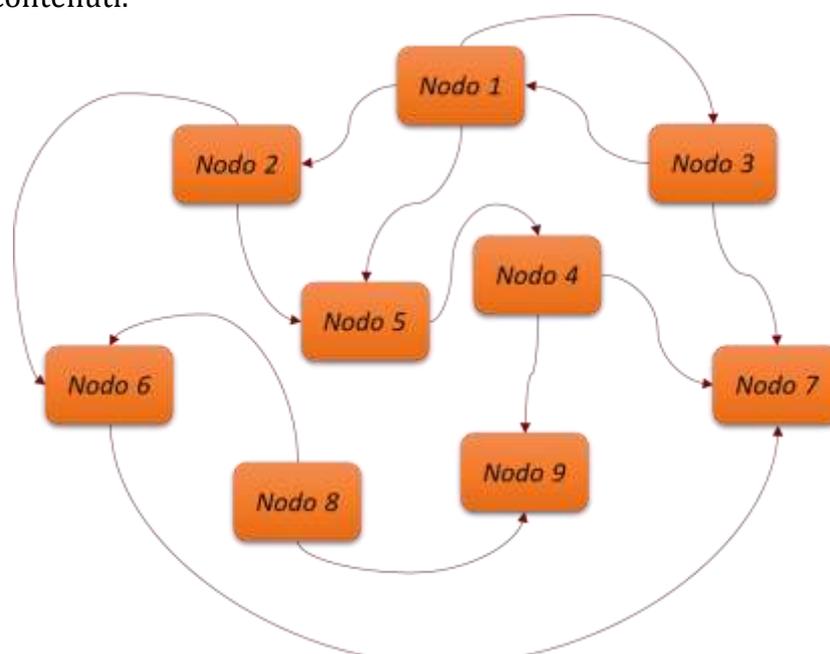


Figura 42 - Schema di Ipertesto a **struttura a rete o differenziale**

I nodi di un grafo a struttura di rete, o differenziale, sono considerati del medesimo livello e collegati in modo non sequenziale né gerarchico, è un modello che si rappresenta con un grafo orientato, in considerazione del fatto che i link mantengono un verso.

Il lettore in un ipertesto a struttura di rete può partire da un nodo e arrivare ad un altro senza l'obbligo di passaggio in nodi intermedi, scegliere cioè il proprio percorso di fruizione.

Per inquadrare questa struttura è interessante la definizione di ipertesto data da Roland Barthes⁵⁸: "In questo testo ideale le reti sono multiple e giocano tra loro senza che nessuna possa ricoprire le altre; questo testo è una galassia di significanti, non una struttura di significati; non ha inizio; è reversibile; vi si accede da più entrate di cui nessuna può essere decretata con certezza la principale; [...]"

Con lo sviluppo e la strutturazione di documenti contenenti diversi tipi di media che possono rappresentare link come le parole chiave si è introdotto il termine **ipermedia** a sottolineare il fatto che la filosofia di navigazione per associazione di contenuti non è strettamente connesso al testo; gli ipermedia sono implementati tipicamente per il WEB e Internet.

In questo contesto è interessante sottolineare la relazione tra ipermedialità e processi educativi; tra le metodologie nate a partire da ipertesti, ipermedia e le mappe concettuali è interessante il caso della **flipped classroom**⁵⁹.

Evoluzione del WEB: verso il Web Semantico

Così come ogni applicazione software o metodologia di comunicazione anche il WEB ha avuto una sua evoluzione, evoluzione logica e strutturale scandita dalla versione, principalmente si parla di **WEB 1.0**, **WEB 1.5** e **WEB 2.0** e ci sono anche definizioni di **WEB 3.0**.

Negli anni queste definizioni si sono intrecciate e sovrapposte, in questo contesto vedremo da una parte lo sviluppo temporale ma parallelizzandoli con le diverse visioni filosofiche e strutturali che vengono attribuite ai diversi WEB.



Figura 43 - Evoluzione del WEB da un punto di vista della comunicazione

A premessa dell'analisi della storia evolutiva, per sommi capi del WEB, va specificato che nonostante l'uso come sinonimi il WEB non è Internet; per Internet si intende una rete globale di computer e devices elettronici interconnessi su cui vengono condivisi diversi servizi e sui cui "girano" diversi protocolli di comunicazione.

⁵⁸ Roland Barthes (Cherbourg, 12 novembre 1915 – Parigi, 26 marzo 1980) è stato uno scrittore, critico letterario e semiologo francese, uno dei principali esponenti della critica francese di orientamento strutturalista. Tra le sue opere principali c'è il saggio *Il grado zero della scrittura* del 1953.

⁵⁹ **Flipped classroom**, *classe capovolta* o *classe ribaltata*, indica una forma di insegnamento-apprendimento che ha lo scopo di far lavorare lo studente prevalentemente a casa attraverso video e multimedia, o leggendo i testi ipertestuali proposti dagli insegnanti o condivisi da altri docenti. I primi esperimenti di classroom a cui si fa riferimento sono quelli condotti negli anni novanta da **Eric Mazur**, professore di fisica presso l'Università di Harvard.

In particolare Internet è una rete che rende possibile agli utenti la ricerca di contenuti la fruizione di servizi, Internet è in grado di elaborare informazioni e servizi usando diversi serie di linguaggi che permettono la trasposizione di informazioni tra computer.

Tra questi linguaggi-protocolli, per esempio, ci sono **IMAP**⁶⁰, **POP3**⁶¹ e **SMTP**⁶² che sono utilizzati dai servizi di mail; i protocolli che vengono utilizzati principalmente dal WEB sono **HTTP**⁶³, **URL**⁶⁴ e **HTML**⁶⁵.

L'evoluzione del WEB è schematizzabile come una evoluzione da una comunicazione monodirezionale ad una bidirezionale (vedi Figura 43), con la possibilità di connessione e comunicazione trasversale tra i diversi attori del processo comunicativo sul WEB.

La prima versione del WEB, universalmente riconosciuta come invenzione di Berners-Lee, è definita **WEB 1.0**; la prima versione del WEB è definito come il "*web dei contenuti*" laddove i siti erano semplici ipertesti, documenti da sfogliare come un libro "non sequenziale"; a questo si deve il nome di **browser**⁶⁶ per i software di navigazione in rete.

I contenuti di questo primo WEB erano sia in formato testo che immagini e video ma oltre ai link predefiniti non c'erano interazioni tra utenti e contenuti; il flusso informativo era monodirezionale, con un **architettura** strettamente **client-server**⁶⁷.

IL crescente utilizzo del WEB evidenziò come un problema l'impossibilità dell'utente di interagire con i contenuti e questo spinse i ricercatori a sviluppare un sistema che permettesse tali interazioni.

I primi risultati di questa ricerca sono stati l'utilizzo di data base e la programmazione con linguaggi per il WEB, come **PHP**⁶⁸ e i **linguaggi di scripting**⁶⁹, che permettevano la creazione

⁶⁰ **SMTP** (Simple Mail Transfer Protocol - *Protocollo Semplice di Trasferimento della Posta*) è il protocollo standard utilizzato per il trasferimento di posta tra server. Il protocollo SMTP funziona in modalità connessa, e la posta è consegnata direttamente al server di posta del destinatario.

⁶¹ **POP** (Post Office Protocol - *Protocollo dell'Ufficio Postale*) permette di recuperare la propria posta su un server remoto, detto server POP. E' utile per le persone che, non essendo connesse in permanenza ad internet, per consultare le proprie mail off-line. Il protocollo funziona con i comandi inviati al server POP da parte del client; ogni comando, concluso con la sequenza **CR/LF** - Carriage Return / Line Feed (equivalenti ad un **INVIO**), è composto da una parola-chiave con eventuali argomenti ed è seguito da una risposta del server di posta composta da un numero e da un messaggio descrittivo. Esistono anche i protocolli POP2 e POP3 con comandi e caratteristiche diverse.

⁶² **IMAP** (Internet Message Access Protocol - *Protocollo di Accesso a Messaggi in Internet*) è un protocollo alternativo al protocollo POP3 con maggiori potenzialità ed in particolare permette la gestione di multi accessi, la gestione di diverse caselle postali e di smistare la posta secondo criteri multipli.

⁶³ **HTTP** (Hypertext Transfer Protocol - *Protocollo di Trasferimento di Ipertesti*), protocollo di rete che permette la trasmissione delle informazioni, in particolare per le applicazioni a struttura client-server.

⁶⁴ **URL** (Uniform Resource Locator - *Localizzatore Unico di Risorse*), è uno schema per l'identificazione univoca di contenuti e servizi del WEB. L'indirizzo, definito come URL, identifica ogni risorsa in rete in maniera univoca e può essere definito come **<protocol>://<node>/<location>**, e più in generale ogni URL si compone normalmente di sei parti:

protocollo://<username:password>@nomehost<:porta></percorso><?querystring><#fragment identifier>.

⁶⁵ **HTML** (HyperText Markup Language - *Linguaggio Marcatura Ipertestuale*), linguaggio di markup con cui sono normalmente scritte le pagine WEB, usato per la formattazione dei documenti ipertestuali per la rete.

⁶⁶ Il **web browser**, *navigatore di rete*, dove il nome **browse**, **scorsura**, è un sostantivo derivante dal verbo *to browse* che significa, tra l'altro sfogliare, scorrere o curiosare. Il primo browser o perlomeno il primo ad essere conosciuto a livello internazionale è stato **Mosaic**, sviluppato da **NCSA** (**National Center for Supercomputing Applications** presso la Università dell'Illinois a Urbana-Champaign).

⁶⁷ L'**architettura client-server** è un'architettura di rete che permette ad un computer client di connettersi ad un server per una semplice fruizione di un servizio o l'accesso ad un contenuto.

⁶⁸ **PHP**, che è un acronimo ricorsivo (**PHP: Hypertext Preprocessor**, *preprocessore di ipertesti*), originariamente era acronimo di **Personal Home Page**, è un linguaggio interpretato, progettato la programmazione di pagine dinamiche sul WEB.

⁶⁹ Un **linguaggio di scripting** è un linguaggio di programmazione interpretato creato con lo scopo di implementare programmi di automazione dei sistema operativo, applicazioni o generare pagine dinamiche sul WEB.

di pagine dinamiche e l'inserimento e la condivisione di commenti ai contenuti on-line; queste prime modifiche strutturali caratterizzano quello che è chiamato **WEB 1.5**.

Con **WEB 2.0** si intende evidenziare una ulteriore evoluzione del WEB, un *WEB dinamico*, facendo riferimento in particolare a tutte le applicazioni che implementano e permettono un alto livello di interattività tra gli utenti e i siti, ma anche tra gli utenti e i siti tra loro; le applicazioni a cui si fa riferimento, tra le altre, sono **blog, forum e wiki**, così come piattaforme di condivisione di media come **YouTube o Flickr**, fino a comprendere i social network come **Facebook, Google+ o LinkedIn**.

Non tutti sono concordi sull'innovazione che è rappresentata dal WEB 2.0 e lo stesso Berners-Lee, pur concordando sulla natura **user-centred** (centrato sull'utente) del 2.0, ha affermato "2.0 [...] nessun vero significato: ognuno in funzione delle tecnologie che vuole proporre ne determina uno".

Il termine WEB 2.0 viene associato a **Tim O'Reilly**⁷⁰ che dal 2004 organizzò il primo **Web 2.0 Summit**, inizialmente noto come **Web 2.0 Conference**, che si è tenuto fino al 2011 a San Francisco ed era un incontro in cui si discuteva e ci si confrontava sul **WWW (World Wide Web)**. L'uso di **2.0** è ispirato dal mondo degli sviluppi software, dove la notazione puntata indica l'indice di sviluppo e successivo rilascio di un software.

Un'analisi del WEB 2.0 evidenzia, a partire dalle applicazioni che ospita, alcune caratteristiche che si possono dividere in ambito tecnologico, linguaggi di programmazione e applicazioni WEB e in ambito psico-sociale sul versante delle interazioni e dello sviluppo delle reti sociali e di partecipazione allo sviluppo dei contenuti in rete.

In particolare per ciascuno dei due aspetti è possibile affermare:

- **livello tecnologico:** il browser è sempre il software di navigazione della rete con la sua specifica interfaccia grafica, che genera uno standard interattivo per le diverse applicazioni, ma si evolvono i linguaggi per la descrizione dei contenuti; significativo l'introduzione e l'uso di **XML**⁷¹ e tecnologie **AJAX**⁷²;
- **livello psico-sociale:** l'interazione genera un nuovo modello di rete in cui gli utenti non solo fruiscono ma generano contenuti, diventano **prosumer**⁷³, e si evolvono le relazioni sviluppando il concetto di **community** e di **social network**.

Oltre le diverse posizioni assunte da chi il WEB ha inventato e di tutti coloro che ne hanno promosso lo sviluppo come strumento sociale e tecnologico, si può affermare che la direzione di sviluppo porti verso un WEB che diventa ciò che voleva essere: **WEB semantico**.

WEB semantico è un termine, che si deve a Berners-Lee, e indica che l'obiettivo è trasformare il WWW in un ambiente nel quale tutti i documenti pubblicati, qualsiasi sia il loro formato e la

⁷⁰ **Tim O'Reilly**, nato a Cork in Irlanda il 6 giugno 1954, è un editore, fondatore della O'Reilly Media (*ex O'Reilly & Associates*) ed è un convinto assertore della filosofia del software libero e dei movimenti open source.

⁷¹ **XML (eXtensible Markup Language, linguaggio marcatore ampliable):** linguaggio marcatore con un sistema di leggi sintattiche per definire e controllare il significato degli elementi contenuti in un documento XML rappresenta un metalinguaggio per la generazione anche di nuovi linguaggi per il WEB, una versione semplificata ed efficace di **SGML (Standard Generalized Markup Language)**. XML è un progetto del **W3C (World Wide Web Consortium)** le cui specifiche divennero raccomandazione. Nel febbraio del 1998 le specifiche divennero una raccomandazione, *Extensible Mark-up Language, versione 1.0*.

⁷² **AJAX (Asynchronous JavaScript and XML):** tecnologia asincrona, i dati vengono caricati in background senza interferire con la pagina Web, per lo sviluppo software dedicata alla realizzazione di applicazioni web interattive, quello che viene definito **Rich Internet Application**, le applicazioni Internet arricchite. AJAX non è propriamente una tecnologia ma un gruppo di tecnologie utilizzate in relazione reciproca.

⁷³ **Prosumer** è una *portmanteau word*, letteralmente una parola valigia o parola macedonia, formata dal mix delle parole **professional** o **producer**, con la parola **consumer**, che nel mondo WEB identifica il fenomeno per cui l'utente abbandona classico ruolo passivo e assume un ruolo nella creazione, produzione e distribuzione

loro codifica di condivisione, sono associati a **metadati**⁷⁴ che permettono di definirne il contesto **semantico**⁷⁵, in modo tale da strutturare un ambiente adatto all'interrogazione e l'interpretazione, soprattutto da parte dei motori di ricerca e all'elaborazione automatica delle informazioni.

L'evoluzione dal WEB al WEB Semantico è scandito dalla definizione, da parte del W3C, dello standard **RDF**⁷⁶ (**R**esource **D**escription **F**ramework) che standardizza la definizione di relazioni tra informazioni ispirandosi ai principi della logica dei predicati, logica predicativa del primo ordine, facendo, tra l'altro ricorso a strumenti tipici del WEB, come ad esempio gli **URI**⁷⁷ (**U**niform **R**esource **I**dentifier).

Un riassunto significativo di cosa rappresenti il WEB semantico è un definizione comparsa sulla rivista Scientific American nel maggio 2001: "*Il Web Semantico è un'estensione del Web corrente in cui le informazioni hanno un ben preciso significato e in cui computer e utenti lavorano in cooperazione*".

WEB 2.0: strumenti e filosofia di utilizzo

Il WEB 2.0 si caratterizza per essere un approccio filosofico alla rete, ancor prima che tecnologico, un approccio con una dimensione sociale. Dal punto di vista tecnologico gli strumenti della rete già presenti nel WEB 1.0, come i forum o i blog e le stesse chat, possano apparire invariati ma è la modalità di utilizzo della rete che, basandosi sulla potenzialità che ogni utente ha di essere fruitore, produttore e gestore dei contenuti on-line, ad aprire nuovi scenari.

La descrizione WEB 2.0 prende sempre le mosse da un confronto con WEB 1.0 ma accentrando l'attenzione non sull'infrastruttura tecnica o la struttura tecnologica ma piuttosto sulle logiche interattive e sui nuovi paradigmi comunicativi; la vera evoluzione è l'importanza che hanno assunto i concetti di condivisione, collaborazione e usabilità, il tutto sostenuto dalla considerazione che pur restando l'ipertesto la struttura del WEB 2.0 si intende evolvere in struttura semantica dei collegamenti e dei documenti ipertestuali.

Prima di procedere nell'analisi delle parole chiave e negli strumenti caratterizzanti il WEB 2.0 è interessante riassumere che le differenze tra WEB 1.0 e 2.0 sintetizzano il passaggio da un WEB monodirezionale ad un WEB bidirezionale. Come premessa è interessante analizzare la tabella generata dalle prime riflessioni di *Tim O'Reilly*:

WEB 1.0	WEB 2.0
Siti	Blogging
Gestione dei contenuti	Wiki
Stickiness (<i>rigidezza</i>)	Syndication ⁷⁸

⁷⁴ **Metadato** (dal greco *μετά*, oltre, dopo e dal latino *datum*, informazione), letteralmente significa "dato oltre un altro dato", è un'informazione che descrive un insieme di dati. Un esempio tipico di metadati è costituito dalla scheda del catalogo di una biblioteca,

⁷⁵ Con **semantica** si indica la parte della linguistica che studia il significato delle parole, delle frasi e dei testi.

⁷⁶ Il **Resource Description Framework** è lo strumento per lo scambio e il riutilizzo di metadati strutturati e consente l'interoperabilità tra applicazioni che condividono le informazioni sul Web. Componenti dell'RDF sono **RDF Model and Syntax**, che descrive la struttura del modello una possibile sintassi e **RDF Schema**, che espone la sintassi per definire schemi e vocabolari per i metadati.

⁷⁷ La locuzione **Uniform Resource Identifier** definisce una stringa che identifica in maniera univoca, e non ambigua, una risorsa generica on-line.

⁷⁸ **Syndication** è il termine inglese che identifica il fenomeno che si crea quando un giornalista vendere il proprio pezzo, tramite una agenzia specializzata, a più giornali o riviste. Nell'accezione **Web Syndication** descrive la diffusione di contenuti a più utenti tramite i canali e gli strumenti WEB.

Per specificare ulteriormente le differenze tra 1.0 e 2.0 in relazione alla categorizzazione semantica dei contenuti (vedi Figura 44) si può dire che il WEB 2.1 cataloga con il concetto di **directory** mentre il WEB 2.0 introduce il **tagging**⁷⁹.

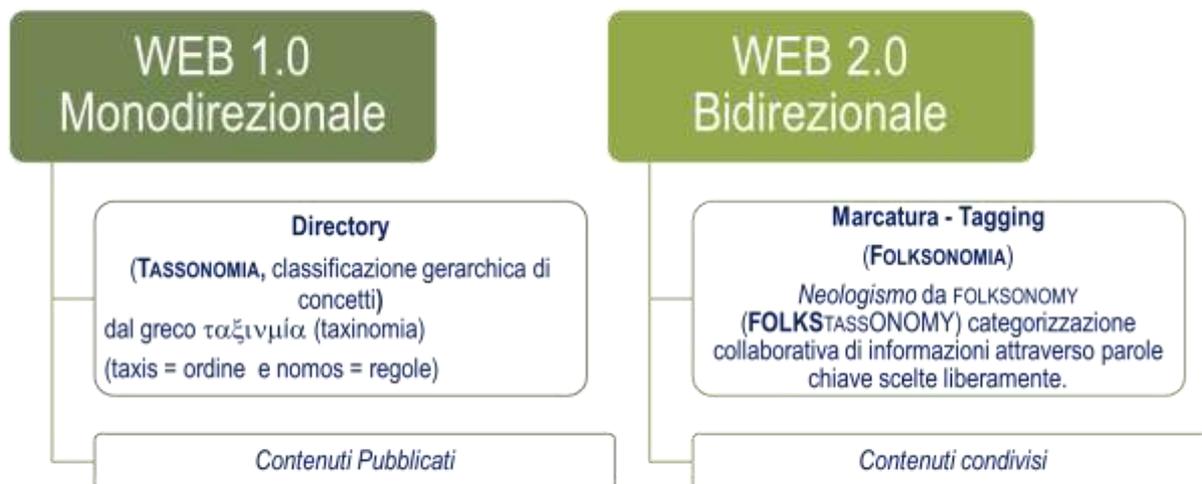


Figura 44 – Catalogazione dei contenuti dal Web 1.0 al WEB 2.0

Questo porta al capovolgimento di prospettiva laddove nel WEB 1.0 i contenuti sono pubblicati e l'utente è solo lettore, mentre nel WEB 2.0 i contenuti sono condivisi e l'utente diventa lettore ed editore.

I concetti caratterizzanti del WEB 2.0 sono ben rappresentati dalla **nube semantica**, costruita nel 2005 dal blogger **Markus Angermeier** (vedi Figura 45), quella che si può chiamare una **buzzword cloud**, nube di parole di moda.



Figura 45 - Nube semantica del WEB 2.0 di Markus Angermeier (The huge cloud lens bubble map web2.0 2005-11-11)

In questa mappa attorno al concetto di WEB 2.0 sono evidenziati i 6 concetti fondanti la filosofia del 2.0 a cui sono associati strumenti che ne rendono possibile la realizzazione e successivamente tutti i software che a tali filosofie si ispirano.

⁷⁹ Il termine **tagging** è un termine che deriva dal termine inglese **tag** e significa attribuire una parola o più parole chiave, il **tag**, che mira a definire semanticamente il documento oggetto della marcatura.

I 6 concetti, parole chiave del WEB 2.0 sono dunque:

- **Partecipazione** (*partecipazione*)
- **Convergence** (*convergenza*)
- **Remixability** (*possibilità di ricombinare i contenuti*)
- **Standardization** (*standardizzazione*)
- **Design** (*progettazione*)
- **Economy** (*economicità*)

Tra questi concetti si prenderanno in considerazione quelli che nel contesto di analisi che è stato definito hanno una maggior rilevanza: **partecipazione** (*participation*) e **ricombinazione** (*remixability*).

La **partecipazione** è sicuramente la nota caratterizzante il WEB 2.0, concetto che immediatamente lo inquadra come uno strumento di condivisione e compartecipazione di documenti e informazioni; questo non solo ha generato strumenti ma anche modelli di sviluppo come i progetti *Open Source*.

Se si considerano quali sono gli “strumenti logici” che sono emanazione e applicazione della partecipazione così come sono rappresentati anche nella nube semantica di Angermeier troviamo: **Wikis, Folksonomy, Recommendation, Social Software e Blogs**, ciascuno dei quali ha poi una serie di software che ne rappresentano una applicazione web.

Tra questi “strumenti logici” si vedranno più in particolare *Wikis, Blog e Folksonomy* come prototipali della filosofia di partecipazione e condivisione.

Il **blog** deve il suo nome alla contrazione della locuzione **web log**, cioè un diario di rete; in particolare la scelta di log, come giornale o diario riprende il termine *logbook*, il diario o giornale di bordo delle navi, termine che dal 1963 è utilizzato anche in informatica **log** e **login**, con i *file di log* che tengono memoria delle cronologie di accesso ad un sito o ad un servizio.

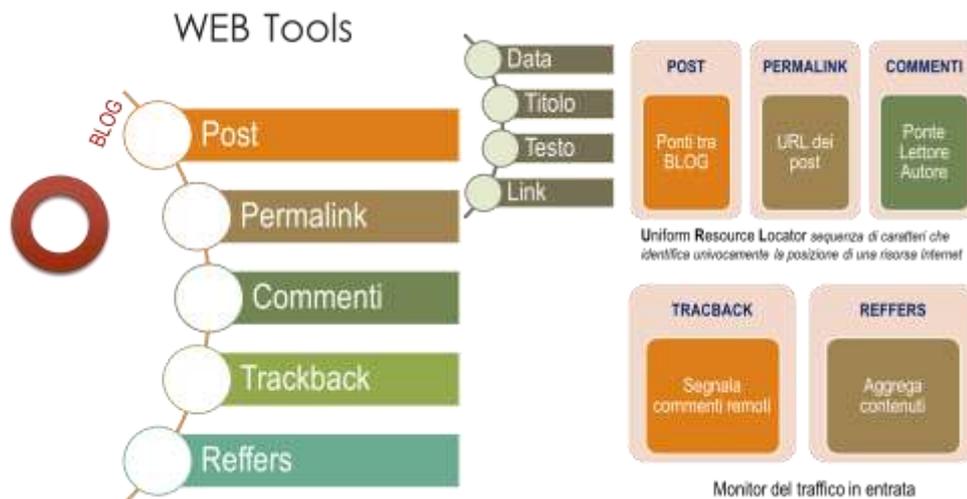


Figura 46 - Rappresentazione della struttura di un Blog

La nascita del blog è da attribuire a **Jorn Barger**, che il 23 dicembre del 1997 apre la sua pagina personale *RobotWisdom* per condividere i risultati delle sue ricerche sul WEB relative al suo hobby, la caccia, diventando uno dei più influenti blogger americani; è comunque il 18 luglio 1997 che viene accreditata come nascita del blog, quando venne sviluppato, da parte di **Dave Winer**⁸⁰, il software per pubblicarli.

⁸⁰ Dave Winer, nato il 2 maggio 1955, a Brookling, New York, è uno sviluppatore software che ha sviluppato i software di pubblicazione del primo blog e del sistema di *feed* RSS (**RDF Site Summary** o **Really Simple Syndication** o **Rich Site Summary**) è uno dei formati più diffusi per la distribuzione di contenuti Web

L'abbreviazione definitiva blog è proposta nel 1999 da **Peter Merholz** sul suo sito *Peterme.com* a cui è seguito anche la creazione del verbo "*we blog*"; in Italia il blog prende piede a partire dal 2001 ed è un fenomeno che si è diffuso in modo tale da far parlare di blogosfera ad indicare l'insieme dei blog, insieme che è di non complessa gestione perché tutti i blog hanno la stessa struttura (vedi Figura 46) facilitando l'utilizzo da parte degli utenti ma anche la reciproca ricerca ed interconnessione.

Le parti strutturali di un blog sono:

- **post**: è il commento, l'unità logica che compone il blog ed è caratterizzato da una *data* di pubblicazione, un *titolo*, un *contenuto* ed *eventualmente un link*; il post ha un URL che è univoco e non viene riutilizzato nemmeno se il post viene cancellato;
- **permalink**: è un link che rimanda in modo permanente ad un determinato post;
- **commenti**: rappresentano il mezzo di dialogo tra gli utenti e il responsabile del blog; gli utenti non possono aprire discussioni ma solo partecipare a quelle definite dal blogger;
- **trackback**: è un sistema che permette di avere segnalazioni se avvengono commenti remoti, cioè richiami ai commenti da parte di altri blog; in generale i blog controllano il traffico in entrata;
- **reffer**: è il sistema che permette l'aggregazione e l'eventuale successiva ricerca dei contenuti del blog.

La definizione della blogosfera e l'aumento di interazione con essa condiziona l'importanza dei contenuti e la loro credibilità in rete, generando quella che decide quindi cosa ha importanza e cosa no, creando quello che **James Suriowecki**, giornalista americano del The New Yorker, chiama "*saggezza delle folle*", cioè ogni individuo aggiunge e definisce il valore di un contenuto.

Il **wiki**, sistema collaborativo, il cui esempio più noto e significativo è **Wikipedia**, è uno dei primi costrutti logici che implementa lo sviluppo del WEB 2.0 ed è un sistema che:

- rende semplice immissione, modifica e condivisione di contenuti;
- non è invasivo, cioè non necessita di installazioni di software e web-based, che lo rende indipendente sia dai device che dalle piattaforme;
- incoraggio e implementa il concetto di community.

Lo sviluppo delle community attorno ai wiki ha fatto sì che ci siano anche delle catalogazioni degli utenti, tra cui si possono elencare:

- **WikiGnome** (*gnome, gnome*): un utente utile e collaborativo, contribuisce alle voci con commenti e modifiche;
- **WikiFairy** (*fairy, fata*): un utente collaborativo che si dedica particolarmente all'abbellimento delle pagine;
- **WikiGremlin** (*gremlin, spiriti scherzosi*): un utente che produce danni con le proprie modifiche.

Un wiki è sostanzialmente un sito web implementato da un software collaborativo detto *software wiki* o semplicemente *Wiki*; i link, nel wiki, sono definiti in **CamelCase**, una modalità di scrittura che prevede una alternanza di lettere maiuscole e minuscole che costituisce un codice per identificare i link ipertestuali e definire a quale ambito semantico la parola appartiene. Gli utenti del sito possono aggiungere, modificare o cancellare contenuti, utilizzando un linguaggio di markup semplificato, si tratta quindi di un insieme ipertestuale di documenti.

L'inventore del wiki è Ward Cunningham che per il nome si ispirò al termine **wiki** termine che in lingua hawaiana significa "rapido" oppure "molto veloce" e che Cunningham trasse dal nome dei bus navetta dell'aeroporto di Honolulu, i *wiki wiki*.

Il software wiki è nato nel **design pattern**⁸¹ della comunità web come una soluzione per scrivere e discutere **pattern languages**, intendendo "linguaggi relativi a un preciso contesto", per i software di programmazione; il *Portland Pattern Repository* è stato il primo wiki, ideato da **Ward Cunningham**⁸² nel 1995.

I metodi di classificazione utilizzati per i documenti in rete caratterizzano fortemente il concetto di partecipazione e libertà di espressione. Nel caso di WEB 1.0 si sono utilizzate le tassonomie, classificazioni gerarchiche, mentre nel WEB 2.0 si parla di **folksonomie**.

Il termine **folksonomia** è un neologismo, in inglese *folksonomy*, che identifica il processo di categorizzare delle informazioni fatta dagli utenti mediante con l'utilizzo di tag scelti in modo libero. Il termine è nato dall'unione di **folk**, popolo o popolare, e **tassonomia**, evincendo così che una folksonomia è un processo tassonomico fondato su criteri individuali.

Si può quindi dire che una folksonomia è la metodologia utilizzata da gruppi di persone che collaborano spontaneamente per organizzare in categorie le informazioni disponibili sul WEB. Le folksonomie possono essere catalogate secondo due linee principali:

- **narrow folksonomy**: utilizza **tag** che si ritengono più adatti, e gli utenti si limitano a navigare tra i tag;
- **broad folksonomy**: utilizzata soprattutto nelle reti sociali, ogni utente assegna ad ogni risorsa **tag** diversi che ritiene utili per la classificazione.

Come conclusione di un'analisi, seppur per sommi capi, del WEB 2.0 e del concetto di WEB semantico che ne deriva, con la relativa visione user-centred, è interessante riportare la caratterizzazione fatta da **Andrew Lih**⁸³ e nota come le **4C** che descrivono il *WEB semantico*:

- Creativity (*creatività*)
- Client-side
- Collective intelligence (*intelligenza collettiva*)
- Copyright e Copyleft

La **creatività** non è da intendersi in maniera letterale; piuttosto si riferisce al fatto che gli utenti sono creatori, cioè possono inserire nuovi contenuti.

L'essere *user-centred*, e cioè dipendente dall'utente, **client-side**, ci sono infinite possibilità di combinazione dei documenti e dei media perché ognuno può connettersi con tutti gli altri per scambiare e combinare informazioni.

Il concetto di **colletive intelligence**, *intelligenza collettiva*, si può intendere secondo la definizione di **Pierre Levy**⁸⁴: "*Che cos'è l'intelligenza collettiva? In primo luogo bisogna riconoscere che l'intelligenza è distribuita dovunque c'è umanità, e che questa intelligenza, distribuita dappertutto, può essere valorizzata al massimo mediante le nuove tecniche, soprattutto mettendola in sinergia. Oggi, se due persone distanti fanno due cose complementari, per il tramite delle nuove tecnologie, possono davvero entrare in comunicazione l'una con l'altra, scambiare il loro sapere, cooperare. Detto in modo assai generale, per grandi linee, è questa in fondo l'intelligenza collettiva*".

⁸¹ Schema progettuale, architettuale e di progettazione.

⁸² **Ward Cunningham**, nato il 26 maggio 1949 è un programmatore americano, inventore del concetto di *wiki*. Cunningham ha fondato il primo sito wiki, il **Portland Pattern Repository**, il 25 marzo 1995. Il sito, che è ancora attivo, è dedicato a "persone, progetti e percorsi" ed è una "storia informale delle idee di programmazione".

⁸³ **Andrew Lih** è un ricercatore dei nuovi media, consulente e scrittore statunitense.

⁸⁴ **Pierre Lévy**, nato a Tunisi nel 1956, è un filosofo francese che studia l'impatto di Internet sulla società e di come Internet è uno strumento per aumentare le capacità di cooperazione.

Questa attenzione collettiva della blogosfera decide quindi cosa ha importanza e cosa no, creando, di fatto, una “saggezza delle folle” (James Suriowecki) dove ogni individuo aggiunge e/o seleziona il valore.

Indice delle Figure

Figura 1 - Mappa di Königsberg ai tempi di Eulero, in evidenza la disposizione dei ponti sul fiume Pregel e la disposizione dei diversi quartieri della città (immagine tratta da Wikipedia – CC BY-SA 3.0).....	5
Figura 2 – Posizione geografica di Kalinirad (immagine trattata dal sito www.aebr.eu)	6
Figura 3 - Rappresentazione del problema attraverso l'astrazione con enti geometrici della mappa reale della città (i nodi del grafo) e della disposizione dei ponti (gli archi del grafo).....	7
Figura 4 - Schema e definizioni di un arco orientato	9
Figura 5- Grafo semplice orientato (A) e grafo semplice non orientato (B)	9
Figura 6 - Multigrafo orientato (A) e multigrafo non orientato (B)	9
Figura 7 – Esempi di sottografo e grafo parziale	10
Figura 8 - Grafi e connessione.....	11
Figura 9- Icosian game o Puzzle di Hamilton, gioco inventato nel 1857	12
Figura 10 -Esempi di grafi euleriani e hamiltoniani	12
Figura 11 - Esempio di grafo planare (A) e grafo non planare (B)	13
Figura 12 - - Esempio di grafo bipartito (A) e grafo bipartito completo (B)	13
Figura 13 - Esempio di albero radicato (nodo radice R, livello 0) di profondità 3	14
Figura 14 - Esempi di memorizzazione di un grafo NON orientato	15
Figura 15 - Grafo schema del problema dello “scambio dei cavalli” e immagine della relativa scacchiera di riferimento	16
Figura 16 - Grafo dello schema di risoluzione del problema “salvare capra e cavolo”; in questo schema i nodi contengono le configurazioni presenti sulla riva di partenza (Riva A) e le relazioni (gli archi) rappresentano gli stati che si ottengono dopo un passaggio.....	18
Figura 17 - Mappa degli Stati Uniti colorata secondo i principi del problema dei quattro colori	20
Figura 18- Rappresentazione che associa ad ogni regione della mappa un nodo del grafo	20
Figura 19 - Linea temporale dello sviluppo della dimostrazione del teorema dei quattro colori.....	21
Figura 20 – Schema grafico dei passi per la soluzione del “problema del commesso viaggiatore” utilizzando che usando l'algoritmo euristico “Nearest Neighbour Algorithm”	22
Figura 21 - Rappresentazione del problema “tre case a tre fornitori”.....	23
Figura 22- (A): esempio di grafo $K_{3,3}$, grafo bipartito completo 3+3 nodi; (B): esempio di grafo K_7 , grafo completo con 7 nodi.....	23
Figura 23- Rappresentativo di una struttura posto-transizione-posto i una P-rete (A); nodi posto che contengono token (B).....	26
Figura 24 - Schema di transizione attiva	27
Figura 25- Schema semplificato di sequenza di 2 transizioni t_i e t_j	28
Figura 26 - Esempi di concorrenza strutturale (A) e concorrenza effettiva (B)	28
Figura 27 - Esempi di conflitto (A) e confusione (B).....	29
Figura 28 - Grafo di raggiungibilità (B) relativo a una rete di Petri descritta in termini di transizione (A)	31
Figura 29 - Esempio di metodo dei posti complementari con $k=4$	32
Figura 30 - Esempio di schema di Macchina a Stati Finiti (fonte iralab - URL: http://irawiki.disco.unimib.it/irawiki/index.php/INFIND2011/12_Rover_Robot)	33
Figura 31- Esempio di Macchina di Turing (fonte RAI Cultura Filosofia - URL: http://www.filosofia.rai.it/articoli/la-macchina-di-turing/13882/default.aspx	34
Figura 32- Schema di arco inibitore per la transizione T_x	35
Figura 33 - Relazione tra le diverse tipologie notevoli di Reti di Petri.....	37
Figura 34 - Esempio di configurazione per rete a scelta simmetrica	38
Figura 35 - Immagine che illustra problema del 5 filosofi (dall'alto e in senso orario Platone, Confucio, Socrate, Voltaire e Cartesio) – Fonte Benjamin D. Esham / Wikimedia Commons.....	39
Figura 36 - Documento "Information Management" con la nota Mike Sendall (fonte sito CERN - http://info.cern.ch/Proposal.html)	41
Figura 37- Schema rappresentativo di un ipertesto (fonte Treccani - http://www.treccani.it/enciclopedia/ipertesto/).....	42
Figura 38 - Il leggio rotante di Ramelli (Ignoto - de.wikipedia, uploaded as de: Bild: Ramellis Bücherrad.jpg by Asb) 43	
Figura 39 - Esempio di schermata di Hypercard 2 (fonte http://trabalhohipermedia.do.sapo.pt/hypercard.htm)	45
Figura 40 - Schema di Ipertesto a struttura assiale	45

Figura 41- Schema di Iper testo a struttura gerarchica	46
Figura 42 - Schema di Iper testo a struttura a rete o differenziale	46
Figura 43 - Evoluzione del WEB da un punto di vista della comunicazione.....	47
Figura 44 – Catalogazione dei contenuti dal Web 1.0 al WEB 2.0.....	51
Figura 45 - Nube semantica del WEB 2.0 di Markus Angermeier (The huge cloud lens bubble map web2.0 2005-11-11).....	51
Figura 46 - Rappresentazione della struttura di un Blog.....	52

APPENDICE - Slide del corso

Lezione 1

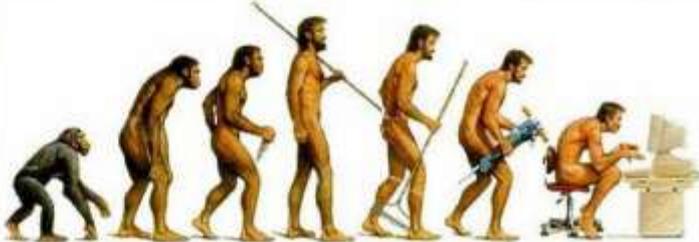


1

Informatica

CdS in «*Scienze e Tecnologie dei Beni Culturali*» – AA 2014-2015
Mini-sito dell'insegnamento: <http://www.unife.it/scienze/beni-culturali/insegnamenti/informatica>

Prof. Giorgio Paletti
giorgio.paletti@unife.it - <http://docente.unife.it/giorgio.paletti>



Introduzione all'insegnamento

«Il computer è così veloce perché non pensa.»
(Gabriel Laub)

Contenuti dell'insegnamento

- **Grafi e logica della strutturazione**
 - Grafi e soluzione dei problemi
 - Reti di Petri e sistemi non deterministici
 - Reti e schematizzazioni della realtà: dalle reti aleatorie alle reti scale free
- **Linguaggi e strumenti del Web 2.0**
 - L'ipertesto e la navigazione logica del sapere
 - Web semantico e Web 2.0
 - Strumenti del Web 2.0: Wiki, Blog, Siti e App
- **Applicazioni interattive on line e Realtà Aumentata**

Sito dell'insegnamento

- **URL:** <http://www.unife.it/scienze/beni.culturali/insegnamenti/informatica>

The screenshot shows the website for the Informatica course. At the top, there is the logo of the University of Ferrara and the text "LAUREA TRIENNALE IN SCIENZE E TECNOLOGIE PER I BENI CULTURALI". Below this is a navigation menu with "Home", "Organizzazione", "Attività didattiche", "Garanzia di qualità", and "Dove siamo". On the left, there is a sidebar menu with "Informativa", "Materiale didattico", and "Archivio". The main content area is titled "Informatica" and includes the text "Avvisi" in red. Below this, it says "Ritorno - Biblioteca on-line" and "Le lezioni inizieranno mercoledì 25 febbraio 2015 alle ore 11 (Aula 1A - Palazzo Turchi di Bagno, Corso Ercole I d'Este, Ferrara)". There are two links: "Link alla Piattaforma FAD (Frequenza A Distanza): <http://boq.unife.it>" and "Link all'iscrizione alla Piattaforma FAD: http://fad.unife.it/index.php/composenti/chioscoformi?chioscoform=buena_sistofad". The name "Prof. Giorgio Polelli" is listed below. At the bottom, there is contact information for the Dipartimento di Studi Umanistici, including the name of the coordinator Prof. Carlo Pavato, the address Corso Ercole I d'Este, 32, and the phone number Tel. 0532/201000. There is also a "Guarda la mappa" link and a page number "35".

Testi di Riferimento e Struttura esame

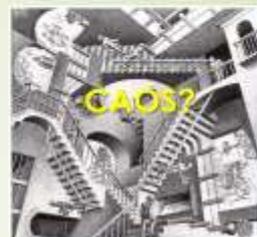
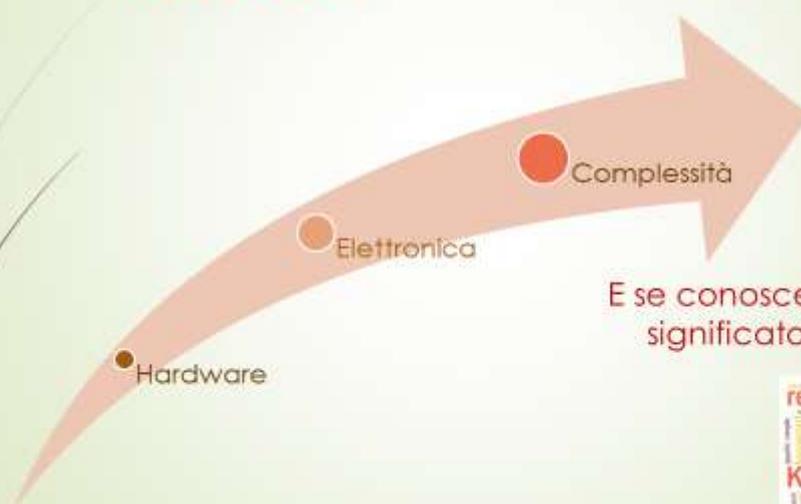
• Testi di Riferimento

- Padula M., Reggiori A., *Fondamenti di informatica per la progettazione multimediale - Dai linguaggi formali all'inclusione digitale*, Collana di Informatica, 2013, Franco Angeli Editore, Milano
- Appunti scaricabili dal sito del corso

• Prova Finale

- Prova scritta 30 items (27 chiusi + 3 aperti) - 85% del voto finale (max. 25,5 punti)
- Prova pratica: produzione di un «prodotto multimediale» (da presentare entro la data dell'esame) - 15% del voto finale (max. 4,5 punti)

Keywords TECNOLOGIA e Informatica



E se conoscessimo regole e significato dei termini?



Keywords TECNOLOGIA e Informatica

- Parlare
- Dire
- Descrivere
- [...]

Tékhne
(Τέχνη)
arte

Tecnologia

Tema
LÉGÔ

Loghía
(Λόγος)
trattato



Keywords Tecnologia e **INFORMATICA**

INFORMATICA

INFORMazione

autoMÁTICA

INFORMAZIONE

- Relazioni
- Regole

AUTOMATIZZAZIONE

- Struttura
- Descrizione Oggettiva



Keywords DATO e Informazione

Ciò che è immediatamente presente alla conoscenza, **prima di ogni forma di elaborazione**: [...] i risultati di una determinata esperienza scientifica; dati di un problema, i valori noti (o presunti noti) di talune grandezze, mediante i quali, sulla base delle relazioni e condizioni...

(treccani.it Vocabolario on line)



Keywords Dato e INFORMAZIONE

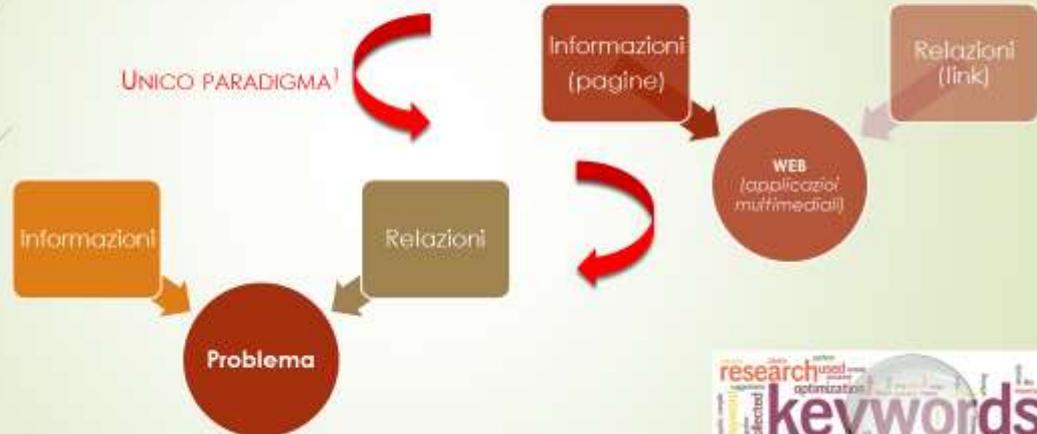
Dal latino **informatio -onis** «nozione, idea, rappresentazione» [...]. – 1. ant. e raro. L'azione dell'informare, di dare forma cioè a qualche cosa [...]

(treccani.it Vocabolario on line)



Keywords

PROBLEMA, WEB E APPLICAZIONI MULTIMEDIALI



¹Paradigma: modello di riferimento, prototipo, modello

ALGORITMO

APPROCCIO MATEMATICO

SINTESI

dato un problema f , costruire un algoritmo A che lo solve

ANALISI

dato un algoritmo A , un problema f , dimostrare che A solve f

CLASSIFICAZIONE

(*complessità strutturale*), data T , quantità di risorse, individuare la classe di problemi che usano al massimo queste risorse



MOTORE INFERENZIALE

Il termine **MOTORE INFERENZIALE** in informatica indica un procedimento che **simula** le modalità con cui la mente trae **conclusioni logiche**, a partire da **premesse**, attraverso il **ragionamento**.

TIPO DEDUTTIVO

(forward chaining)

Ricavare una **verità particolare** a partire da una **verità generale**; applicare le regole, contenute nella base di conoscenza, a un caso particolare, contenuto nei fatti noti, per trarne alcune conclusioni.

(Ad esempio Sherlock Holmes)

TIPO INDUTTIVO

(backward chaining)

Ricavare una verità generale a partire da alcune verità particolari; partire da un possibile risultato e verificare l'esistenza di una regola capace di fornire un risultato simile. Il procedimento si applica fino a trovare una regola i cui antecedenti siano fatti noti, quindi verificati per definizione.

(Ad esempio Aristotele).



MOTORE INFERENZIALE

(COMPONENTI)

- **Sceglie** le regole da applicare

INTERPRETE

- **Predisporre** l'ordine di applicazione delle regole

SCHEDULATORE

MEMORIA
DI
LAVORO

RAFFORZATORE
DI
CONSISTENZA

- **Contiene** l'elenco delle operazioni svolte e da svolgere

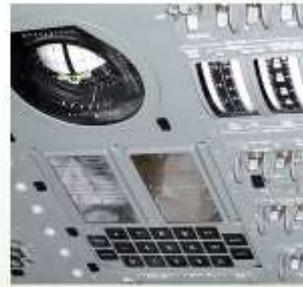
Verifica la veridicità delle ipotesi fatte



CONCLUSIONE

- La formalizzazione e le definizioni di dati e relazioni è fondamentale
- Sono le cose che sappiamo e che ci sono chiare che riusciamo a spiegare con meno parole e meno tempo (*risorse minime*)

«Non hai veramente capito qualcosa finché non sei in grado di spiegarlo a tua nonna» (Albert Einstein)



- ➔ Progetto MIT
- ➔ Unità di calcolo da 2 MHz
- ➔ 2 Kwords di memoria **RAM**
- ➔ trentina abbondante di Kwords di memoria **ROM**

Apollo Guidance Computer (AGC)



2

Informatica

CdS in «*Scienze e Tecnologie dei Beni Culturali*» – AA 2014-2015
Mini-sito dell'insegnamento: http://www.unife.it/scienze/beni_culturali/insegnamenti/informatica

Prof. Giorgio Paletti
giorgio.paletti@unife.it - <http://docente.unife.it/giorgio.paletti>



Problemi e metodi di soluzioni

«La formulazione di un problema è spesso più importante della soluzione stessa.»
(Jacob Getzels e Albert Einstein)

Contenuti della lezione

- **Metologia di soluzione dei problemi**
 - Soluzione dei problemi
 - Test di Turing
 - Problem Solving e Euristica
- **Minimo descrittivo e i linguaggi logici**
 - Il principio di **Minimum Description Length (MDL)**
 - Teoria dell'informazione e Rasoio di Occam
 - Linguaggi dichiarativi e Logica del I ordine

Soluzione dei problemi



WIRED

TEST di TURING, dal "gioco dell'imitazione", a tre partecipanti: un uomo A, una donna B, e una terza persona C. Quest'ultimo è tenuto separato dagli altri due e tramite una serie di domande deve stabilire qual è l'uomo e quale la donna. Dal canto loro anche A e B hanno dei compiti: A deve ingannare C e portarlo a fare un'identificazione errata, mentre B deve aiutarlo. Affinché C non possa disporre di alcun indizio (come l'analisi della grafia o della voce), le risposte alle domande di C devono essere dattiloscritte o similmente trasmesse.

Il **TEST di TURING** si basa sul presupposto che una macchina si sostituisca ad A. Se la percentuale di volte in cui C indovina chi sia l'uomo e chi la donna è simile prima e dopo la sostituzione di A con la macchina, allora la macchina stessa dovrebbe essere considerata intelligente, dal momento che, in questa situazione, sarebbe indistinguibile da un essere umano.

Il Problem Solving

PROBLEM FINDING

- Insiemi di procedimenti in grado di «scoprire» l'esistenza di un problema

PROBLEM SHAPING

- Insiemi di procedimenti in grado di meglio definire un problema complesso

PROBLEM SETTING

- Insiemi di procedimenti in grado di configurare in maniera cognitiva il problema riconosciuto

PROBLEM TALKING

- Insiemi di procedimenti in grado di descrivere spiegare e comunicare il problema

Il Problem Solving

PROBLEM FINDING

Intuizione

Dal latino *intueor* (*in* («dentro») + *tueor* («guardare»), cioè «entrar dentro con lo sguardo»); forma di sapere non spiegabile a parole, che si rivela per lampi improvvisi (*Knowledge discovery o Data Mining*)... Serendipità



Creatività

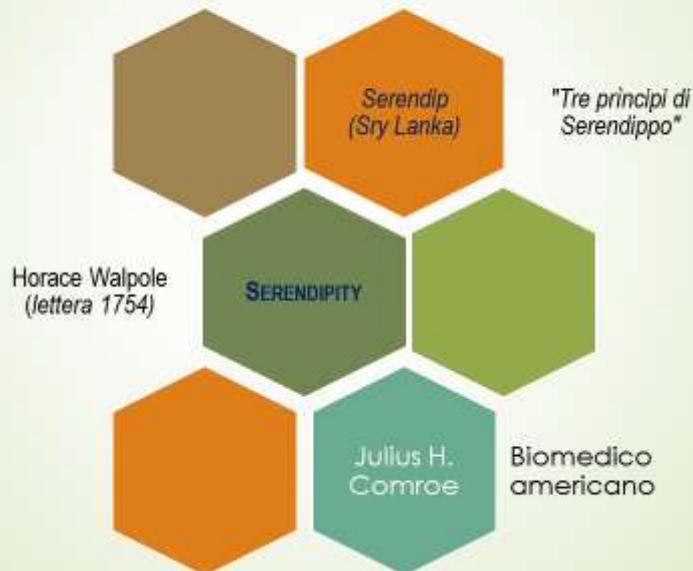
Capacità cognitiva di creare e inventare; dal «creare» latino, condivide «crescere» la radice KAR. In sanscrito, "KAR-TR" è «colui che fa» (dal niente).

SCOPRIRE UN PROBLEMA

Dal greco πρόβλημα (*próblēma*) «sporgenza, promontorio, impedimento, ostacolo»

Il Problem Solving

PROBLEM FINDING



Il Problem Solving

PROBLEM FINDING



Il Problem Solving

EURISTICA

dal greco εὕρισκω: scoprire trovare (scovare), inventare, conoscere

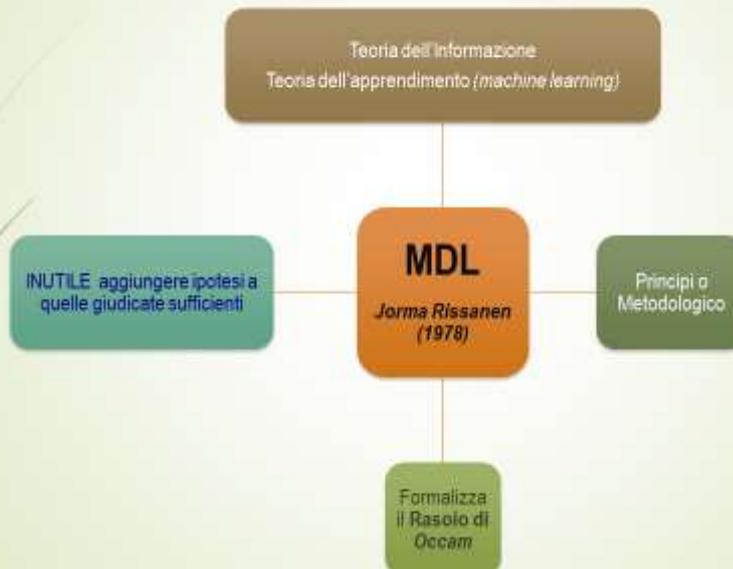
Il sistema cognitivo umano è un sistema a risorse limitate che, non potendo risolvere problemi tramite processi algoritmici, fa uso di euristiche come efficienti strategie per semplificare decisioni e problemi.



Note

Euristica: parte della ricerca il cui compito è quello di favorire l'accesso a nuovi sviluppi teorici o a scoperte empiriche (parte dell'epistemologia)
Epistemologia (episteme, «conoscenza certa» → «scienza»): filosofia della scienza, studia i fondamenti delle diverse discipline scientifiche.

Minimum Description Length (MDL)



William of Ockham - XIV sec.
(Guglielmo di Occam)



Minimum Description Length (MDL)

Rasoio di Occam

Eliminare con tagli di lama e approssimazioni successive le ipotesi più complicate

E inutile fare con più ciò che si può fare con meno.

Non moltiplicare gli elementi più del necessario

A parità di sforzi la spiegazione più semplice è da preferire

Non considerare la pluralità se non è necessario

Dico ergo ad quoniam quod pluralitas non est ponenda sine necessitate et non est necessitas quare occurrat plura de factum minus motum augere. nam

Parte di una pagina di Ordinale di Duns Scoto:
Pluralitas non est ponenda sine necessitate
(Non considerare la pluralità se non è necessario.)

Minimum Description Length (MDL)



Reverendo
Thomas Bayes
(1702-1761)

Rasoio di Occam

Approccio all'**inferenza statistica**: le probabilità **non** sono interpretate come **frequenze** o **proporzioni**, ma piuttosto come **livelli di fiducia** nel verificarsi di un dato evento.

Teoria dell'Informazione

Concetti di Inferenza Bayesiana

Procedimento per cui si **inducono** le caratteristiche di una popolazione dall'osservazione di una parte di essa (il campione), parte determinata, normalmente, mediante un esperimento casuale (aleatorio).

Linguaggi Dichiarativi



Logica proposizionale (o enunciativa)

- linguaggio formale, semplice struttura sintattica
- basata su proposizioni elementari (atomi) e su connettivi logici di tipo vero-funzionale (come AND, OR, NOT...)

Linguaggio dichiarativo: linguaggio di programmazione in cui si descrivono l'**obiettivo** e un **compito**, ma senza scrivere il codice per realizzare il compito.

Linguaggio del I ordine



Simboli per

variabili (infiniti): x_1, x_2, x_3 , costanti individuali (anche nessuno): a_1, a_2, \dots , predicati (o relazioni) con il loro numero di argomenti: $P_1, Q_1, P_2, Q_2, \dots$, funzioni con associato un numero corrispondente al numero di argomenti: f, g, h, \dots , punteggiatura: "(", ")", "e" e la virgola ",", connettivi logici: negazione, implicazione, se e solo se), e, oppure simboli e quantificatori: \forall (quantificatore universale), \exists (quantificatore esistenziale)

Definizioni

1. una costante individuale è un termine
2. una variabile è un termine
3. se t_1, \dots, t_n sono n termini e f è un simbolo per n-funzione $f(t_1, \dots, t_n)$ è un termine
4. nient'altro è un termine

Definizione

1. **formula atomica** una sequenza di simboli del tipo $A(t_1, \dots, t_n)$, **A simbolo** e t_1, \dots, t_n , **termini**
 - i. ogni formula atomica è **fbf**
 - ii. Se A e B sono **fbf** e x è una variabile ogni combinazione di A, B e X attraverso connettivi logici e quantificatori è **fbf**
 - iii. **tutte e sole le fbf** sono definite dalle regole precedenti

Logica dei predicati del I ordine

Logica del I ordine

Sistema formale

1. *Alfabeto*
2. *Grammatica*
3. *Assiomi*
4. *Regole di Inferenza*

Enunciati (proposizioni)

Proposizione: soggetto e *predicato*, e del verbo (copula): (è) vero oppure (è) falso
Validità: un enunciato P è valido *se e solo se* è vero in ogni interpretazione.

Falsità: un enunciato P è falso *se e solo se* è falso in ogni interpretazione.

Inconsistenza: un insieme di enunciati è inconsistente *se e solo se* l'insieme non è consistente

Definizione di implicazioni logiche

Equivalenza: due enunciati P e Q sono equivalenti *se e solo se* in nessuna interpretazione P e Q hanno valori differenti

Consistenza: un insieme di enunciati è consistente *se e solo se* c'è almeno una interpretazione in cui tutti gli elementi dell'insieme sono veri

Un insieme G di enunciati implica logicamente un enunciato P *se e solo se* in nessuna interpretazione tutti gli elementi di G sono veri e P è falso



Conclusioni

«La logica, che può dare soltanto la certezza, è lo strumento della dimostrazione; l'intuizione, lo strumento dell'invenzione.»
(Henri Poincaré - *Il valore della scienza*)



Informatica

CdS in «*Scienze e Tecnologie dei Beni Culturali*» – AA 2014-2015

Mini-sito dell'insegnamento: <http://www.unife.it/scienze/beni.culturali/insegnamenti/informatica>

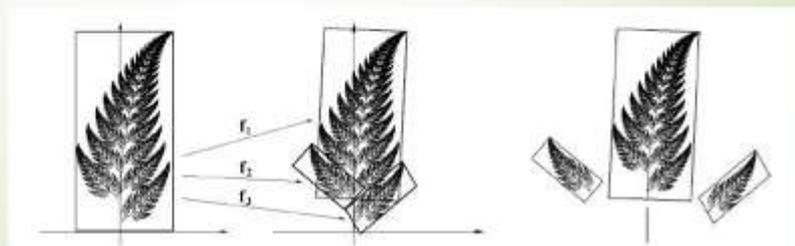
Prof. Giorgio Paletti

giorgio.paletti@unife.it - <http://docente.unife.it/giorgio.paletti>



Rappresentazione dei Problemi

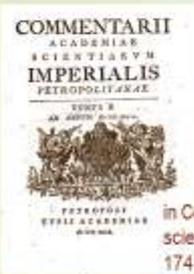
«La scienza è conoscenza organizzata.»
(Herbert Spencer)



Topologia e approccio alla soluzione dei problemi

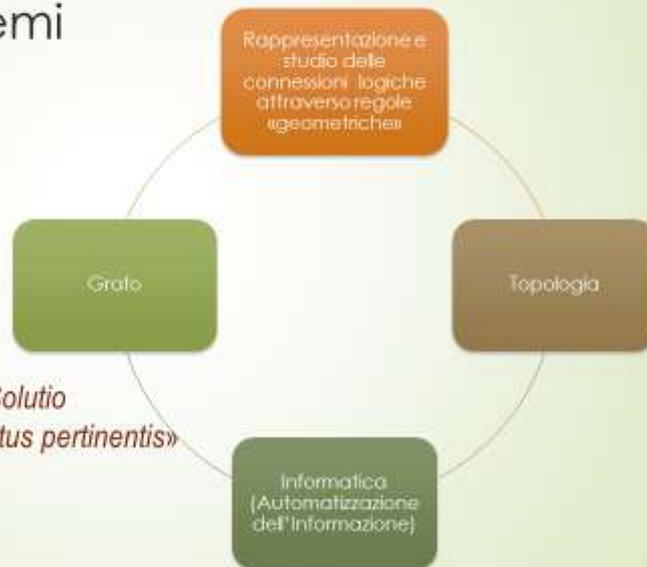


Topologia e approccio alla soluzione dei problemi



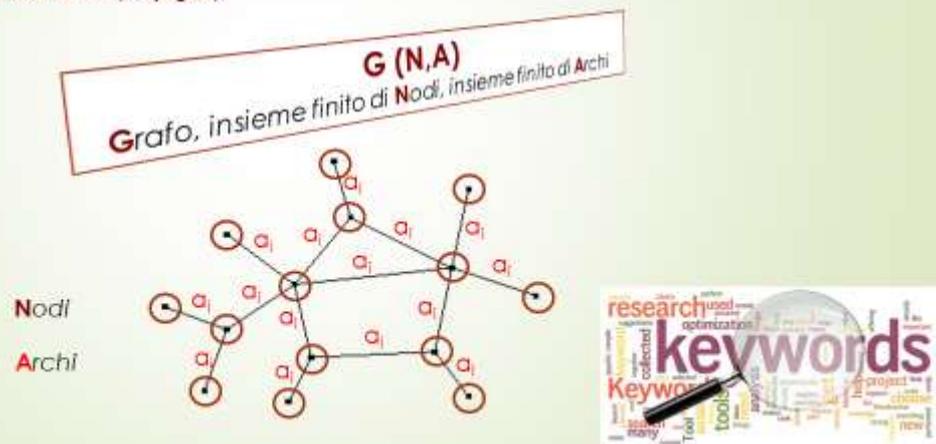
in Commentarii academiae scientiarum Petropolitanae 8, 1741, pp. 128-140

Grafo per la prima volta In «*Solutio problematis ad geometriam situs pertinentis*»
Eulero, 1736



Grafo, Eulero e i Ponti di Königsberg

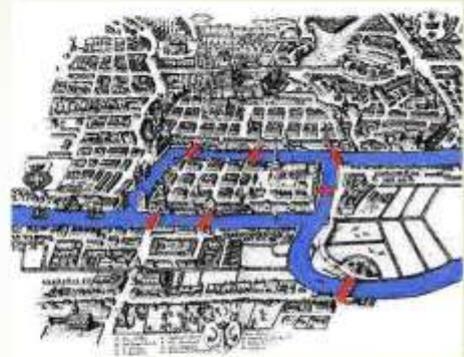
Struttura **relazionale** composta da un insieme finito di oggetti (un insieme finito di punti) detti **nodi** (o **vertici**) e da un insieme di relazioni (geometricamente segmenti di retta o di curva) tra coppie di oggetti detti **archi** (o **spigoli**).



Grafo, Eulero e i Ponti di Königsberg

Problema dei
Ponti di
Königsberg

Teorema
di Eulero
(Leonhard Euler)



Mappa di Königsberg ai tempi di Eulero, e la disposizione dei ponti sul fiume Pregel

Teorema di Eulero (il grafo euleriano)

«Condizione necessaria e sufficiente affinché un grafo sia percorribile completamente partendo da un nodo e ritornandovi passando una volta solamente per ciascun arco è che esista un percorso (cammino) fra ogni coppia di nodi e che ogni nodo sia toccato da un numero pari di archi.»

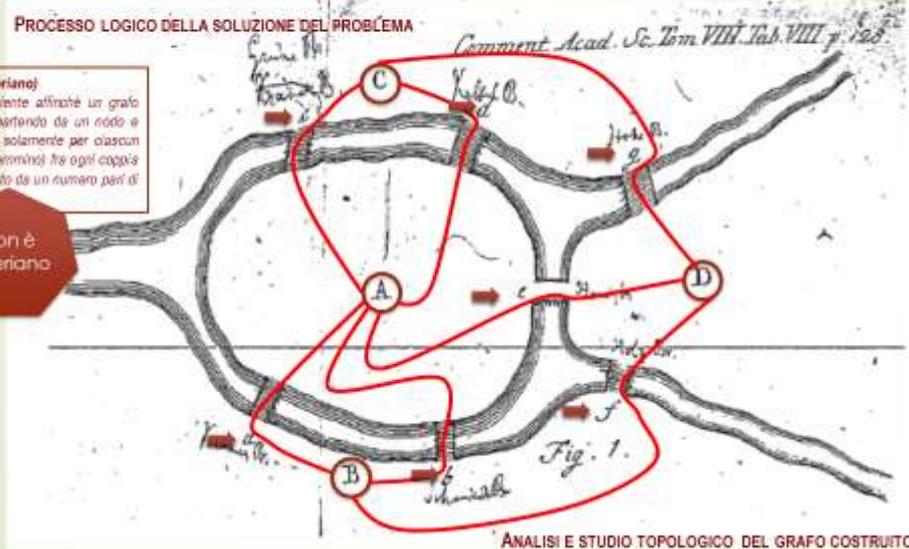
Grafo, Eulero e i Ponti di Königsberg

PROCESSO LOGICO DELLA SOLUZIONE DEL PROBLEMA

Teorema di Eulero (il grafo euleriano)

«Condizione necessaria e sufficiente affinché un grafo sia percorribile completamente partendo da un nodo e ritornandovi passando una volta solamente per ciascun arco è che esista un percorso (cammino) tra ogni coppia di nodi e che ogni nodo sia toccato da un numero pari di archi.»

Non è euleriano

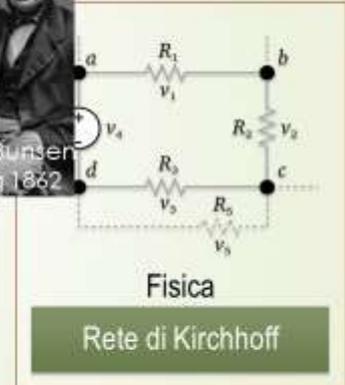
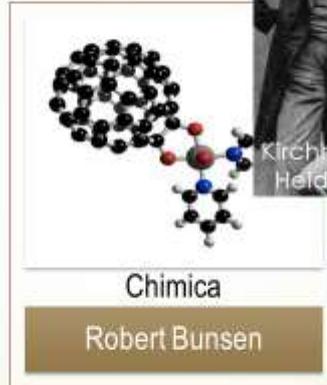


ANALISI E STUDIO TOPOLOGICO DEL GRAFO COSTRUITO

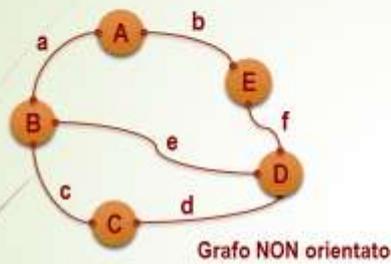
Grafo: schema di relazioni



Grafo: primi esempi



Grafo: sistemi di memorizzazione

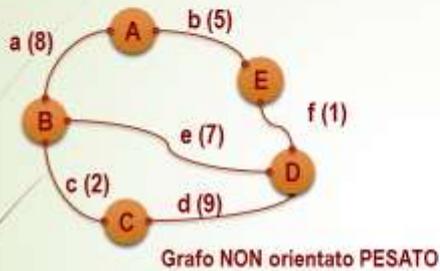


LISTA DI ADIACENZA	
A	B,E
B	A,C,D
C	B,D
D	B,C,E
E	A,D

MATRICE DI CONNESSIONE	A	B	C	D	E
A	0	1	0	0	1
B	1	0	1	1	0
C	0	1	0	1	0
D	0	1	1	0	1
E	1	0	0	1	0

MATRICE DI INCIDENZA	A	B	C	D	E
a	1	1	0	0	0
b	1	0	0	0	1
c	0	1	1	0	0
d	0	0	1	1	0
e	0	1	0	1	0
f	0	0	0	1	1

Grafo: sistemi di memorizzazione

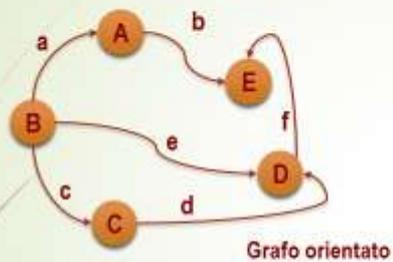


LISTA DI ADIACENZA	
A	B,E
B	A,C,D
C	B,D
D	B,C,E
E	A,D

MATRICE DI CONNESSIONE	A	B	C	D	E
A	0	8	0	0	5
B	8	0	2	7	0
C	0	2	0	9	0
D	0	7	9	0	1
E	5	0	0	1	0

MATRICE DI INCIDENZA	A	B	C	D	E
a	1	1	0	0	0
b	1	0	0	0	1
c	0	1	1	0	0
d	0	0	1	1	0
e	0	1	0	1	0
f	0	0	0	1	1

Grafo: sistemi di memorizzazione



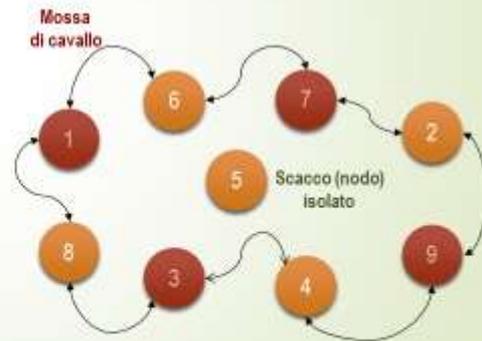
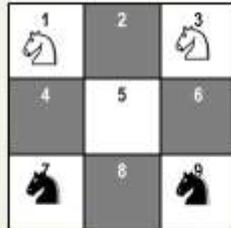
LISTA DI ADIACENZA	
A	E
B	A,C
C	D
D	E
E	

MATRICE DI CONNESSIONE	A	B	C	D	E
A	0	0	0	0	1
B	1	0	1	1	0
C	0	0	0	1	0
D	0	0	0	0	1
E	0	0	0	0	0

MATRICE DI INCIDENZA	A	B	C	D	E
a	-1	1	0	0	0
b	1	0	0	0	-1
c	0	1	-1	0	0
d	0	0	1	-1	0
e	0	1	0	-1	0
f	0	0	0	1	-1

Grafo: il problema dei 4 cavalli

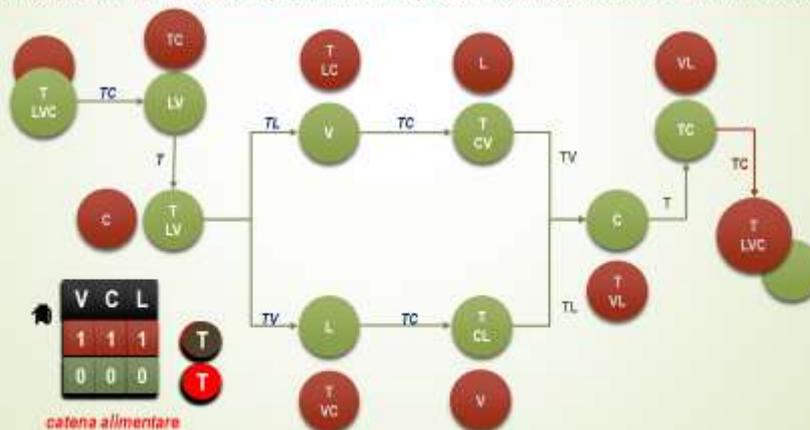
«Data una scacchiera 3x3 e numerate le caselle da 1 a 9, e posti i cavalli degli scacchi nelle caselle 1 e 3 (cavalli bianchi) e nelle caselle 7 e 9 (cavalli neri), ci si domanda «se è possibile cambiare di posto ai cavalli (i bianchi in 7 e 9 ed i neri in 1 e 3) **spostando un cavallo alla volta secondo la modalità degli scacchi** senza mai avere due cavalli nella medesima casella.»



Grafo: salvare "capra e verzi"

Tartaglia (libro 16, N. 141) scrive anche «e da questo è nasciuto un certo proverbio fra gli huomini, dicendo in qualche proposito, egli ha salvato la capra e i verzi».

Si tratta di questo: «un uomo (T) vuole traghettare da una sponda all'altra di un fiume un lupo (L), una capra (C) un cavolo (V) su di una barca capace solo di ospitare l'uomo e il cavolo ed una sola delle due bestie».

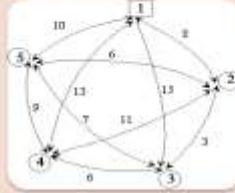


Grafo: i 4 problemi fondamentali



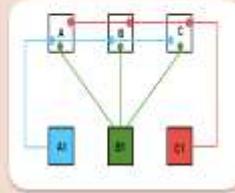
I ponti di KÖNIGSBERG

(EULERO 1736)



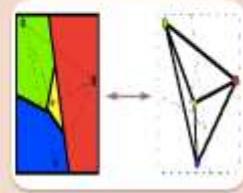
Il percorso del
commesso
viaggiatore

(TSP - TRAVELLING
SALES MAN PROBLEM
HAMILTON - 1856)



Le tre case e le tre
forniture

(PROBLEMA DI
TOPOLOGIA SUL PIANO)



Il teorema dei quattro
colori

(CONGETTURA DI
FRANCIS GUTHRIE,
1852)

Grafi e Reti





Informatica

4

CdS in «*Scienze e Tecnologie dei Beni Culturali*» – AA 2014-2015

Mini-sito dell'insegnamento: http://www.unife.it/scienze/beni_culturali/insegnamenti/informatica

Prof. Giorgio Paletti

giorgio.paletti@unife.it - <http://docente.unife.it/giorgio.paletti>



Richard Phillips Feynman

Grafi e Problemi: esempi notevoli

«Non so che cosa non va nella gente: non imparano usando l'intelligenza, ma solo meccanicamente o giù di lì. Il loro sapere è così fragile.»

(Richard Phillips Feynman – da «*Stascherzando, Mr. Feynman*».)



Contenuti della lezione

■ Grafi e Problemi

- Tipologia e rappresentazione: problemi-grafi
- I quattro problemi fondamentali
- Problemi, grafi e applicazioni

■ Problemi di SP (Short Path)

- Esempi notevoli di SP
- Algoritmo di Dijkstra
- Grafi e reti sociali: i 6 gradi di separazione

I 4 problemi fondamentali: applicazioni



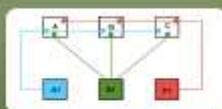
IL PROBLEMA DEI PONTI DI KÖNIGSBERG

1. Distribuzione, controllo e manutenzioni di reti, ad esempio elettriche, idriche o stradali
2. Ottimizzazione di percorsi ad esempio distribuzione della posta (Chinese Postman's Problem)



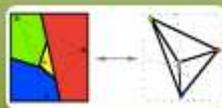
IL PROBLEMA DEL COMMESO VIAGGIATORE

1. Flussi di merci, ad esempio distribuzione merci tra magazzini, clienti e fornitori
2. Minimizzazione di percorsi (problemi SP, Short Path)



IL PROBLEMA DELLE TRE CASE E LE TRE FORNITURE

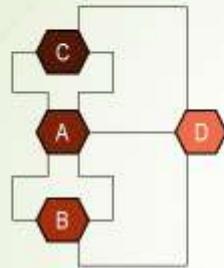
1. Layout di reti idriche, stradali, elettriche e circuiti stampati
2. Layout di reti telematiche, connessione e collegamento tra computer



IL TEOREMA DEI QUATTRO COLORI

1. Test di controllo di circuiti stampati
2. Allocazione e assegnazione registri CPU e frequenza radiotelevisive

I ponti di Königsberg (sviluppi)

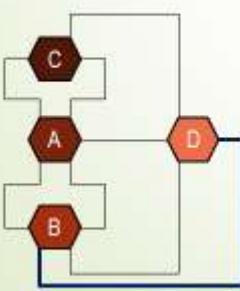


PROBLEMA BASE

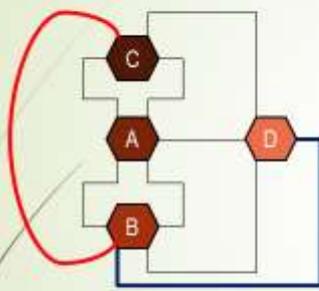
La città di Königsberg, è percorsa dal fiume Pregel e da suoi affluenti e presenta due estese isole che sono connesse tra di loro e con le due aree principali della città da sette ponti. Ci si pone la questione se sia possibile con una passeggiata seguire un percorso che attraversa ogni ponte una e una volta sola e tornare al punto di partenza.

L'OTTAVO PONTE DEL PRINCIPE BLU

Il principe Blu, dopo aver analizzato il sistema dei ponti cittadini con l'aiuto della teoria dei grafi, si convince dell'impossibilità di passare i ponti. Decide allora di costruire di nascosto un ottavo ponte che gli permetta la sera di passare i ponti partendo dal suo **Schloß** (castello) e finendo alla **Gasthaus** (osteria) dove potersi vantare della sua riuscita; e inoltre fa in modo che il principe Rosso non riesca a fare altrettanto a partire dal suo Schloß.. Dove costruisce l'ottavo ponte il principe Blu?



I ponti di Königsberg (sviluppi)

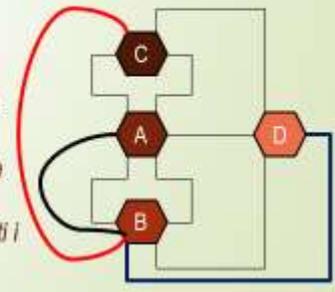


IL NONO PONTE DEL PRINCIPE ROSSO

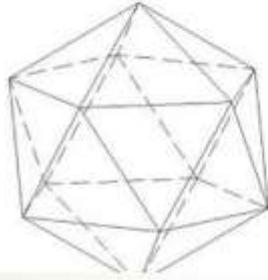
Il principe Rosso, imbufalito per la mossa del fratello, capisce che può reagire solo dopo aver studiato la teoria dei grafi; dopo un attento studio anche lui decide di costruire di nascosto un altro ponte che consenta a lui di traversare i ponti in modo di raggiungere dal suo Schloß la Gasthaus e qui prendere per i fondelli il fratello al quale diventa impossibile passare i ponti alla sua maniera.

IL DECIMO PONTE DEL VESCOVO

Il Vescovo ha dovuto assistere alla dispendiosa contesa cittadina con crescente irritazione. Essa ha portato alla formazione di due facinorose fazioni e ha fatto crescere il numero degli eccessivi frequentatori della Gasthaus, con danno della quiete pubblica. Quindi anche lui, dopo un accurato studio della teoria dei grafi, decide di costruire un decimo ponte che consenta a tutti i cittadini di passare tutti i ponti e fare ritorno alla propria casa tra i tranquilli affetti familiari.



Il problema del commesso viaggiatore: i grafi hamiltoniani



William Rowan Hamilton (1805-1865) inventò un gioco da tavolo, il **puzzle di Hamilton** o **icosian game**.



Un grafo è detto **hamiltoniano** se è possibile trovare un percorso che tocca tutti i vertici del grafo una e una sola volta (**analogo** del grafo euleriano per gli archi)

Il problema del commesso viaggiatore: i grafi hamiltoniani

- Archi
- Condizione necessaria e sufficiente per definire un grafo euleriano

Teorema di Eulero

Teorema di Dirac

- Nodi
- Condizione sufficiente per un grafo hamiltoniano.

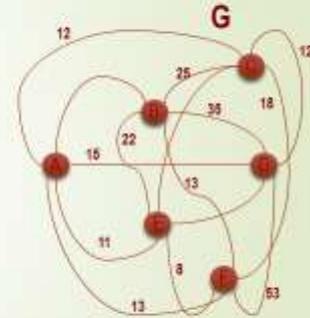
TEOREMA DI DIRAC
Condizione sufficiente (ma non necessaria) affinché un grafo con n vertici sia hamiltoniano: il grado di ogni vertice (cioè il numero di archi incidenti) deve essere maggiore o uguale a $n / 2$.

Il problema del commesso viaggiatore e i grafi hamiltoniani

(TSP: Travelling Salesman Problem)

Data una rete di città, connesse tramite delle strade, trovare il percorso di minore lunghezza che un commesso viaggiatore deve seguire per visitare tutte le città una e una sola volta?

Esiste un cammino hamiltoniano, la mappa è un grafo hamiltoniano?



Reti di città rappresentata in G

- città → nodi
- strade → archi
- distanze → i pesi sugli archi



Problema tipico per lo studio dell'informatica teorica e DELLA TEORIA DELLA COMPLESSITA' (complessità descrittiva, detta anche Teoria K-C-S da Kolmogorov, Chaitin e Solomonoff)

Il problema del commesso viaggiatore: una classe di problemi

Problema del Commesso Viaggiatore
TSP
(Travel Salesman Problem)

Telecomunicazioni
(MIN-DELAY PATH PROBLEM)

Problema di Cammino Minimo
SP
(Short Path)

«Dato un **grafo pesato** qual è il **cammino** che unisce 2 nodi dati che è minimo rispetto al valore della somma dei costi (pesi) associati a ciascun arco?»

Short Path problem: Algoritmo di Dijkstra

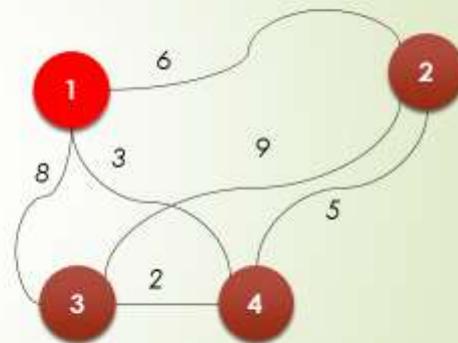


Short Path problem: Algoritmo di Dijkstra

$S=\{1\}$, $T=\{2,3,4\}$
 $p(1,2) = 6$
 $f(3)=8$
 $J(3)=1$

PREMESSE e NOTAZIONI

- Grafo con n nodi distinti
- Nodi numerati (1 partenza, n arrivo)
- Archi "pesati" e indicato $p[i,k]$
- Etichetta $f(i)$ peso del cammino per giungere al nodo i
- Peso di ogni arco >0
- Etichetta $J(i)$ nodo che precede il nodo i nel cammino minimo
- S insieme dei nodi etichettati
- T l'insieme dei nodi non etichettati



Short Path problem: Algoritmo di Dijkstra

DESCRIZIONE ALGORITMICA: (G GRAFO DI RIFERIMENTO)

INIZIALIZZAZIONE

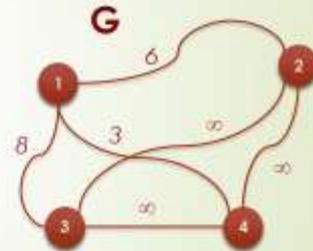
- $S=\{1\}$, $T=\{2,3,4\}$, $f(1)=0$, $J(1)=0$
- $f(i)=p(1,i)$, $J(i)=1$ per tutti i nodi adiacenti a 1
- $f(i) = \infty$ (infinito) per tutti gli altri nodi

ASSEGNAZIONE DI ETICHETTA PERMANENTE

- Se $f(i)=\infty$, per ogni i in $T \rightarrow$ STOP
- Si trova j per cui $f(j) = \min f(i)$ di tutti gli i in T
- $T=T-\{j\}$ e $S=S \cup \{j\}$
- Se $T=\emptyset$ STOP

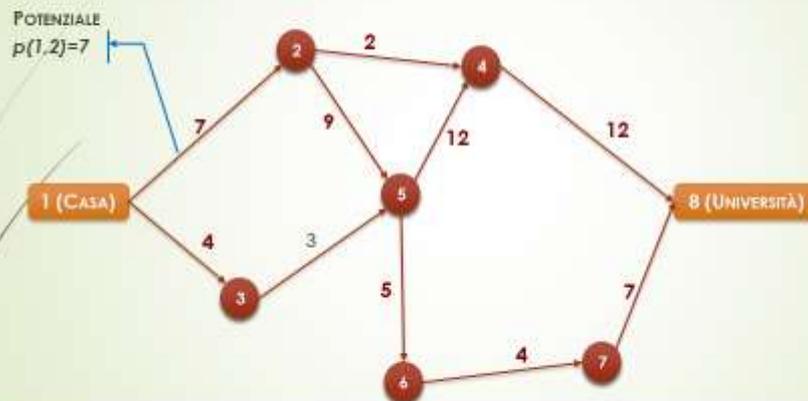
ASSEGNAZIONE DI ETICHETTA PROVVISORIA

- Per ogni i in T , adiacente a j per cui $f(i) > f(j) + p(i,j)$
 - $f(i) = f(j) + p(i,j)$
 - $J(i) = j$
- Si torna alla procedura "ASSEGNAZIONE DI ETICHETTA PERMANENTE"



Short Path problem: Algoritmo di Dijkstra

Percorso Casa-Università

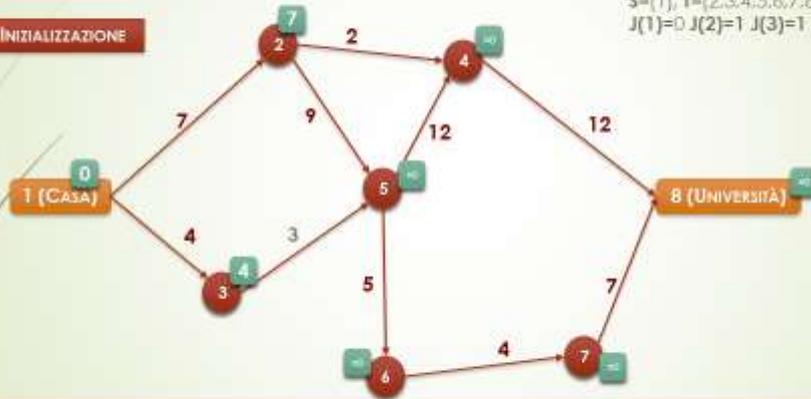


Grafo schema del problema

Short Path problem: Algoritmo di Dijkstra

Percorso Casa-Università

INIZIALIZZAZIONE

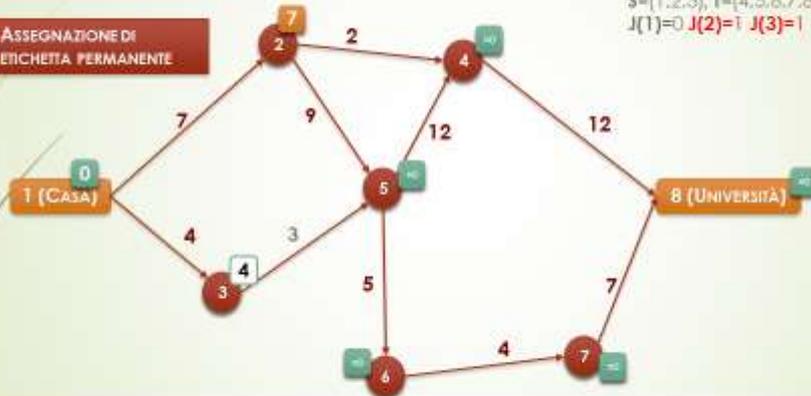


Grafo schema del problema

Short Path problem: Algoritmo di Dijkstra

Percorso Casa-Università

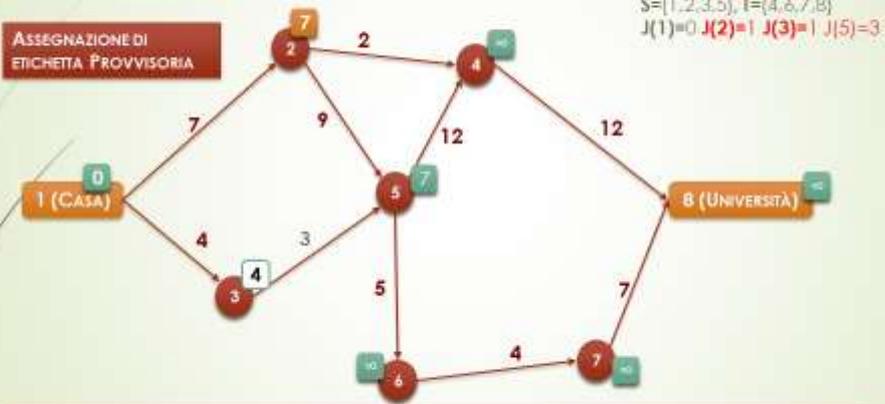
ASSEGNAZIONE DI
ETICHETTA PERMANENTE



Grafo schema del problema

Short Path problem: Algoritmo di Dijkstra

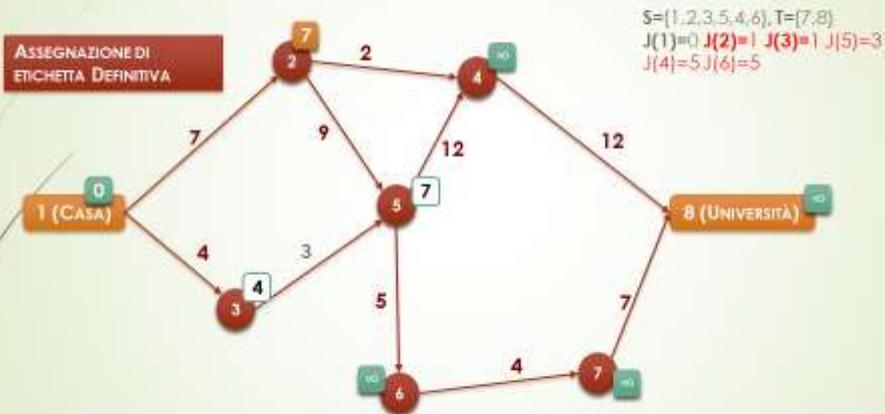
Percorso Casa-Università



Grafo schema del problema

Short Path problem: Algoritmo di Dijkstra

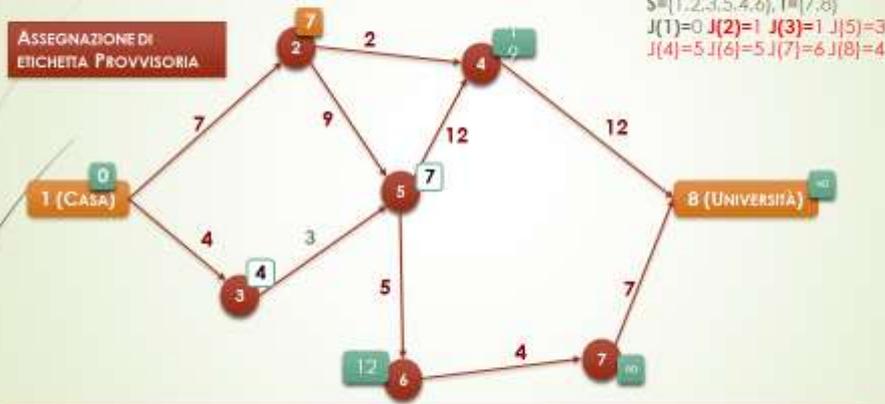
Percorso Casa-Università



Grafo schema del problema

Short Path problem: Algoritmo di Dijkstra

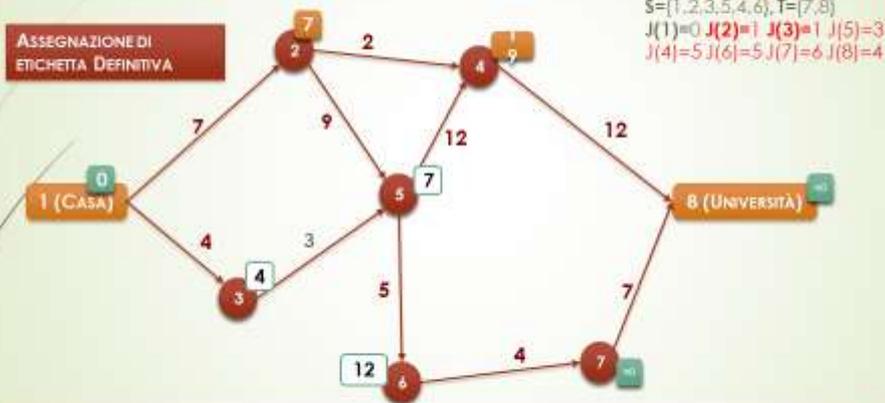
Percorso Casa-Università



Grafo schema del problema

Short Path problem: Algoritmo di Dijkstra

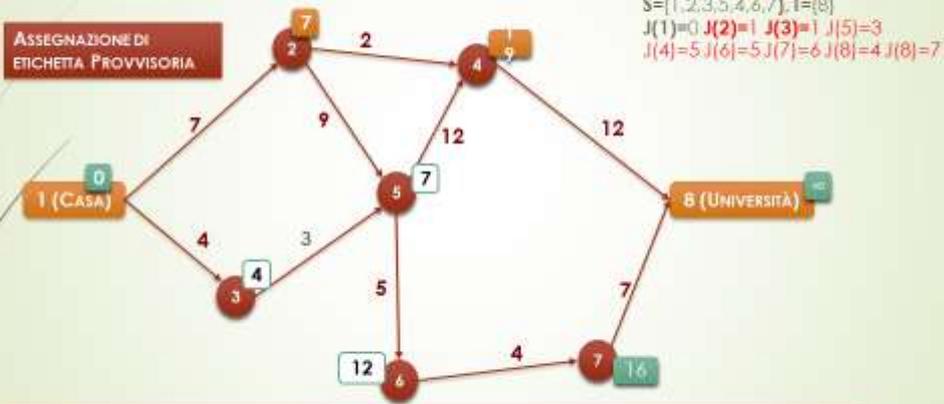
Percorso Casa-Università



Grafo schema del problema

Short Path problem: Algoritmo di Dijkstra

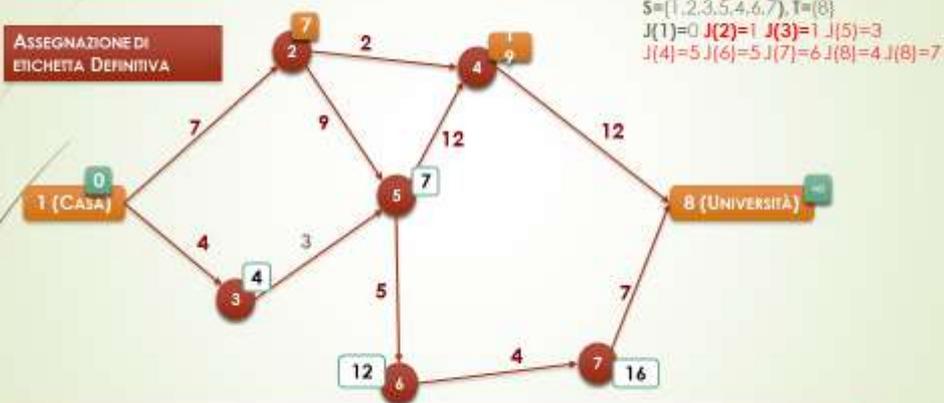
Percorso Casa-Università



Grafo schema del problema

Short Path problem: Algoritmo di Dijkstra

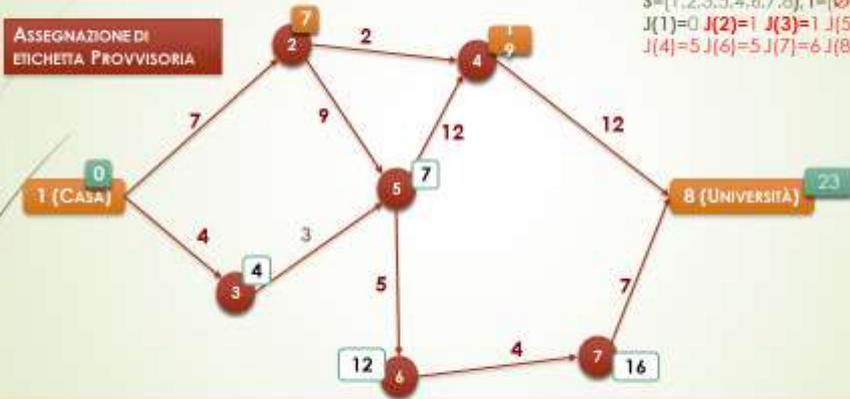
Percorso Casa-Università



Grafo schema del problema

Short Path problem: Algoritmo di Dijkstra

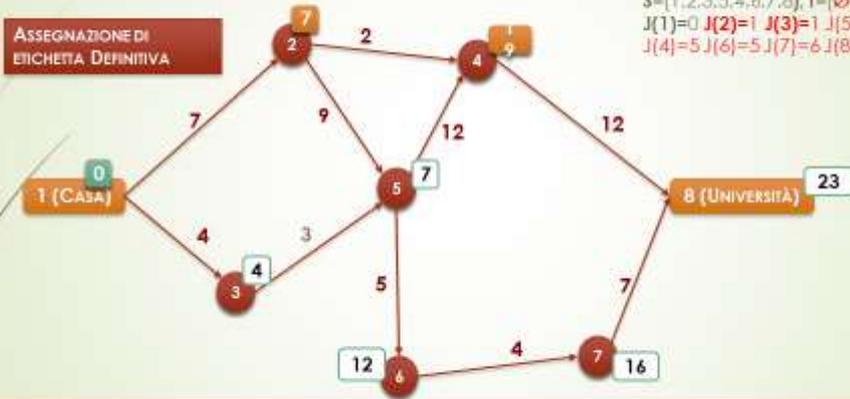
Percorso Casa-Università



Grafo schema del problema

Short Path problem: Algoritmo di Dijkstra

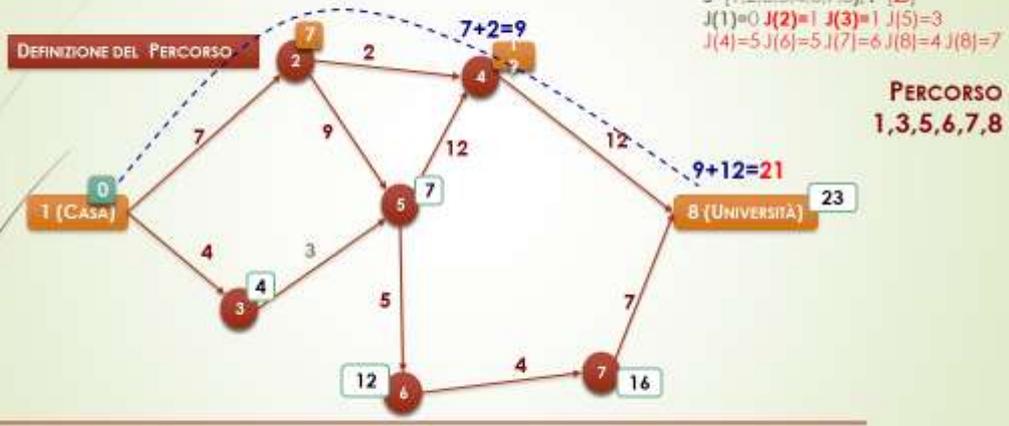
Percorso Casa-Università



Grafo schema del problema

Short Path problem: Algoritmo di Dijkstra

Percorso Casa-Università



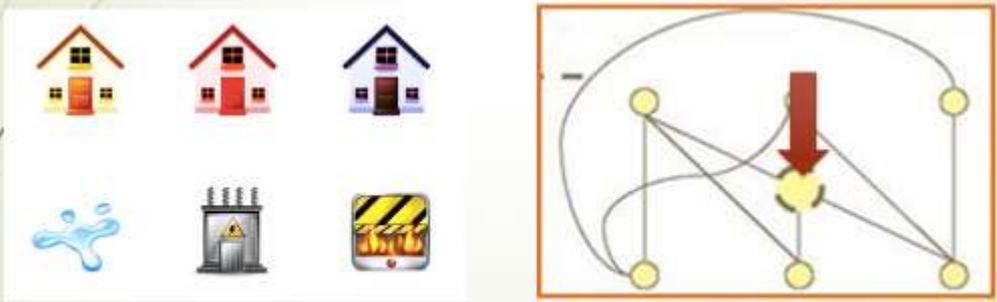
$S = \{1, 2, 3, 5, 4, 6, 7, 8\}$, $T = \{8\}$
 $J(1)=0$ $J(2)=1$ $J(3)=1$ $J(5)=3$
 $J(4)=5$ $J(6)=5$ $J(7)=6$ $J(8)=4$ $J(8)=7$

PERCORSO
 1, 3, 5, 6, 7, 8

Grafo schema del problema

Il problema delle tre case e delle tre forniture

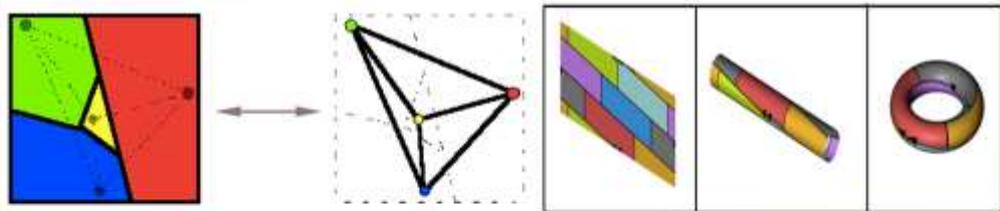
Su una pianura sono presenti una centrale idrica, una centrale elettrica ed una centrale del gas che devono rifornire tre case, anch'esse poste nella pianura. Ciascuna centrale utilizza delle linee di distribuzione (cavi elettrici, gasdotti ed idrodotti) interrati che, per motivi di sicurezza e manutenzione, non possono incrociare quelle di altre centrali. Posto che ciascuna casa deve essere fornita, mediante una linea propria, sia dell'acqua che del gas e della corrente elettrica, riesci a collegare ciascuna centrale a ciascuna casa senza che nessuna linea di distribuzione ne incroci un'altra?



Teorema di Kazimierz Kuratowski

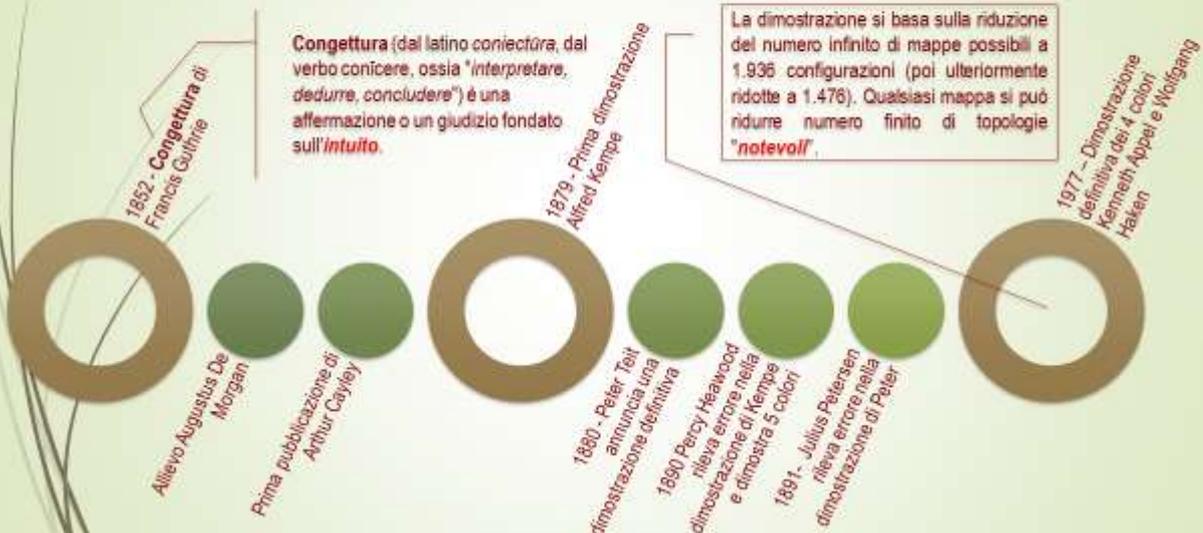
Il problema dei quattro colori

Data una superficie piana divisa in regioni connesse, come ad esempio una carta geografica politica, sono sufficienti quattro colori per colorare ogni regione facendo in modo che regioni adiacenti non abbiano lo stesso colore? (Due regioni sono dette adiacenti se hanno almeno un segmento di confine in comune)



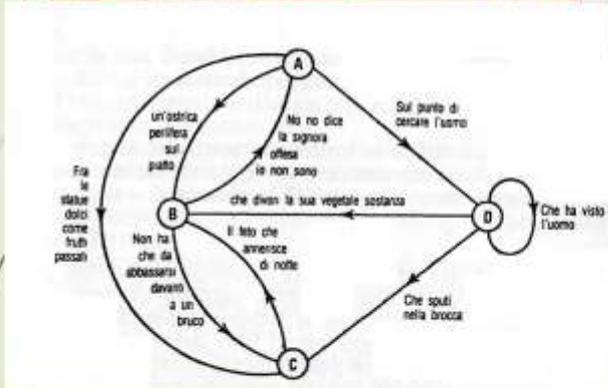
Teoria dei grafi: i nodi di un grafo **planare** possono essere colorati utilizzando al massimo quattro colori, in modo tale che due vertici adiacenti non ricevano mai lo stesso colore (ogni grafo planare è 4-colorabile)

Il problema dei quattro colori

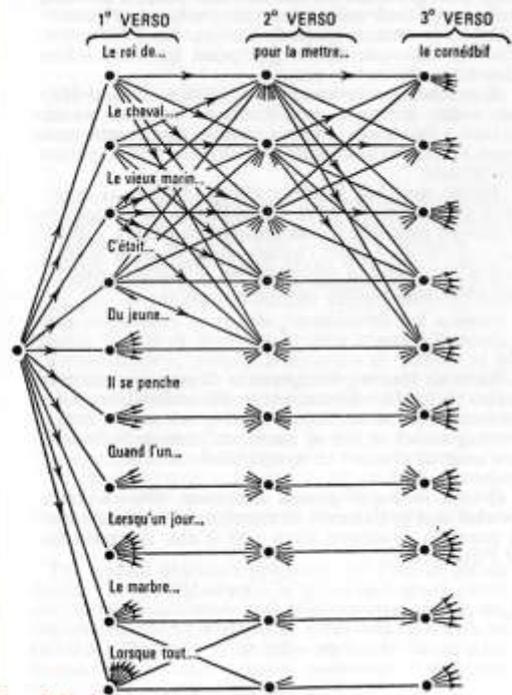


Grafi e poesie

Grafo per "poesie hamiltoniane"
 (da <http://keespoingna.blogspot.it/2012/02/poesia-in-forma-di-grafo.html>)



Grafo semplificato dell'opera di Raymond Queneau
 (da <http://keespoingna.blogspot.it/2012/02/poesia-in-forma-di-grafo.html>)



Contenuti della lezione

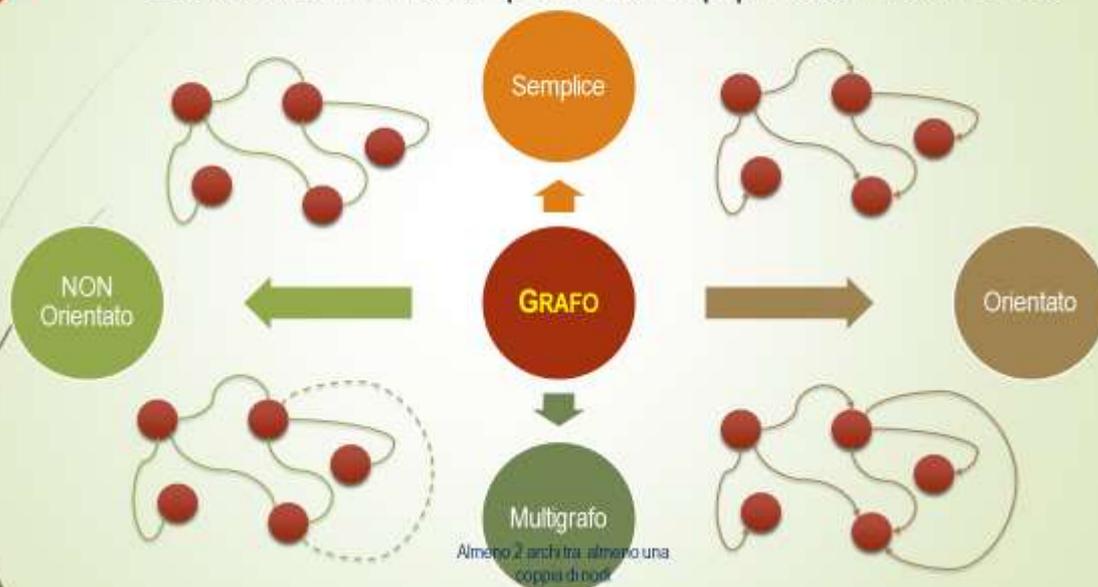
■ Grafi

- Catalogazione dei grafi
- Tipologie di grafi
- Alberi

■ Reti

- Sistemi distribuiti discreti
- Reti di Petri e soluzione di problemi non deterministici

Grafo, struttura per la rappresentazione



Grafo: cammino come rappresentazione delle soluzioni

NODI ADIACENTI

- Coppia di nodi *connessi* da un arco

NODI ISOLATI

- Nodi privi di nodi adiacenti

ARCHI, NON ORIENTATI, ADIACENTI

- Coppia di archi non orientati con un nodo in comune

ARCHI, ORIENTATI, ADIACENTI

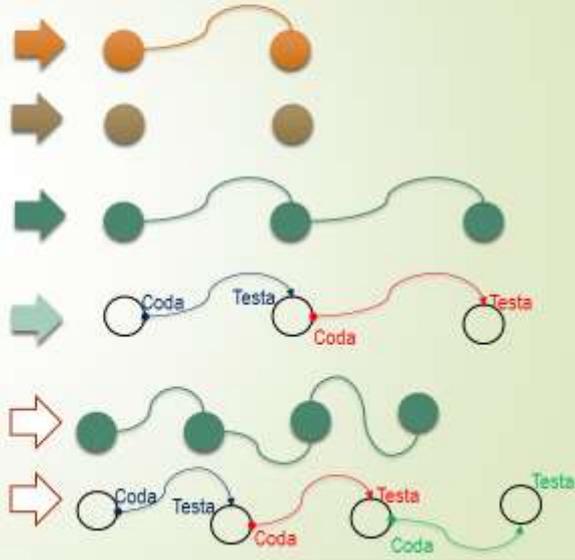
- coppia di archi, orientati con un nodo estremo in comune **Testa-Coda**

CATENA (CAMMINO NON ORIENTATO)

- Sequenza di archi adiacenti

CAMMINO

- Sequenza di archi orientati adiacenti



Grafo: cammino come rappresentazione delle soluzioni

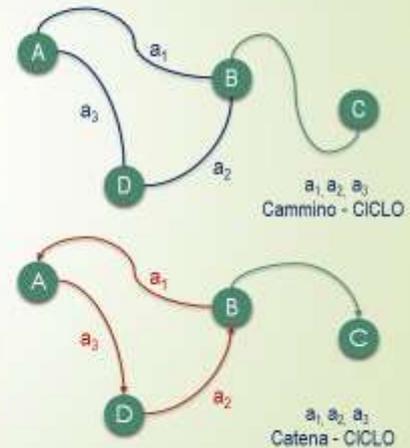
CICLO

Cammino

Catena

Il nodo finale coincide...

...con il nodo iniziale



Grafo: cammino come rappresentazione delle soluzioni

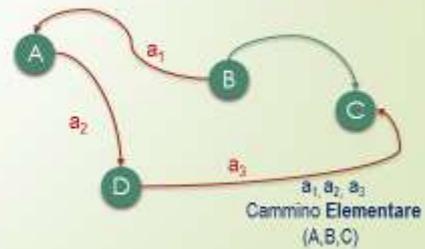
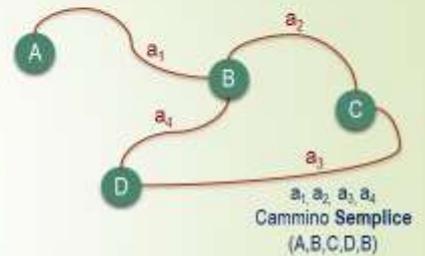
GRAFO ORIENTATO

Cammino Semplice

non passa mai 2 volte per lo stesso **arco**
(a_1, a_2, a_3, a_4)

Cammino elementare

non passa mai 2 volte per lo stesso **nodo**
(a_1, a_2, a_3)



Grafo: cammino come rappresentazione delle soluzioni

CONNESSIONE

Grafo NON Orientato

Grafo Connesso
(esiste una catena per ogni coppia di nodi)

Grafo Orientato

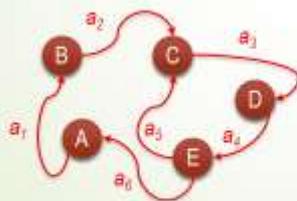
Grafo Debolmente Connesso
(esiste un cammino NON Orientato tra ogni coppia di nodi)

Grafo Fortemente Connesso
(esiste un cammino Orientato tra ogni coppia di nodi)

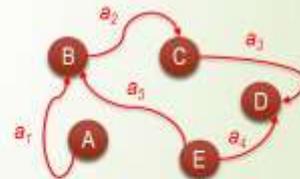
Grafo: cammino come rappresentazione delle soluzioni



GRAFO NON ORIENTATO CONNESSO



GRAFO ORIENTATO FORTEMENTE CONNESSO



GRAFO ORIENTATO DEBOLMENTE CONNESSO

Grafo: alcuni indicatori

Grafo Completo

- Un arco per ogni coppia di nodi
- Internet (ARPANET)

Cardinalità di un nodo

- Numero di archi che incidono su un nodo
- *Teorema di Eulero*, tutti i nodi a cardinalità pari

Grafo Regolare

- Tutti i nodi con uguale cardinalità (*k*-regolare)
- *Teorema delle 3 forniture* ($k=3$)

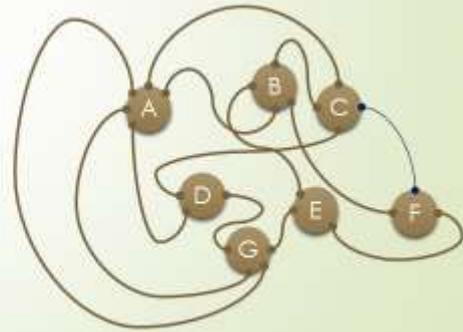
Grafi e disposizioni nel piano

Planare

- Archi tutti su un piano
- Archi non si intersecano

Non Planare

- Archi tutti su un piano
- Impossibile disporre gli archi senza che si intersechino



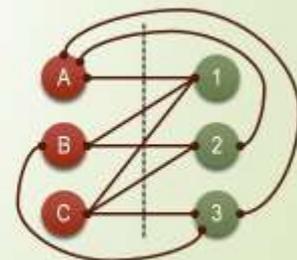
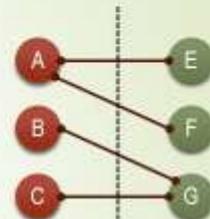
Grafi e disposizioni nel piano

BIBARTITO Planare

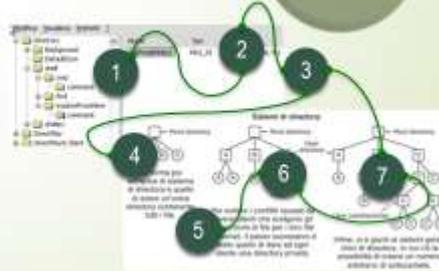
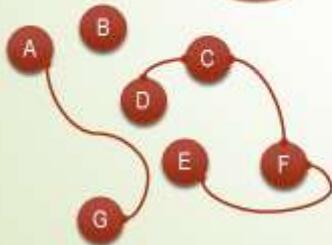
- Grafo planare
- nodi divisibili due insiemi tali per cui ogni arco ha un estremo in un insieme e un estremo nell'altro insieme

BIPARTITO Non Planare

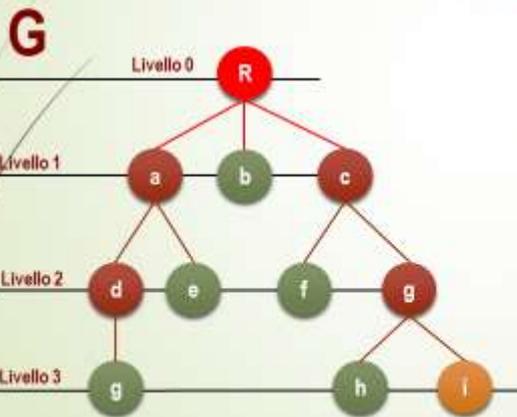
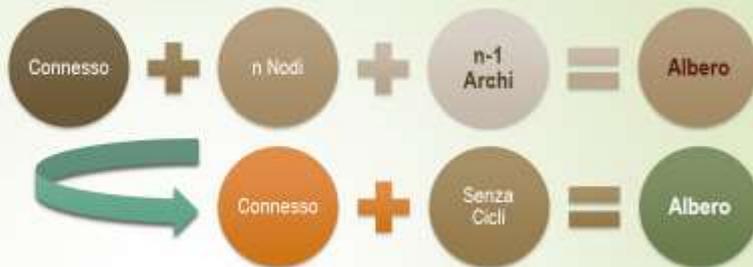
- Grafo NON planare
- nodi divisibili due insiemi tali per cui ogni arco ha un estremo in un insieme e un estremo nell'altro insieme



Grafi notevoli



Albero



NODO X ANTENATO DI Y: se X è nel cammino tra R e Y (ad esempio a e g)

- **RADICE:** nodo senza padre
- **FOGLIA:** nodo senza figli
- **GRADO:** numero massimo di figli (3 per l'albero G)
- **LIVELLO (profondità) DI UN NODO:** lunghezza del cammino da R al nodo (ad esempio d, e, f e g sono di livello 3, e R è di livello 0)
- **Altezza di un albero:** il livello (la profondità massima), 3 per il grafo G

ALBERI RADICATI: alberi in cui è definito un nodo detto RADICE

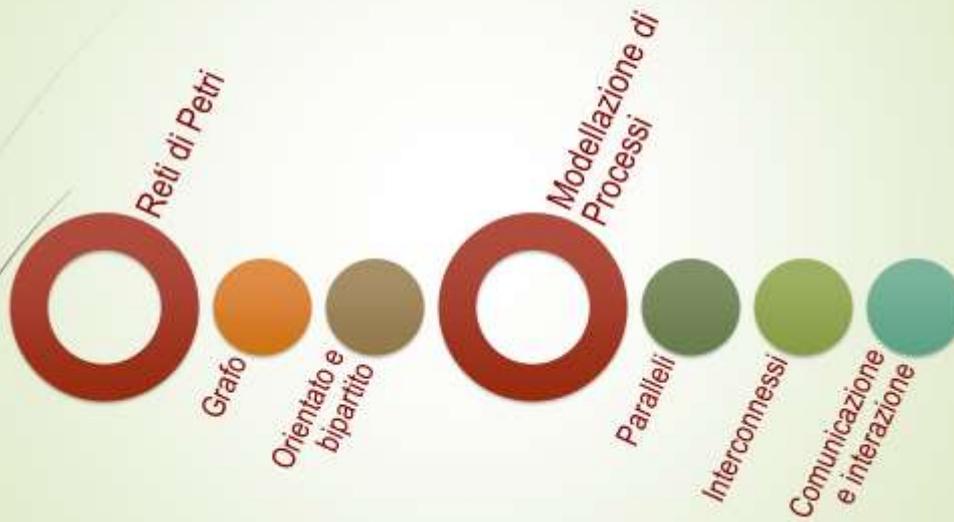
Sistemi **distribuiti** discreti



Sistemi **distribuiti** discreti



Rappresentazione di sistemi distribuiti discreti



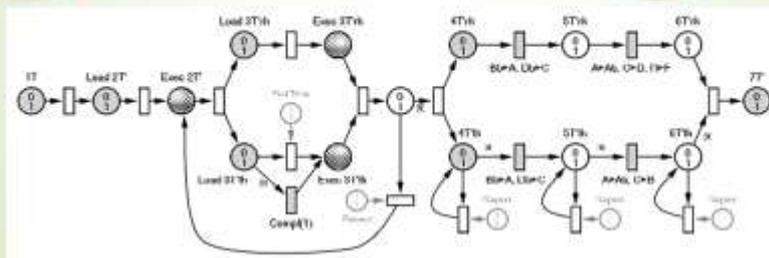
Rete di Petri (P-Rete) (1962, Carl Petri)

Processo

descritto in termini di

Elementi

Interazioni



Rete di Petri (P-Rete), Posto-Transizione (P/T): struttura

Grafo

Bipartito

Orientato

nodì Stato
(I insieme)

nodì Transizione
(II insieme)

archi con testa e coda

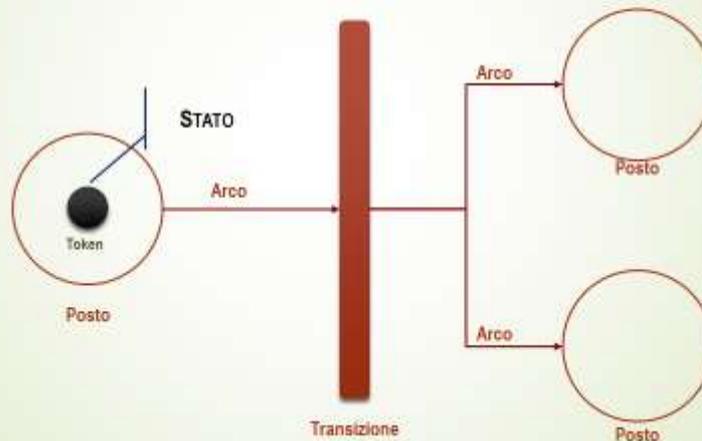
da stato a transizione

da transizione a stato

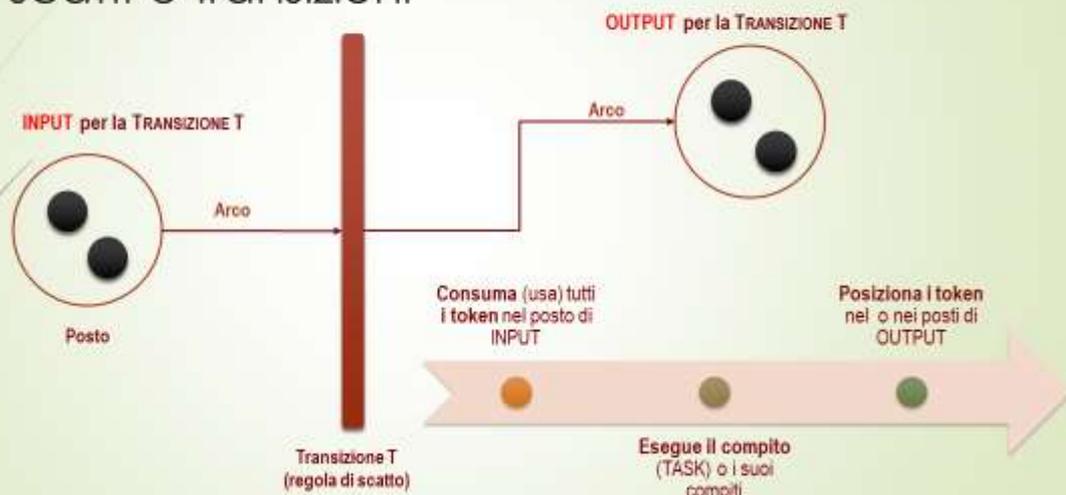
Una P-Rete è una tripla $N = (P, T, F)$

- P è un insieme dei posti
 - T è un insieme di transizioni
 - F è una relazione di flusso
- P e T sono due insiemi finiti

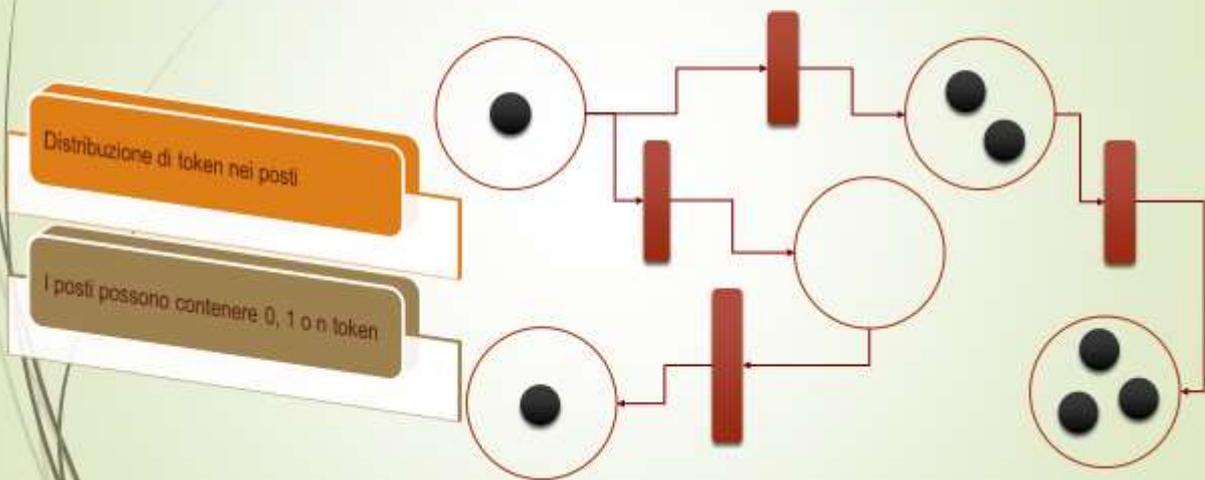
Rete di Petri (P-Rete) modello strutturale grafico



Rete di Petri (P-Rete) scatti e transizioni



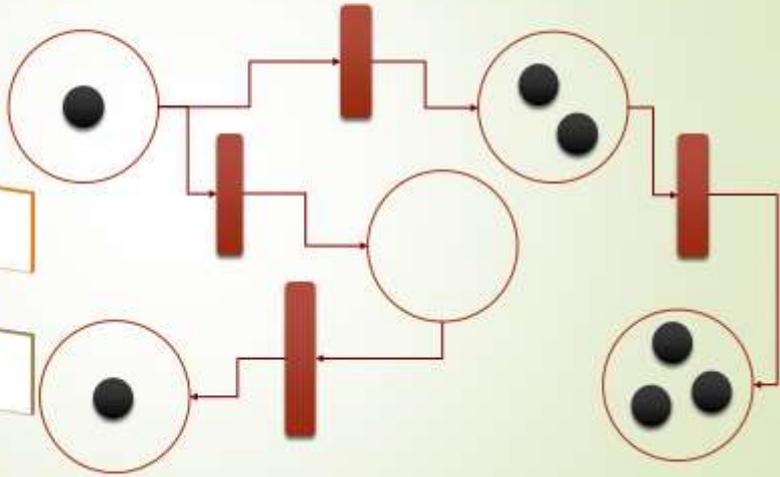
Rete di Petri (P-Rete) marcatura



Rete di Petri (P-Rete) marcatura

Distribuzione di token nei posti

I posti possono contenere 0, 1 o n token



Rete di Petri (P-Rete) schema di transizione e regole di scatto

Transizione

Regola di scatto
(Firing)

Token

Token necessari in stato di
INPUT

Transizione Abilitata
(può scattare)

Rete di Petri (P-Rete) rete NON DETERMINISTICA





Informatica

CdS in «*Scienze e Tecnologie dei Beni Culturali*» – AA 2014-2015

Mini-sito dell'insegnamento: http://www.unife.it/scienze/beni_culturali/insegnamenti/informatica

Prof. Giorgio Paletti

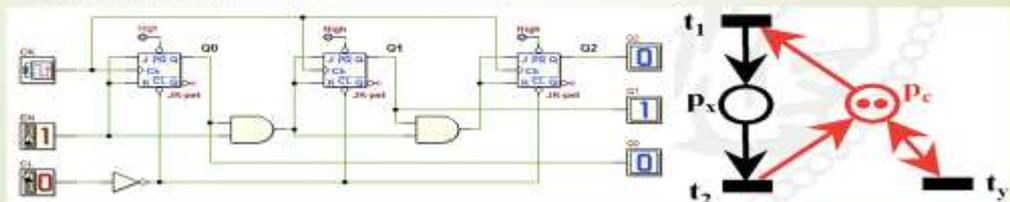
giorgio.paletti@unife.it - <http://docente.unife.it/giorgio.paletti>



Grafi e Reti

«Tutti sanno che una cosa è impossibile da realizzare, finché arriva uno sprovveduto che non lo sa e la inventa.»

(Albert Einstein)



Contenuti della lezione

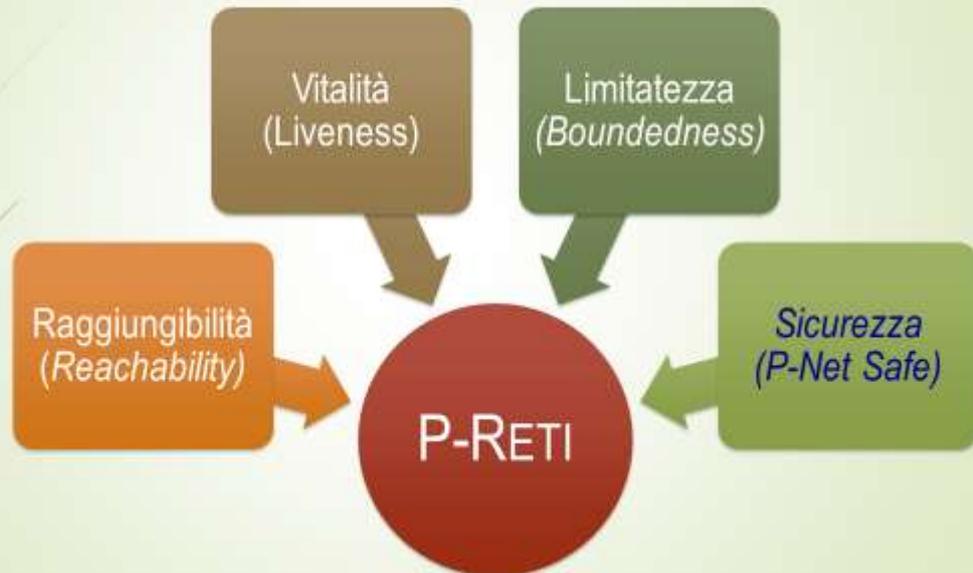
Reti di Petri

- Caratteristiche delle P-Reti
- Problemi classici
- Schema di risoluzione di un problema con le P-Reti
- Tipologie principali delle P-Reti

Reti Scale Free

- Note sulle reti a invarianza di scala
- Dalle reti casuali alle reti ad invarianza di scala

Caratterizzazione delle P-Reti



Raggiungibilità nelle P-Reti

Raggiungibilità
(Reachability)

- Data una **MARCATURA INIZIALE** M_0 in una *rete di Petri* G si indica con $R(G, M_0)$ l'insieme delle MARCATURE RAGGIUNGIBILI a partire da M_0 .



Una Marcatura M_q è **raggiungibile** se esistono scatti che la rendono una marcatura possibile a partire da M

Problema della raggiungibilità
 $M_q \in R(G, M_0)$?

La marcatura in esame appartiene all'insieme delle marcature possibili?

- Sotto quali condizioni M_q è uno stato sbagliato?
- Non può e non deve essere raggiungibile.

Esempio

PORTE APERTE E ASCENSORE NON PRESENTE

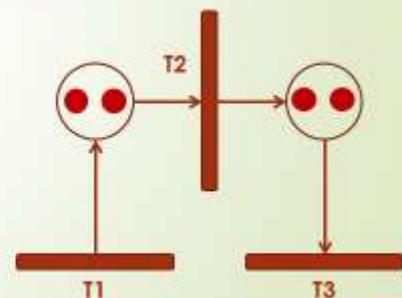
Raggiungibilità nelle P-Reti

Raggiungibilità
(Reachability)

- Data una **MARCATURA INIZIALE** M_0 in una *rete di Petri* G si indica con $R(G, M_0)$ l'insieme delle MARCATURE RAGGIUNGIBILI a partire da M_0 .

GRAFO DI RAGGIUNGIBILITÀ: grafo in cui i nodi sono le possibili marcature e gli archi le transizioni che modificano una marcatura

Buon metodo per trovare gli stati "sbagliati" ovvero che non devono essere raggiunti (...barriere alzate e treno in passaggio...)

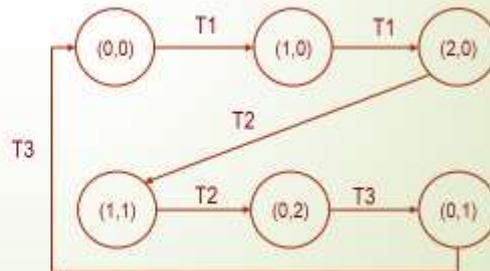
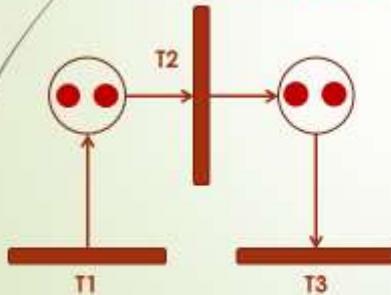


Grafo di raggiungibilità delle P-Reti

Raggiungibilità
(Reachability)

- Data una **MARCATURA INIZIALE** M_0 in una *rete di Petri* G si indica con $R(G, M_0)$ l'insieme delle **MARCATURE RAGGIUNGIBILI** a partire da M_0 .

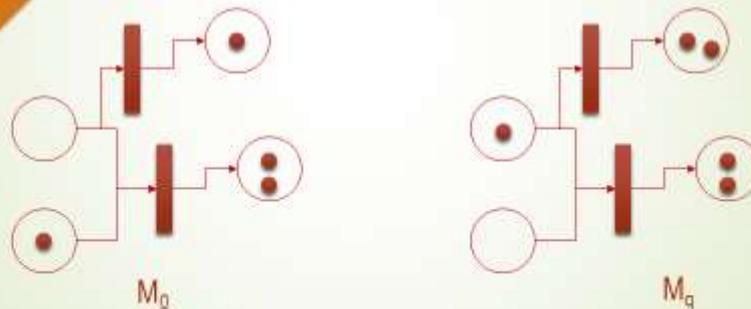
GRAFO DI RAGGIUNGIBILITÀ: grafo in cui i nodi sono le possibili marcature e gli archi le transizioni che modificano una marcatura



Limitatezza nelle P-Reti

Limitatezza
(Boundedness)

- **Posto Limitato (k-limitato):** k è il numero massimo di token nel posto per una qualsiasi marcatura possibile della rete
- **P-Rete LIMITATA** se ogni posto è limitato

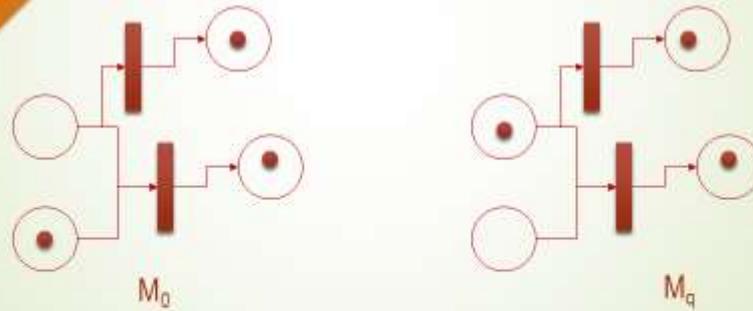


Esempio di P-Rete 2-limitata

Sicurezza nelle P-Reti

Sicurezza
(P-Net Safe)

- Una rete 1-limitata (k-limitata con $K=1$) si dice SICURA



Esempio di P-Rete 1-limitata, P-Net Safe

Vitalità nelle P-Reti

Vitalità
(Liveness)

- L è detto LIVELLO DI VITALITÀ e rappresenta l'attivabilità di una transizione T in una marcatura raggiungibile

La rete è viva se...

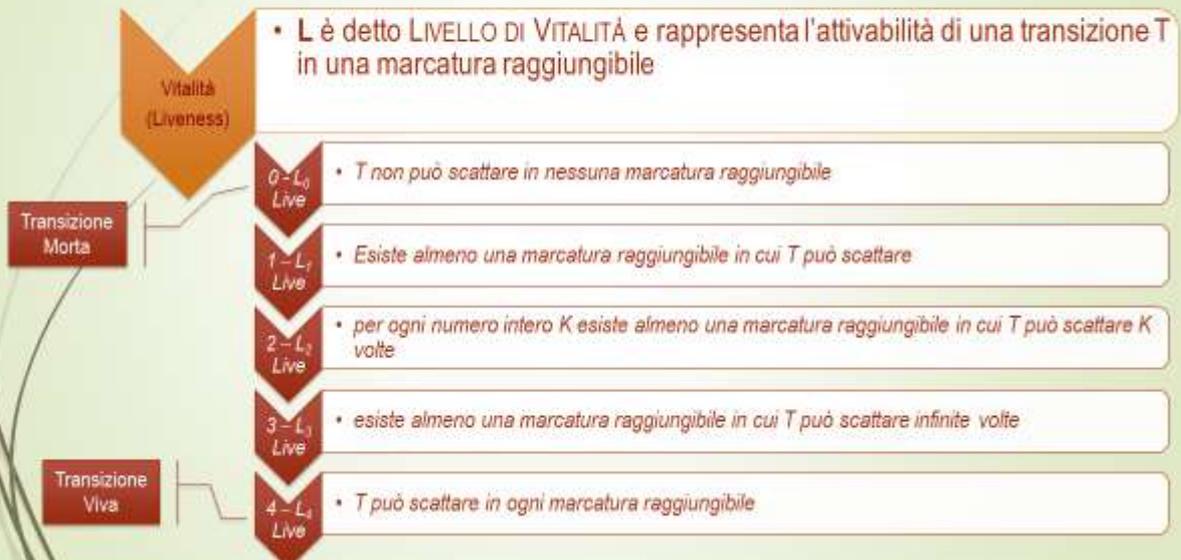
...è sempre possibile far scattare una transizione T qualsiasi a seguito di una qualsiasi sequenza di scatti

...detta M_j marcatura qualsiasi raggiungibile da M_0

P-Rete è k -LIVE

ogni transizione T è k -Live

Vitalità nelle P-Reti



Problemi classici per P-Reti



Problema dei 5 filosofi affamati

(dining philosophers problem, Dijkstra)

Schema per problemi di concorrenza e condivisione di risorse



Problema del barbiere che dorme

Schematizza problemi analoghi a quelli di un help desk informatizzato

Problema del barbiere che dorme



Problema del barbiere che dorme

(Schematizza problemi analoghi a quelli di un help desk informatizzato)

Un barbiere possiede un negozio con una sola sedia da lavoro e un certo numero limitato di posti per attendere. Se non ci sono clienti il barbiere dorme altrimenti, all'arrivo del primo cliente il barbiere si sveglia ed inizia a servirlo. Se dovessero sopraggiungere clienti durante il periodo di attività del barbiere, essi si mettono in attesa sui posti disponibili. Al termine dei posti di attesa, un ulteriore cliente viene scartato.

Una corretta programmazione concorrente deve far "dormire" il barbiere in assenza di clienti, attivare il barbiere sul primo cliente al suo arrivo e mettere in coda tutti i successivi clienti tenendoli inattivi.

Problema dei 5 filosofi affamati



Problema della cena dei 5 filosofi affamati

(Dining philosophers problem, Dijkstra)

Schematizza problemi problemi di concorrenza e condivisione di risorse

1. Un filosofo può prendere solo le due bacchette che stanno alla sua destra e alla sua sinistra, una per volta, e solo se sono libere, non può sottrarre la risorsa bacchetta ad un altro filosofo che l'ha già acquisita e sta mangiando (no preemption, non c'è predominanza).

2. Finché non riesce a prendere le bacchette, il filosofo deve aspettare affamato. Quando invece è sazio posa le bacchette al loro posto e si mette a pensare per un certo tempo.

*Una corretta programmazione concorrente deve essere in grado di far mangiare alternativamente tutti i filosofi evitando che qualcuno in particolare soffra di **starvation** ed evitando che si verifichino stalli in fase di "acquisizione delle bacchette".*

STARVATION, ("inedia") in informatica lo stato di un processo pronto per essere eseguito ma che non riesce ad ottenere le risorse di cui necessita

Problema dei 2 filosofi affamati



Problema della cena dei 2 filosofi affamati

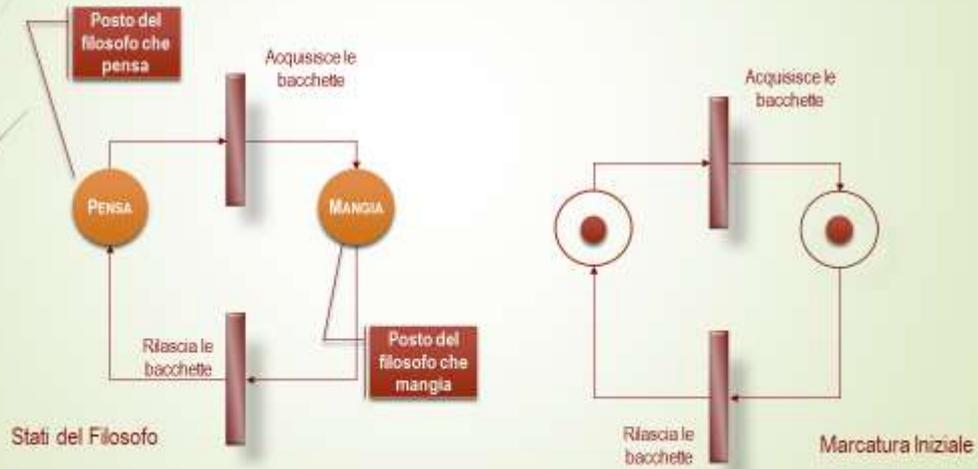
(Dining philosophers problem, Dijkstra)

1. Nietzsche ed Eraclito mangiano spesso assieme
2. Siedono attorno ad un tavolo rotondo e hanno, ognuno, a disposizione un piatto di cibo e due singole bacchette sono collocate ai lati dei loro piatti
3. Sempre, o pensano o mangiano
4. Quando uno dei due comincia ad avere fame cerca di prendere possesso delle bacchette alla sua destra e alla sua sinistra, in ordine arbitrario
5. Qualora riesca a prenderle entrambe, mangia per un po'. Successivamente depone le bacchette e si rimette a pensare
6. Nessuno dei due è in grado di mangiare con una singola bacchetta o con le mani
7. Il problema consiste nel far in modo che entrambi i filosofi riescano a cibarsi e pensare periodicamente

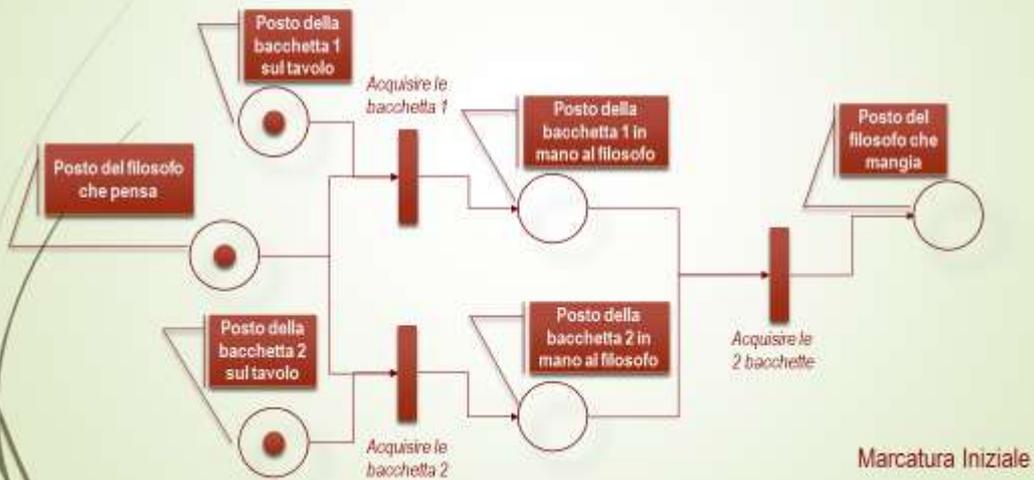
Problema dei 2 filosofi affamati Il schema



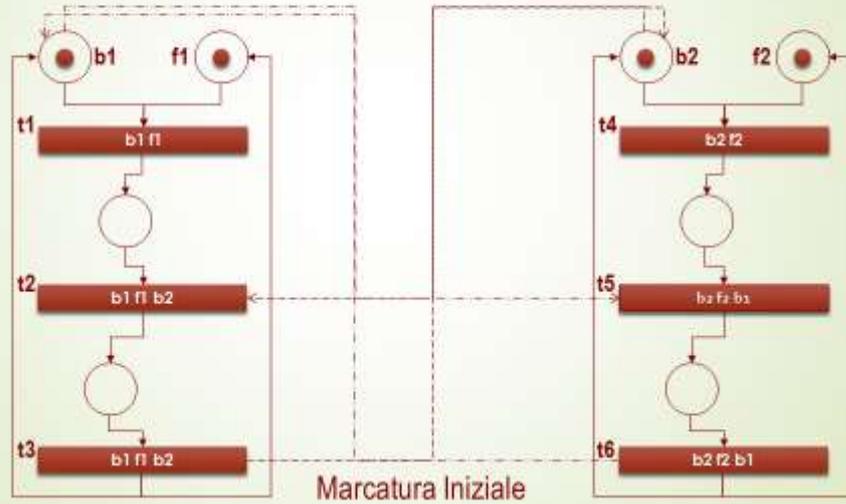
Problema dei 2 filosofi affamati Il schema, logico per un filosofo



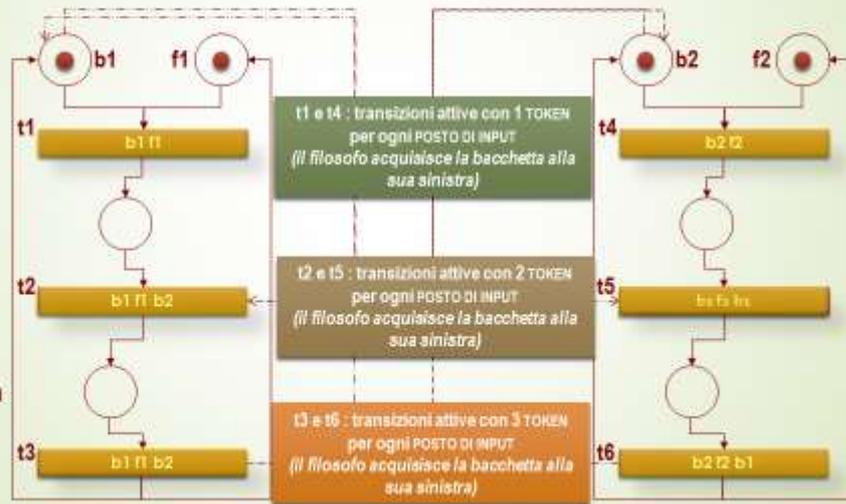
Problema dei 2 filosofi affamati Il schema, logico per un filosofo



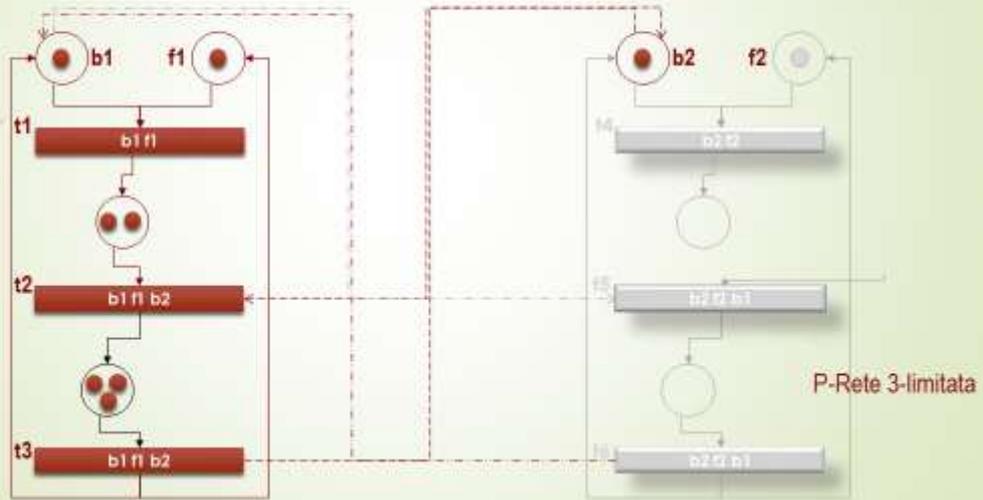
Problema dei 2 filosofi affamati Schema logico del problema



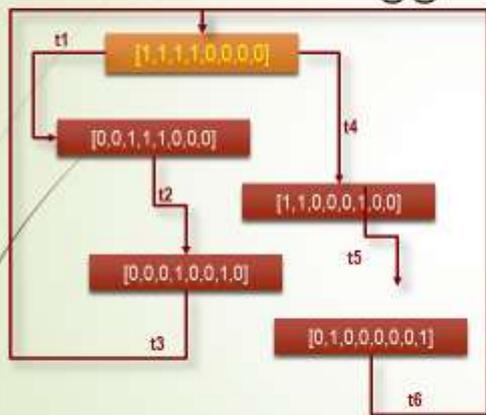
Problema dei 2 filosofi affamati Schema logico del problema



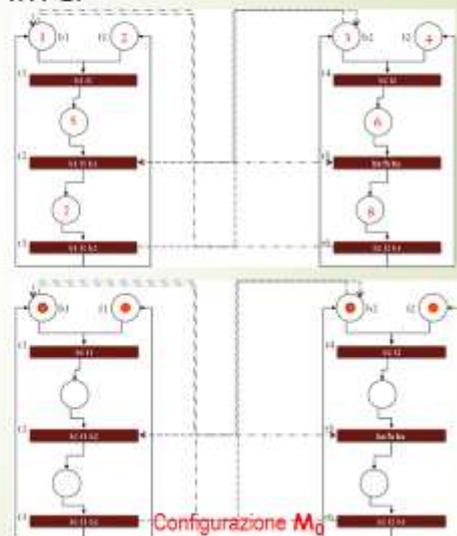
Problema dei 2 filosofi affamati Schema funzionamento per 1 filosofo



Problema dei 2 filosofi affamati Grafo di raggiungibilità



MARCATURA, un insieme di 8 valori, in ordine, 1 se ci sono TOKEN 0 se non ci sono nello stato i .
6 marcature possibili



Costrutti nelle P-Reti



Concorrenza



Conflitto



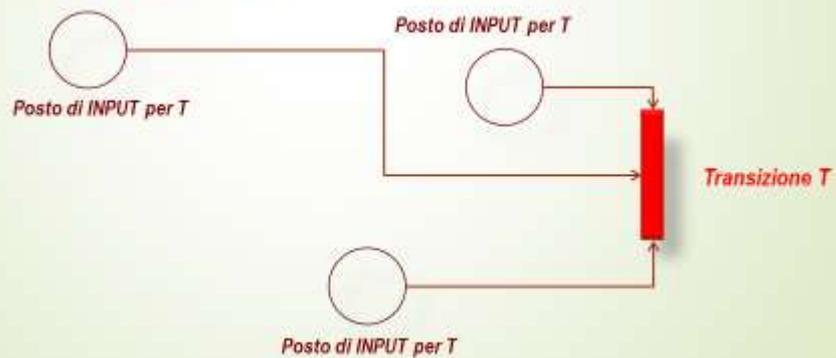
Confusione

Concorrenza nelle P-Reti



Concorrenza

Una TRANSIZIONE ha più POSTI di INPUT



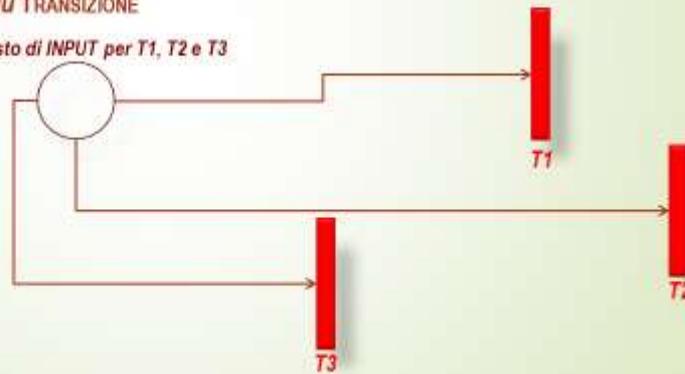
Conflitto nelle P-Reti



Conflitto

Una POSTO è INPUT per più TRANSIZIONE

Posto di INPUT per T1, T2 e T3



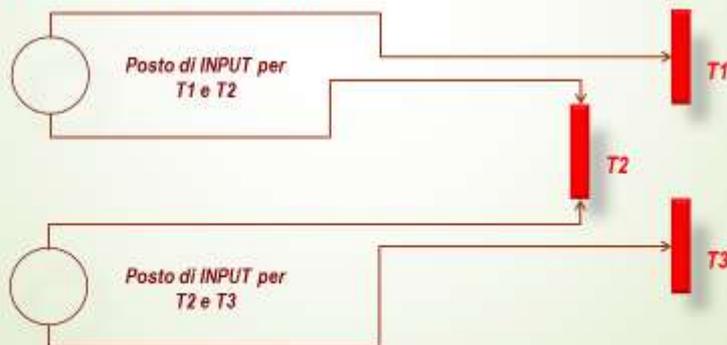
Confusione nelle P-Reti



Confusione

Unione di CONCORRENZA e CONFLITTO

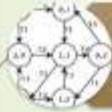
Posto di INPUT per T1 e T2



P-Reti: Tipologie notevoli



Macchine a Stati Finiti (SM: State Machine)



Grafo Marcato (MG: Marked Graph)



Scelta Libera (FC: Free Choice)



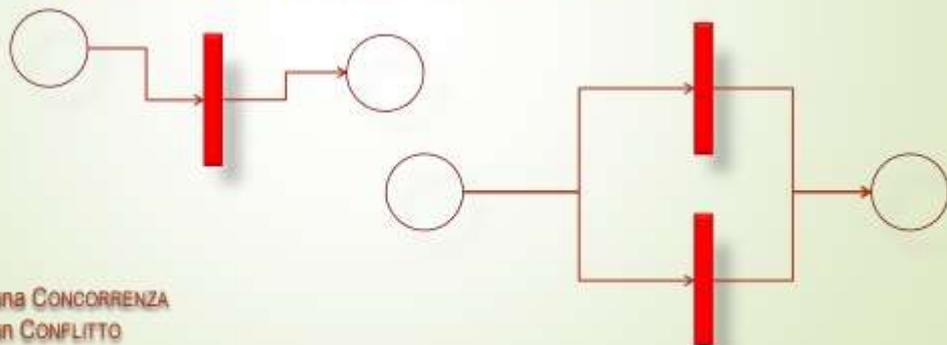
Scelta Asimmetrica (AC: Asymmetric Choice)

P-Reti: Tipologie notevoli



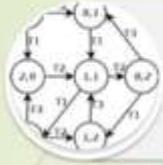
Macchine a Stati Finiti (SM: State Machine)

Ogni TRANSIZIONI ha un solo arco in entrata e uno solo in uscita



- Nessuna CONCORRENZA
- Nessun CONFLITTO

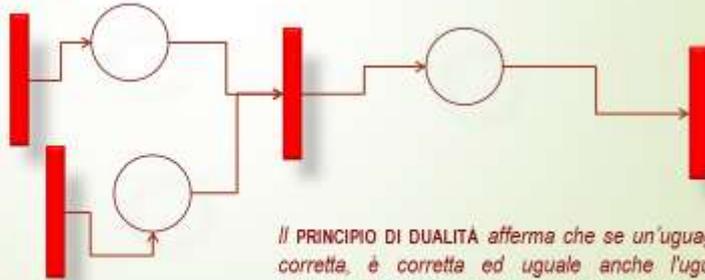
P-Reti: Tipologie notevoli



Grafo Marcato (MG: Marked Graph)

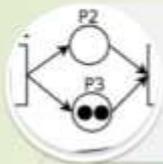
Ogni **POSTO** ha un solo arco di entrata e uno solo di uscita
(Duale della Macchina a Stati Finiti)

- **Possibili Concorrenze**
- **Nessun Conflitto**



Il PRINCIPIO DI DUALITÀ afferma che se un'uguaglianza è corretta, è corretta ed uguale anche l'uguaglianza ottenuta sostituendo da tutte e due le parti 1 con 0 e 0 con 1.

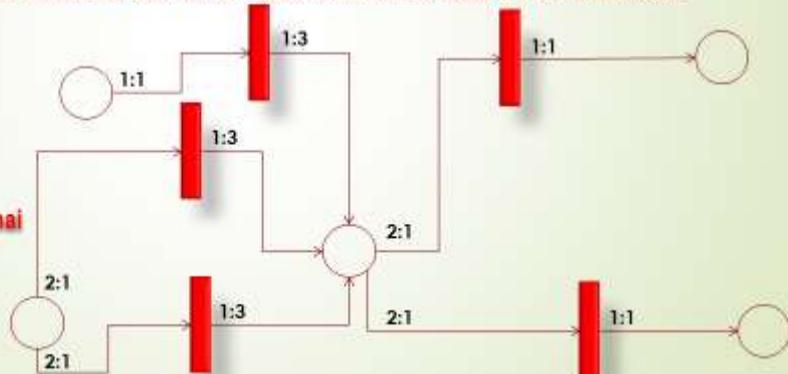
P-Reti: Tipologie notevoli



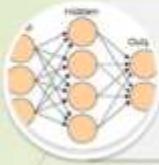
Scelta Libera (FC: Free Choice)

Ogni **ARCO** è o l'unico che esce da un **POSTO** o l'unico che entra in una **TRANSIZIONE**.

- **Possibili Concorrenze**
- **Possibili Conflitti**
- **Conflitti e Concorrenze mai simultaneamente**



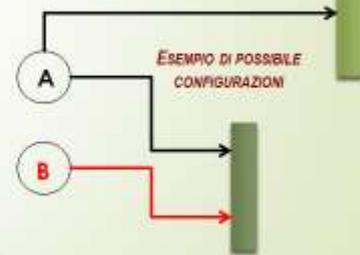
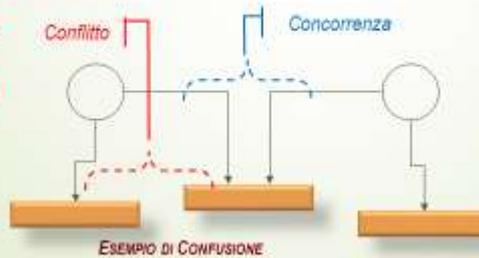
P-Reti: Tipologie notevoli



Scelta Asimmetrica (AC: Asymmetric Choice)

Se 2 Posti (A e B) sono POSTI di INPUT per una stessa TRANSIZIONE, l'insieme delle TRANSIZIONI per cui A è POSTO di INPUT contiene le TRANSIZIONI per cui B è POSTO di INPUT

- Possibili Concorrenze
- Possibili Conflitti
- **Conflitti + Concorrenze (Confusione MAI ASIMMETRICAMENTE)**



Contenuti della lezione

■ Iper testo

- Il concetto di Iper testo
- Storia dell'iper testo
- Evoluzione tecnologica dell'iper testo
- Internet come Iper testo

■ Internet e il WEB

- Il WEB 2.0
- Evoluzione del Web
- Interoperabilità, strumenti e filosofia del WEB 2.0

Schema generale



Storia dell'Iper testo

1531

Agostino Ramelli

Bookwheel

«Le diverse et artificiose machine»



1945

Vannevar Bush

As We May Think (Come potremmo pensare), articolo in Atlantic Monthly, luglio 1945, riflessione sul sistema ipertestuale.

MEMEX scrivania analogica con sistema di archiviazione



1960

Ted Nelson

Progetto Xanadu, primo progetto di ipertesto, rete di computer con interfaccia utente semplice.

«Un'interfaccia utente dovrebbe essere così semplice da poter essere compresa, in caso di emergenza, nel giro di dieci secondi da un principiante»

1. Computer Lib/Dream Machines (1974)
2. Literary Machines (1981)



Storia dell'Iper testo

1960

Ted Nelson

Douglas Engelbart

Studiano e producono testi con struttura ispirata ai concetti di Bush

Engelbart inventa il **MOUSE** (con William 'Bill' K. English)

9 dicembre 1968, San

Francisco, prima presentazione del mouse per computer



1965

Ted Nelson

Introduce per la prima volta il termine **hypertext** (testo multidimensionale, dal prefisso greco «*υπερ*», sopra o oltre), come «qualsiasi sistema di scrittura non lineare che implichi applicazioni informatiche»



1987

Bill Atkinson

Nascita di **HyperTalk**, realizzato alla Apple.

«Sistema di gestione di grandi masse di testo e immagini, con un linguaggio di programmazione ipertestuale»



Storia dell'Ipertesto

1980-1989

Tim Berners-Lee
Robert Cailliau

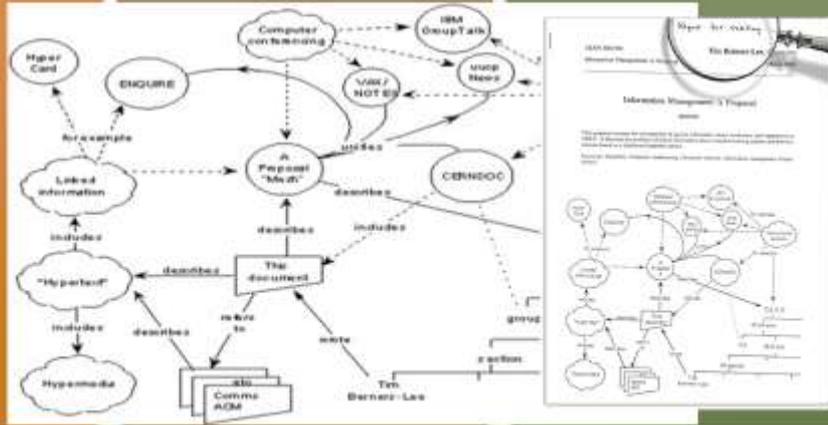
CERN, si implementa un software Enquire, base concettuale del futuro World Wide Web

1987

Nasce al MIT il W3C

1993

Intervista TG1



Storia dell'Ipertesto

6 agosto 1991

Tim Berners-Lee

On line il primo sito
<http://info.cern.ch/>



1993-1994

Aumentano i Server

Server noti
500 → 10.000



2012

Server nell'era moderna

Circa 908.586.000 server
Google usa almeno 1.000.000 server



Web e Web Semantico

PAROLE CHIAVE

CONDIVISIONE

COLLABORAZIONE

INTERAZIONE
(utenti, piattaforme
e tecnologie)

- Visualizzare e navigare DOCUMENTI ipertestuali STATICI (HTML)

WEB 1.0

WEB 1.5

- Visualizzare e navigare DOCUMENTI ipertestuali DINAMICI (DB e CMS, CSS e linguaggi di SCRIPT)

- Siti come contenitori di SERVIZI e APPLICAZIONI

WEB 2.0

MONODIREZIONALE

BIDIREZIONALE

Web e Web Semantico - Architettura

WEB 1.0
Monodirezionale

Client/Server
(Siti)

Browsing

Click & Wait

WEB 2.0
Bidirezionale

Orientato ai servizi
(Broker)

Applicazioni

Asincronia
(interazione senza ricaricamento totale)

Web e Web Semantico, il 2.0



2.0 ...nuovo modo di intendere la Rete: pone al centro i contenuti, le informazioni, l'interazione.
(Tim O'Reilly - Editore)



2.0 ... nessun vero significato: ognuno in funzione delle tecnologie che vuole proporre ne determina uno (Tim Berners-Lee)



Tim Berners-Lee

WEB semantico, una visione USER-CENTRED

Tim O'Reilly

Web e Web Semantico

Web 2.0: le 4C di Andrew Lih (Hong Kong University)



Web e Web Semantico

Web 2.0: le 4C di Andrew Lih (Hong Kong University)

Collective Intelligence

 L'Intelligence collective. Pour une anthropologie du cyberspace
• Pierre Lévy (1994)

 Global Brain: The Evolution of Mass Mind from the Big Bang to the 21st Century
• Howard Bloom (2001)

 Smart Mobs: The Next Social Revolution
• Howard Rheingold (2003)

La capacità di una comunità umana di evolvere verso una capacità superiore di risolvere problemi, di pensiero e di integrazione attraverso la COLLABORAZIONE e l'INNOVAZIONE

George Pôr

The Quest for Cognitive Intelligence (1995)

Web e Web Semantico

Web 2.0: le 4C di Andrew Lih (Hong Kong University)



Copyright
Copyleft


CopyLEFT
Modello alternativo della gestione dei diritti


CopyRIGHT
Diritto di copia



- GNU LGPL (GNU Lesser General Public License)
- GNU GPL (GNU General Public License)

GNU is Not Unix, 1983 (<http://www.gnu.org>)

Richard Matthew Stallman

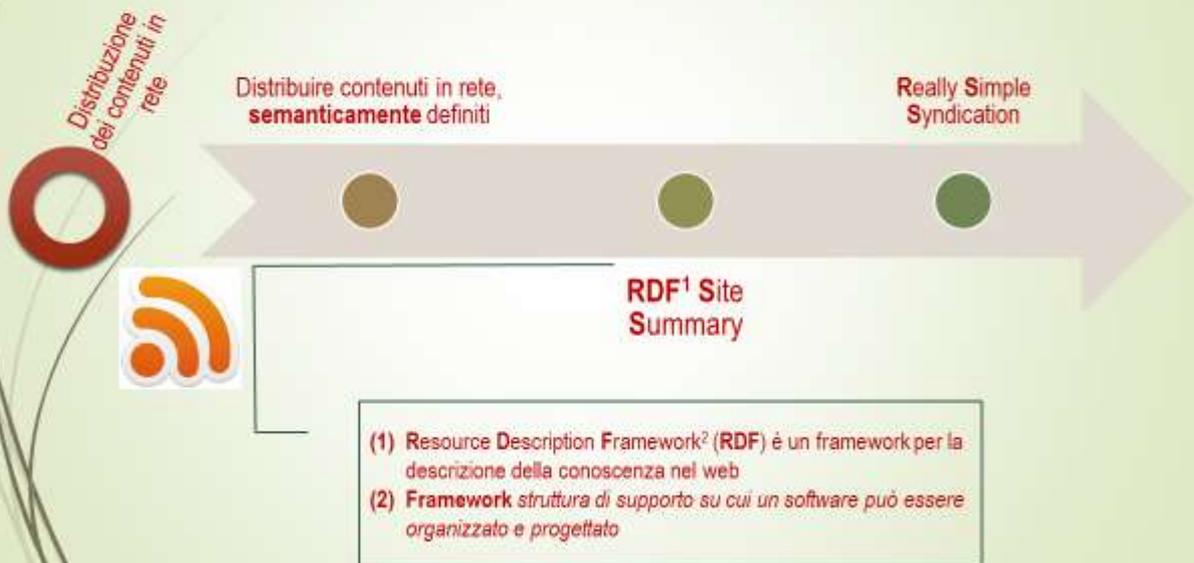


WEB: **Indipendenza** e Interoperabilità



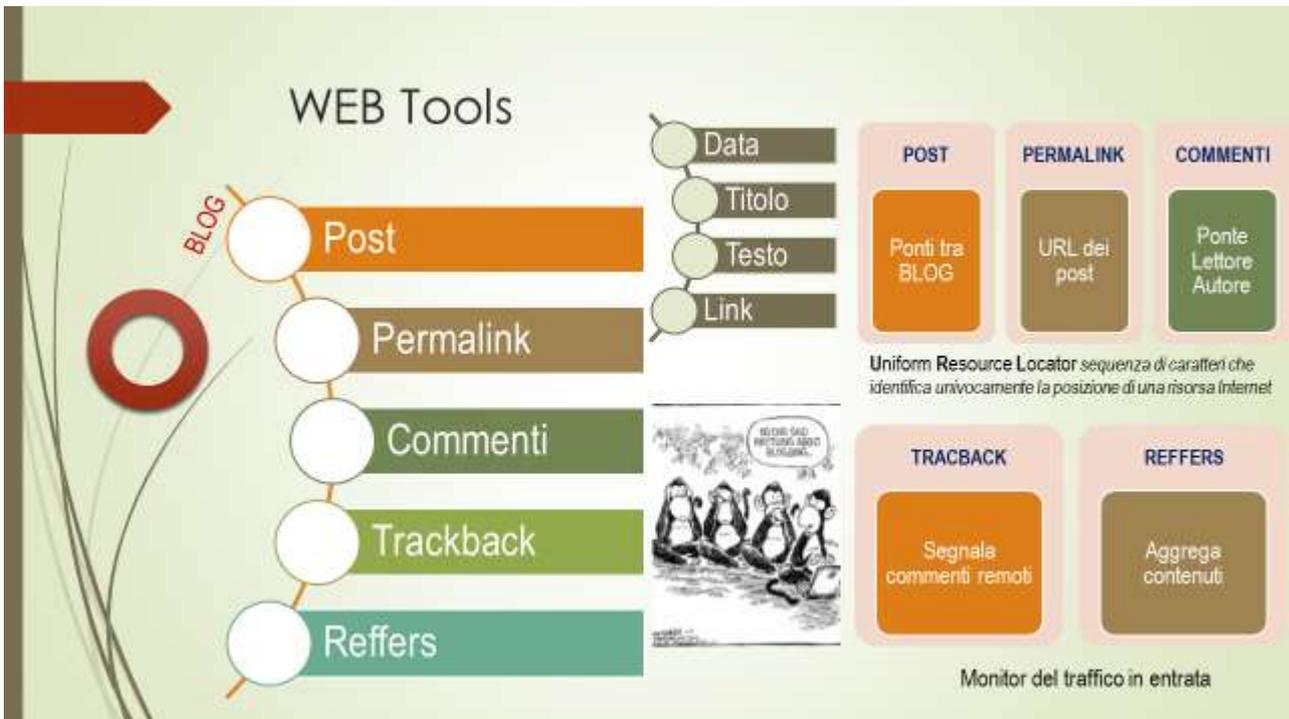
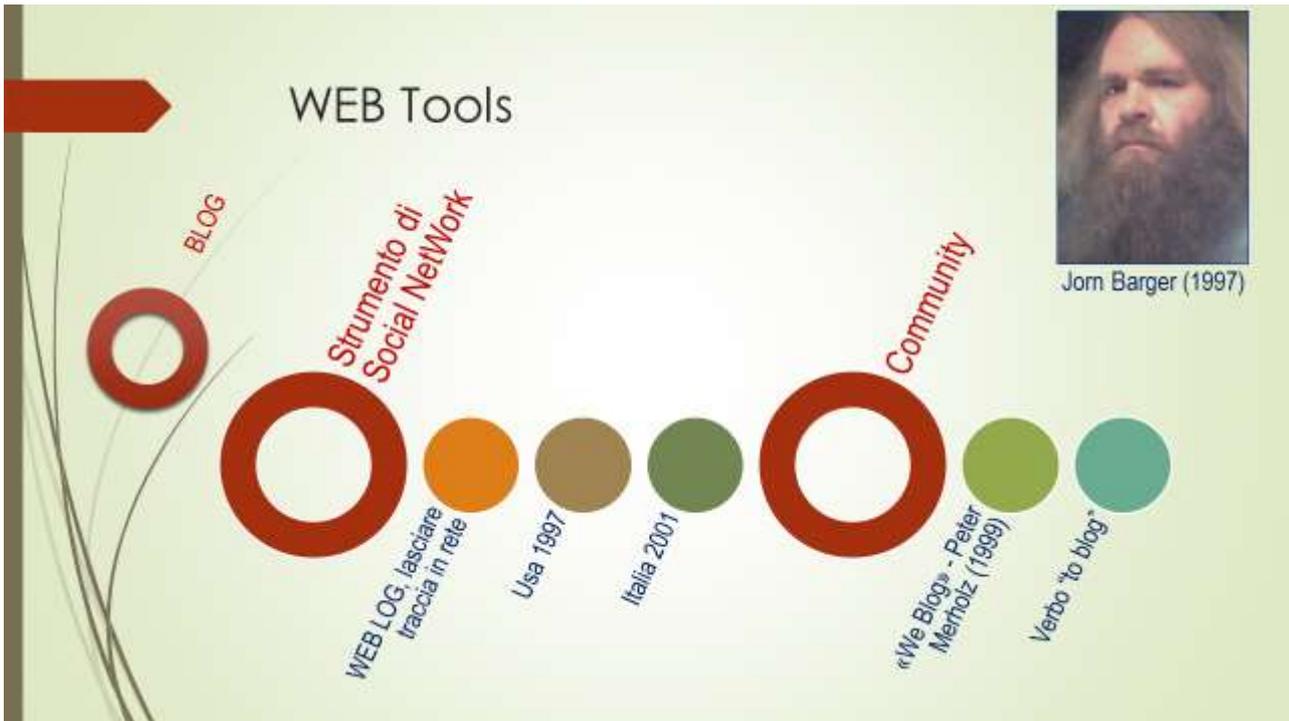
XML (eXtensible Markup Language) metalinguaggio creato e gestito dal World Wide Web Consortium (W3C), direttamente dal presidente del "consorzio", **Michael Sunshine**. È una semplificazione e adattamento dell' SGML, da cui è nato nel 1998, e permette di definire la grammatica di diversi linguaggi specifici derivati.

WEB: Indipendenza e **Interoperabilità**



- (1) **Resource Description Framework² (RDF)** è un framework per la descrizione della conoscenza nel web
- (2) **Framework** struttura di supporto su cui un software può essere organizzato e progettato





WEB

WEB 2.0



WEB Tools

PODCAST
VIDEOCAST

PODCAST

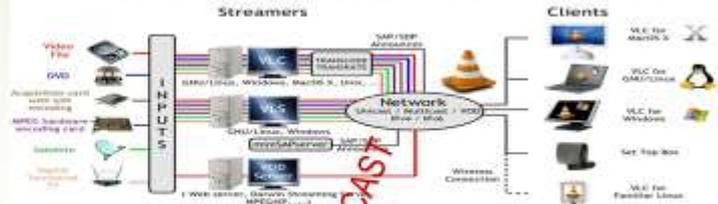
VIDEOCAST

iPod + Broadcasting

RSS

Video On-Demand

VideoLAN Streaming Solution



BROADCASTING: trasmissione di informazioni da un sistema trasmittente ad un insieme di sistemi riceventi non definito a priori

