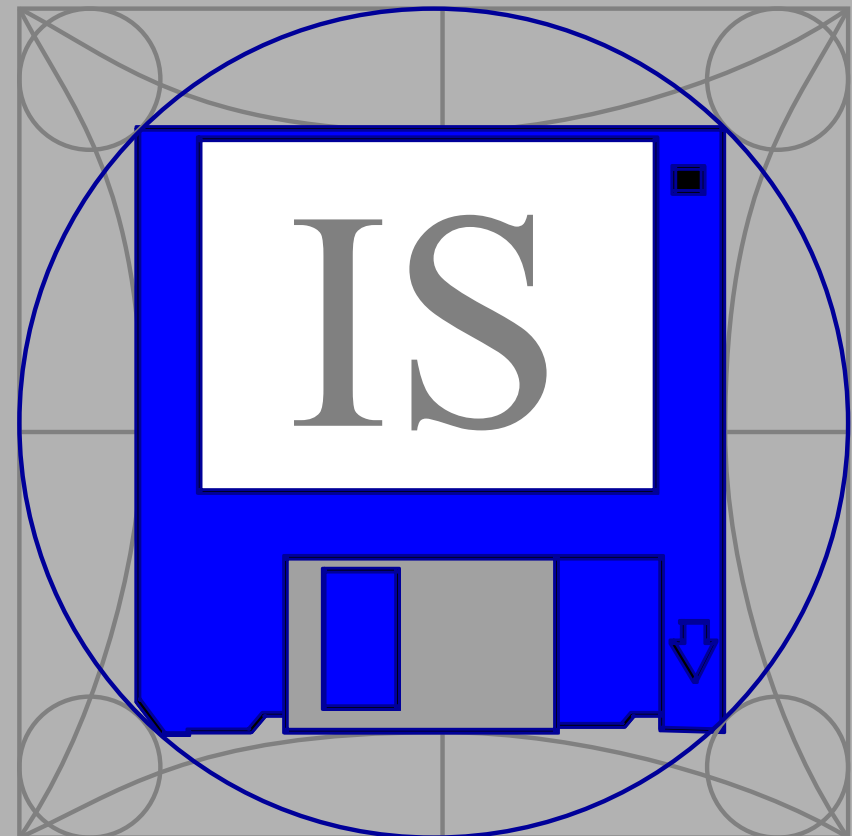


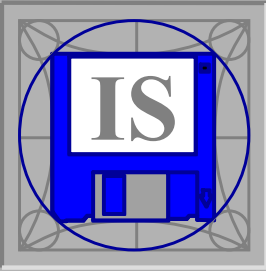
Gestione di progetto

Ingegneria del Software

V. Ambriola, G.A. Cignoni,
C. Montangero, L. Semini

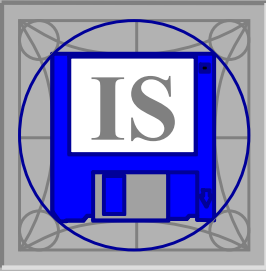
Aggiornamenti: T. Vardanega (UniPD)





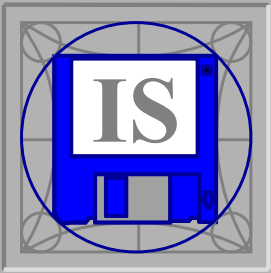
Cosa è un progetto?

- ❑ Due definizioni complementari: Kerzner, SEMAT
- ❑ Una terza origina dalla relazione con il ciclo di vita del SW
- ❑ Progetto è un insieme ordinato di attività, ciascuna delle quali istanziate da processi di ciclo di vita
 - Le attività sono fatte di compiti assegnati a singoli individui
 - Le attività sono pianificate prima di essere svolte
 - Ogni attività ha specifici obiettivi e vincoli che derivano dal processo di appartenenza
 - Ogni singola attività di progetto deve ricercare economicità
- ❑ Il progetto nel suo complesso è sempre collaborativo

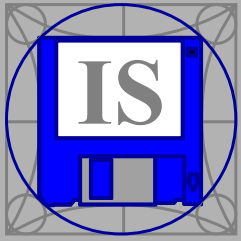


Fondamenti di gestione – 1/2

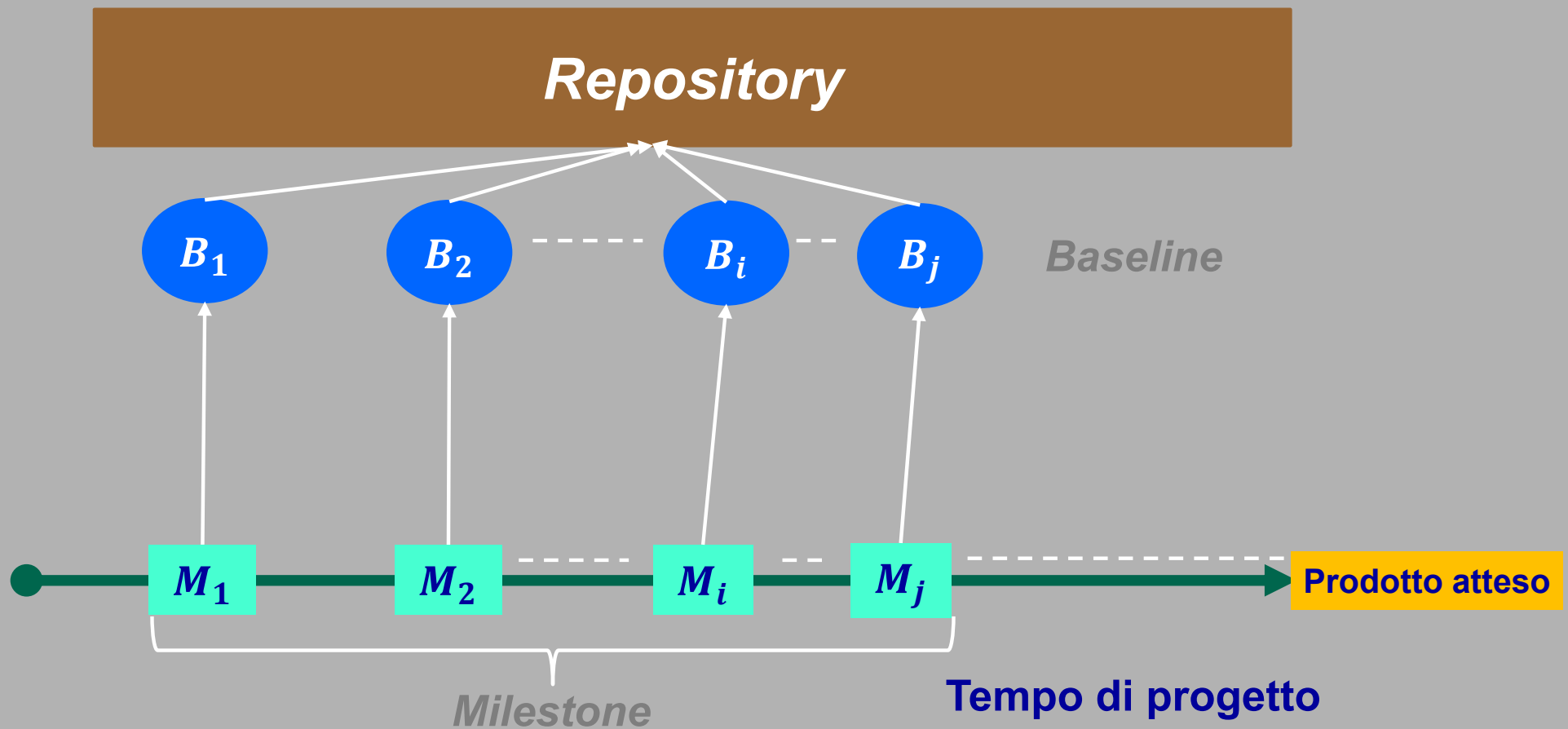
- ❑ Stabilire il proprio *way of working*
 - Adattando processi di ciclo di vita alle proprie necessità
 - Istanziando quei processi in attività di progetto
- ❑ Determinare le risorse disponibili (ore-persona, calendario)
- ❑ Fissare gli obiettivi di avanzamento
 - In una successione di **milestone**, da quella finale all'indietro
 - Orientando le attività al raggiungimento di quegli obiettivi
- ❑ Determinare le risorse necessarie per svolgere quelle attività
 - Questo si chiama «**preventivo**»
- ❑ Adattare gli obiettivi alle disponibilità effettive

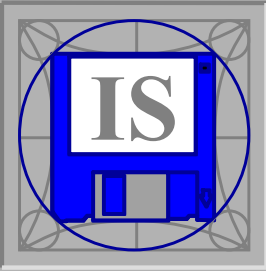


- ❑ Una **milestone** è una data di calendario che fissa un punto di avanzamento atteso
 - Naturale strumento di pianificazione preventiva e consuntiva
- ❑ Il raggiungimento di quegli obiettivi di avanzamento è sostanziato da una **baseline**
- ❑ Una **baseline** è la versione approvata di un prodotto di lavoro (parte di un progetto) che può essere modificato solo attraverso procedure formali di controllo delle modifiche
 - Ogni **baseline** è soggetta a controllo di versione e di configurazione
- ❑ Quali sono le **baseline** minime nel progetto didattico di IS?
- ❑ Il prodotto di progetto è un aggregato di SW e di documentazione!



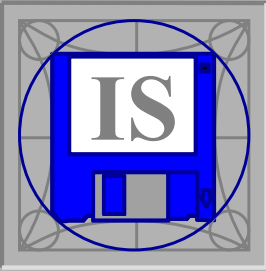
Milestone e baseline





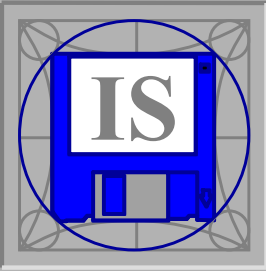
Fondamenti di gestione – 2/2

- ❑ Controllare l'avanzamento con frequenza e regolarità
 - In modalità *push*: il completamento dell'azione causa notifica
 - Invece che *pull*: «chiedo per sapere»
 - Per risultati: **baseline** che raggiungono **milestone**
 - Per costi sostenuti: **consuntivo di periodo**
- ❑ Aggiornare la pianificazione futura in funzione dell'avanzamento rilevato (**preventivo a finire**)
- ❑ Usare la tecnica agile della «**retrospettiva**»
 - Cosa abbiamo imparato nel periodo precedente?
 - Cosa significa questo per il futuro?



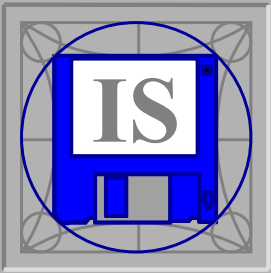
Buone qualità di *milestone*

1. **Specifiche per obiettivi di avanzamento, e dimostrabile per risultati attesi**
 - In una successione naturalmente incrementale
2. **Obiettivi coerenti con la strategia di progetto**
 - Significativi per il *team* e per gli *stakeholder*
3. **Obiettivi tempestivi e coerenti con le esigenze di calendario**
4. **Incrementi delimitati per ampiezza e ambizioni**
 - Realisticamente raggiungibili
5. **Misurabili per quantità di impegno necessario**
6. **Traducibili in compiti assegnabili a singoli individui**
 - Corrispondenti a uno *sprint* di metodo agile



Ruoli e funzioni di progetto

- ❑ Le organizzazioni specializzano il proprio personale per **funzioni**
 - Direzione, Amministrazione finanziaria, Sviluppo, «Sysadmin», Controllo di qualità, ...
- ❑ In un progetto, le persone assumono **ruoli**
 - Ogni ruolo ha responsabilità (*ownership*) su specifiche attività di specifici processi
- ❑ I gruppi di progetto didattico sono organizzazioni temporanee, nei cui ruoli i membri ruotano, per ragioni formative

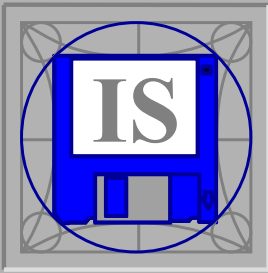


□ Analisti

- **Conoscono il dominio del problema e hanno esperienza professionale**
- **Hanno molta influenza sul successo del progetto**
- **Sono pochi: non seguono il progetto fino alla consegna**

□ Progettisti

- **Hanno competenze tecniche e tecnologiche aggiornate**
- **Determinano le scelte realizzative**
- **Sono pochi: seguono lo sviluppo, non la manutenzione**

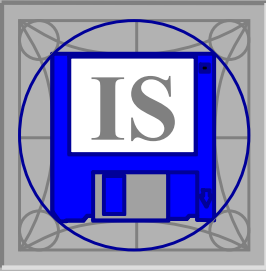


□ Programmatori

- Contribuiscono alla realizzazione e manutenzione del prodotto
- Hanno competenze tecniche ma deleghe limitate
- Formano la categoria più popolosa

□ Verificatori

- Sono presenti per l'intera durata del progetto
- Hanno competenze tecniche, esperienza professionale, conoscenza del *way of working*
- Hanno capacità di giudizio e di relazione



□ Responsabile (*project manager*)

○ Governa il *team* e rappresenta il progetto verso l'esterno (livello *customer*)

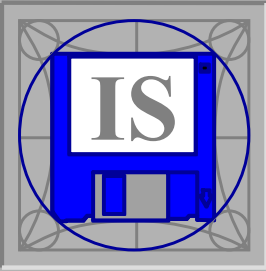
- Accentra le responsabilità di scelta e approvazione
- Partecipa al progetto per tutta la sua durata

○ Ha responsabilità su

- Pianificazione e gestione delle risorse
- Controllo, coordinamento e relazioni esterne

○ Deve avere conoscenze e capacità tecniche

- Per valutare rischi, scelte, alternative



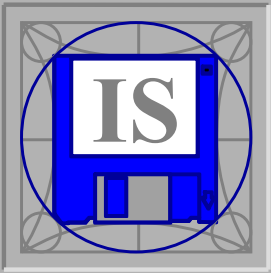
□ **Amministratore di sistema (*sysadmin*)**

○ **Definisce, controlla, e manutiene l'ambiente IT di lavoro**

- Selezione e messa in opera di risorse informatiche a supporto del *way of working*
- Azione proattiva meglio che reattiva
- Gestione delle segnalazioni (*ticket*) su non-funzionamento dell'infrastruttura

○ **Funzione o ruolo?**

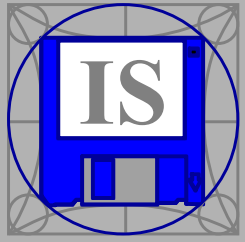
- Funzione aziendale in organizzazioni strutturate, con più progetti simili (per ragioni di standardizzazione)
- Altrimenti ruolo di progetto



Gestione qualità

- ❑ La funzione di più recente introduzione
 - Funzione aziendale, non ruolo di progetto
- ❑ La **qualità** ha più dimensioni
 - Riguarda sia i prodotti che i processi
 - Interessa sia il committente che la direzione aziendale
- ❑ La garanzia di qualità produce confidenza
 - Richiede applicazione rigorosa dei processi adottati
 - E loro manutenzione migliorativa → ciclo PDCA

Di questo parleremo
ampiamente più avanti



Pianificazione di progetto – 1/2

□ Definizione delle attività

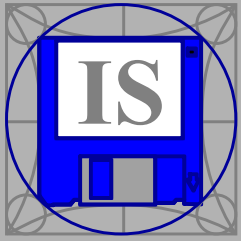
- Per pianificarne lo svolgimento e valutare il progresso
- Per avere una base su cui gestire l'allocazione delle risorse
- Per stimare e controllare scadenze e costi

□ Strumenti per la pianificazione

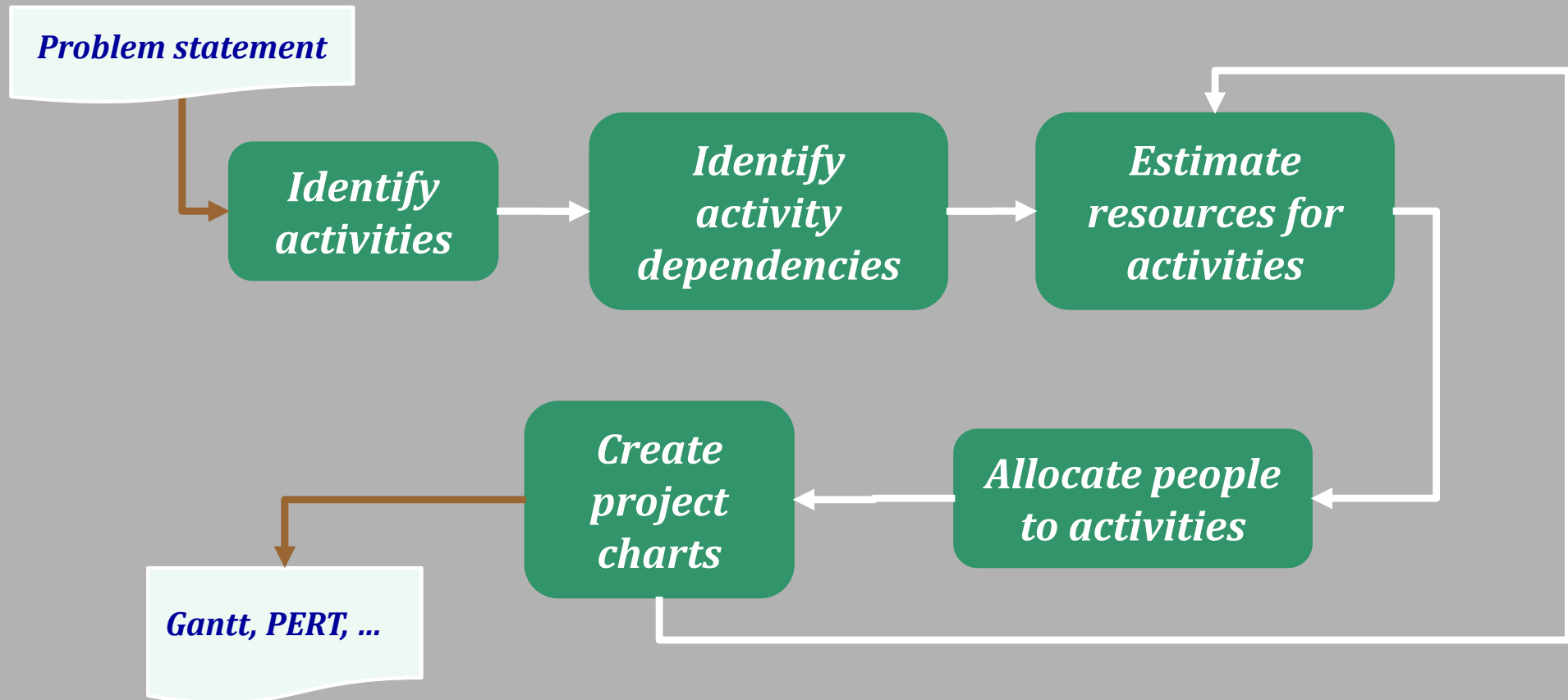
○ Diagrammi di Gantt

- ("Work, Wages and Profit", Henry L. Gantt, The Engineering Magazine, NY, 1910)

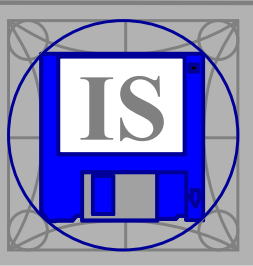
○ *Programme Evaluation and Review Technique* (PERT)



Pianificazione di progetto – 2/2

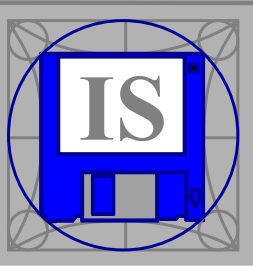


Tratto da: Ian Sommerville, *Software Engineering*, 8th ed.



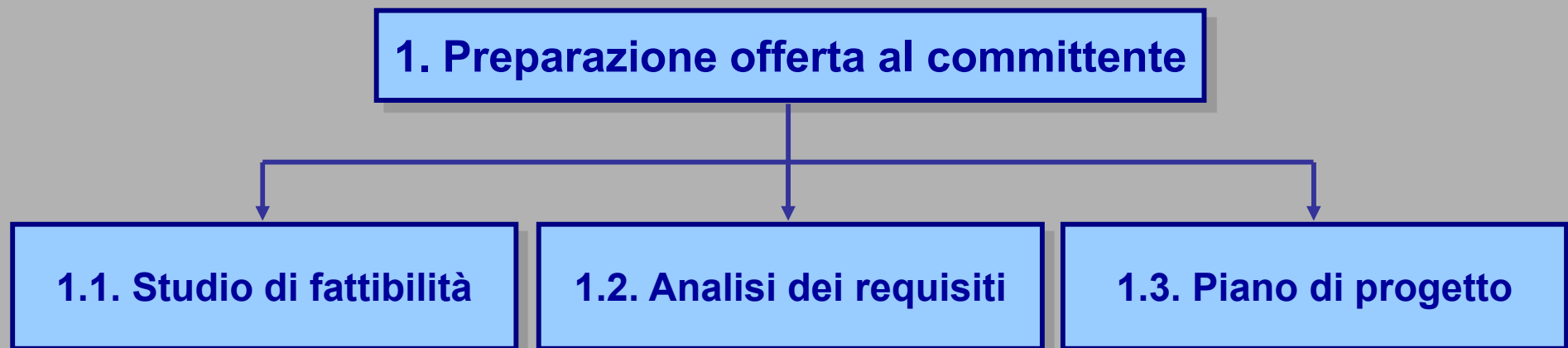
Ora produttiva vs ora di orologio

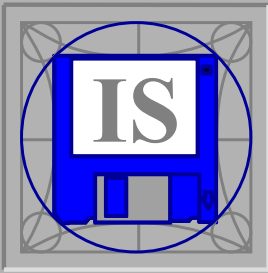
- ❑ La “taglia” di un progetto, per lunghezza temporale e costi economici, è determinata in sede di preventivo dal numero di **ore produttive** stimate necessarie per la realizzazione di quanto richiesto
- ❑ Lo svolgimento di ogni singola attività di progetto consuma
 - Ore di orologio, O^O , nella misura del tempo personale
 - Ore produttive, O^P , in funzione del tasso di raggiungimento degli obiettivi
- ❑ Il rapporto $R = \frac{O^O}{O^P}$ dice la pressione sulla persona ($R > 1$) o il margine utile del fornitore ($R < 1$)



Definizione delle attività

- ❑ La pianificazione deve scendere a un dettaglio idoneo a individuare attività brevi
 - Ciascuna assegnabile a un singolo incaricato
- ❑ Le attività hanno struttura gerarchica, ad albero
 - Ogni macro-attività genitore si compone di micro-attività figlie
 - Dislocate nel tempo in modo da soddisfare i loro vincoli di precedenza e non creare attese

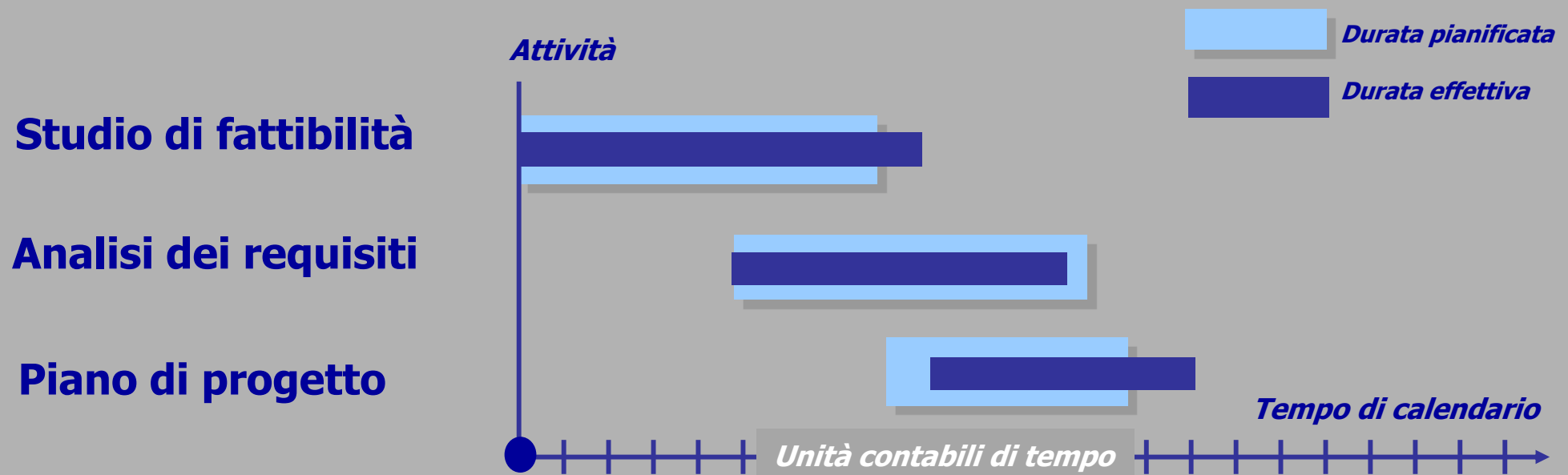


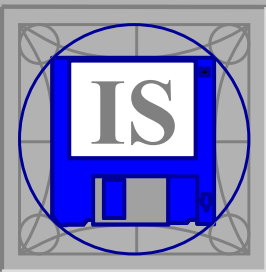


Diagrammi di Gantt

□ Dislocazione temporale delle attività

- Per rappresentarne la durata
- Per rappresentarne sequenzialità e parallelismo
- Per confrontare le stime con i progressi

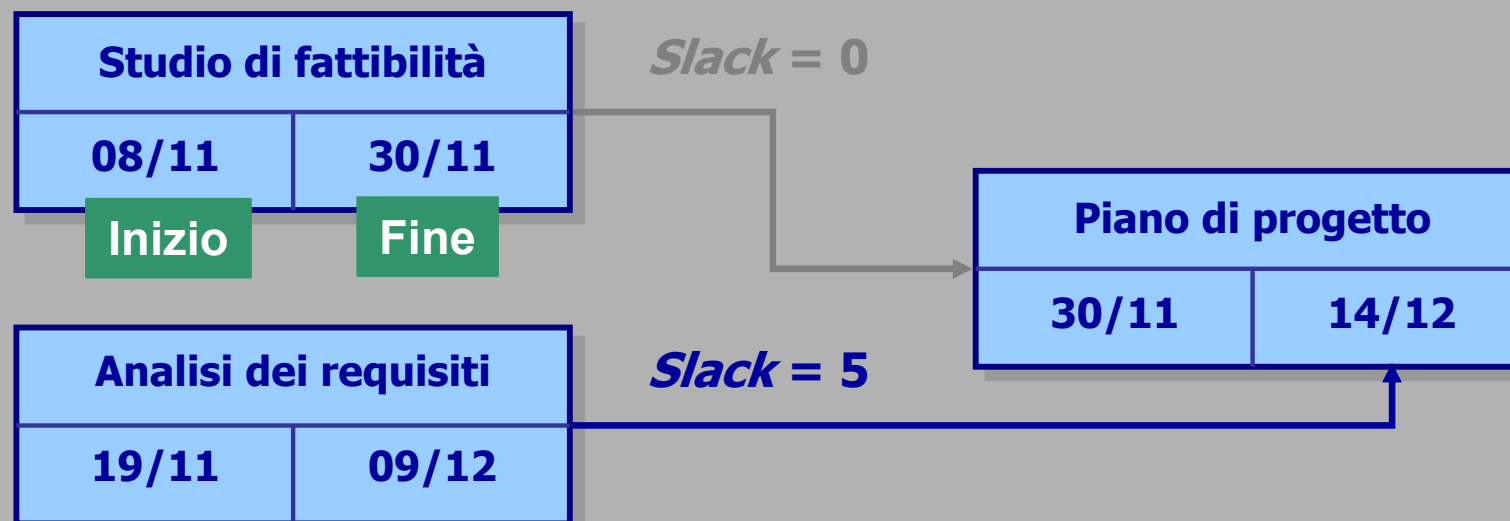


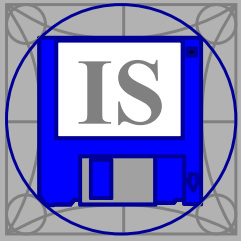


Diagrammi PERT – 1/2

□ Dipendenze temporali tra attività

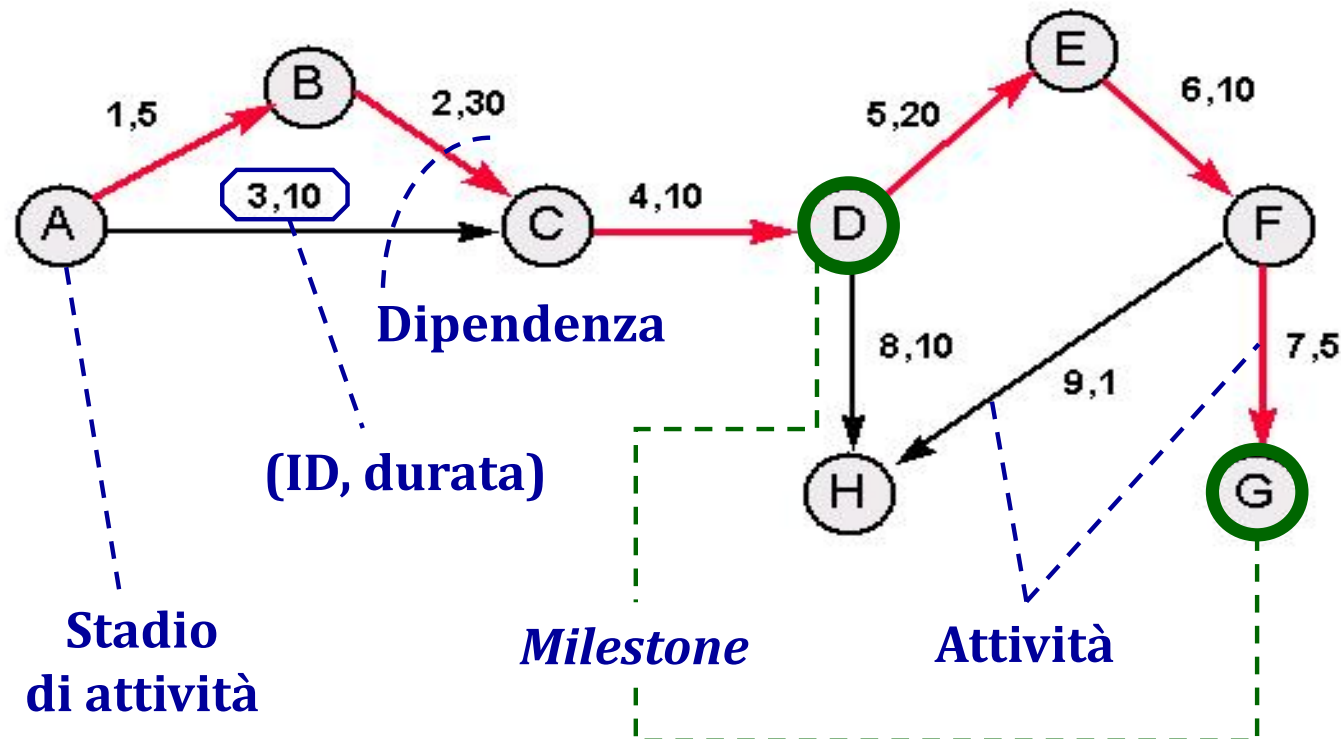
- Per ragionare all'indietro sulle scadenze di progetto, individuando il possibile margine temporale (*slack time*)
- Individuando i **cammini critici** → sequenze di attività ordinate, con esito importante, e dipendenze temporali tra loro molto strette

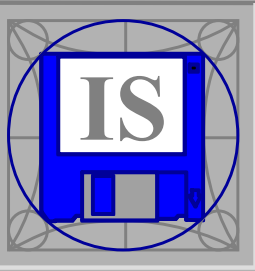




Diagrammi PERT – 2/2

Forma semplificata – in rosso il “cammino critico”

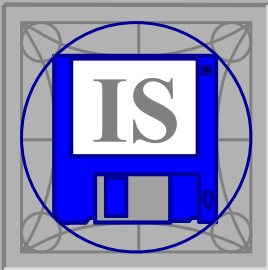




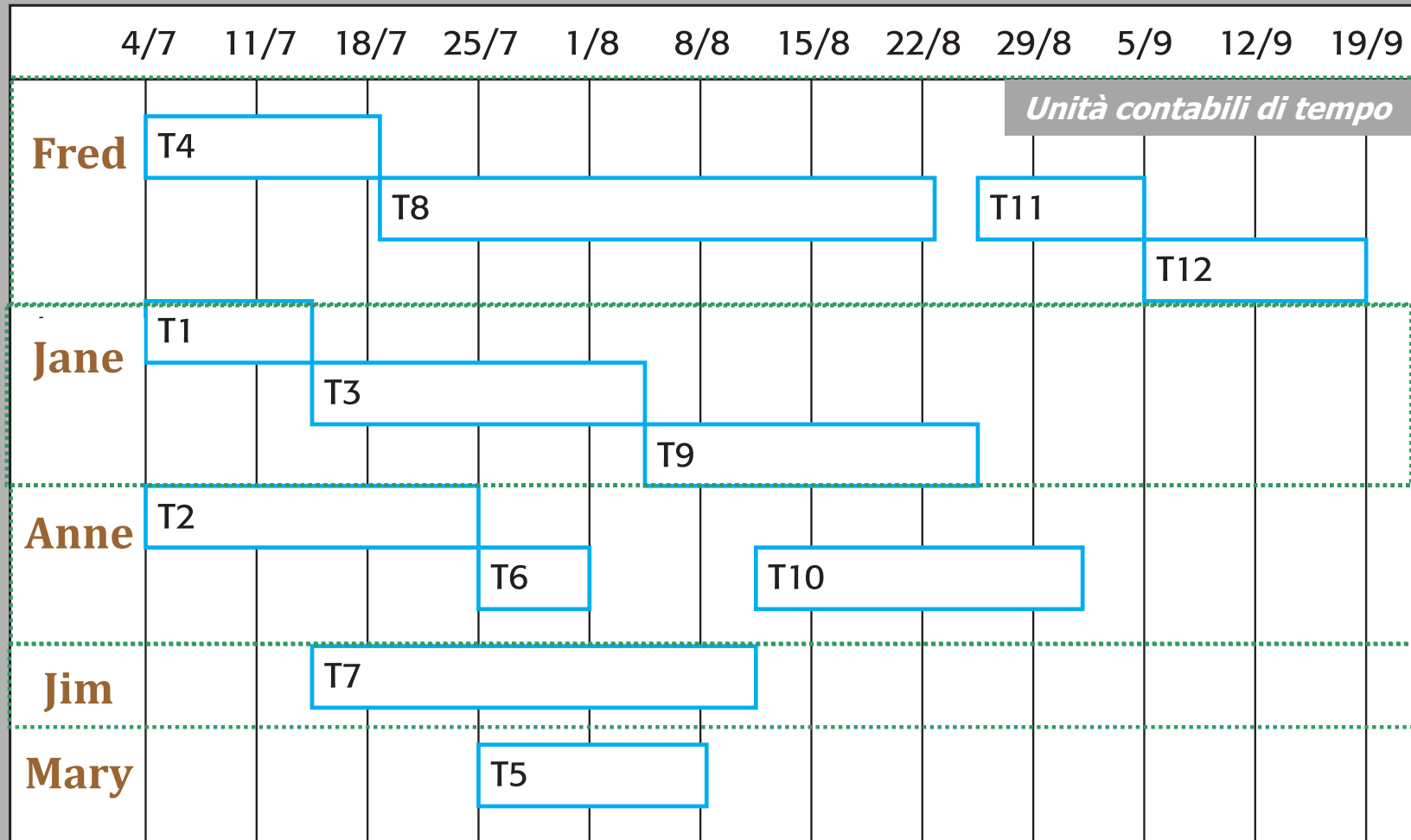
Allocazione delle risorse – 1/2

- ❑ **Assegnare attività a ruoli e ruoli a persone**
- ❑ **Difficoltà**
 - Non sottostimare
 - Non sovrastimare
- ❑ **Molte risorse sono impegnate su più progetti**
 - Aziendalmente, per non incorrere in sotto-utilizzo
 - Per voi, perché avete molti altri obblighi oltre a IS
- ❑ **Gestire più “cammini critici” su più progetti**

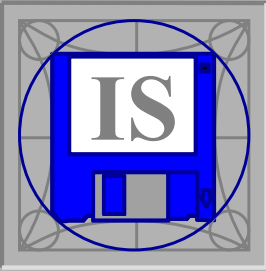




Allocazione delle risorse – 2/2



Tratto da: Ian Sommerville, *Software Engineering*, 8th ed.



Stima dei costi di progetto

❑ Come pianificare?

- Con strumenti che permettano di organizzare le attività
- Con strumenti che permettano di evidenziare le criticità
- Con strumenti che permettano di studiare scenari

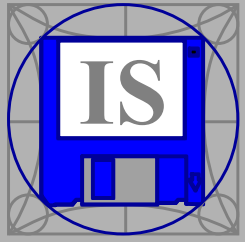
❑ Come definire durata e costo delle attività?

- Prima calcolando il **tempo/persona** stimato necessario
- Poi rapportandolo al tempo di calendario

❑ Come stimarlo?

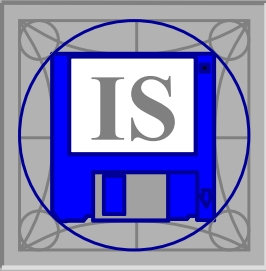
- Esperienza, analogia, competizione, algoritmo predittivo, raffinamenti

❑ Grana grossa sull'insieme, grana fine entro periodi brevi



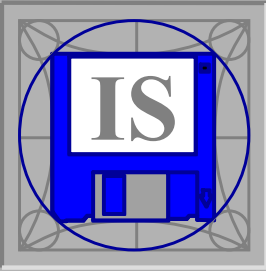
Fattori di influenza sulle stime

- ☐ Dimensione del progetto
- ☐ Esperienza del dominio
- ☐ Familiarità con le tecnologie
- ☐ Produttività dell'ambiente di lavoro
- ☐ Qualità attesa



Fonti di rischio

- ❑ Tecnologie di lavoro e di produzione SW
- ❑ Rapporti interpersonali
- ❑ Organizzazione del lavoro
- ❑ Requisiti e rapporti con gli *stakeholder*
- ❑ Tempi e costi



Gestione dei rischi – 1/2

❑ Identificazione

- Nel progetto, nel prodotto, nel mercato

❑ Analisi

- Probabilità di occorrenza, conseguenze possibili

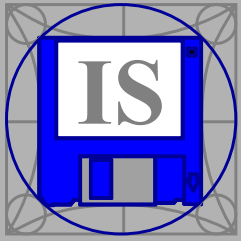
❑ Pianificazione

- Come evitare i rischi o mitigarne gli effetti

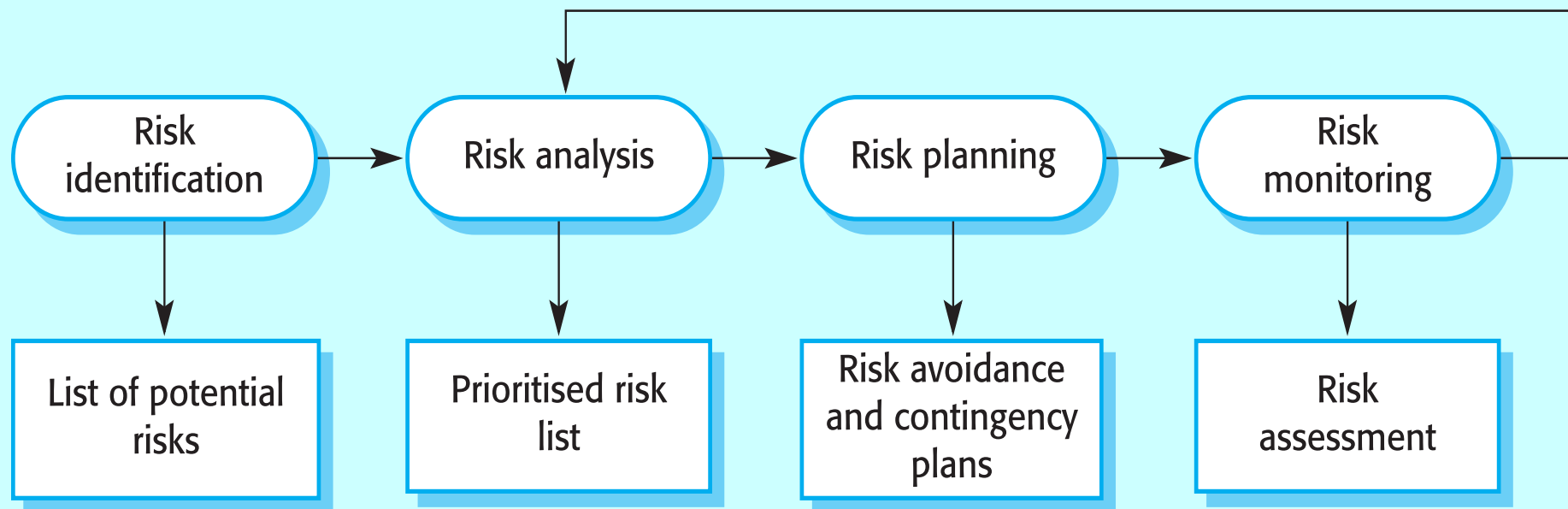
❑ Controllo

- Attenzione continua tramite rilevazione di indicatori
- Attuazione delle procedure di mitigazione
- Raffinamento delle strategie

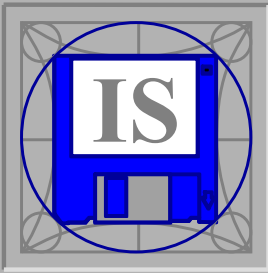




Gestione dei rischi – 2/2



Tratto da: Ian Sommerville, *Software Engineering*, 8th ed.



Secondo Standish Group nel 1994

❑ Progetti di successo (dati USA)

- In tempo, senza costi aggiuntivi, prodotto soddisfacente
- 16.2%

❑ Progetti a rischio

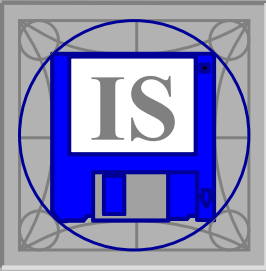
- Fuori tempo, o con costi aggiuntivi, o con prodotto difettoso
- 52.7%

❑ Fallimenti

- Progetti cancellati prima della fine
- 31.1%

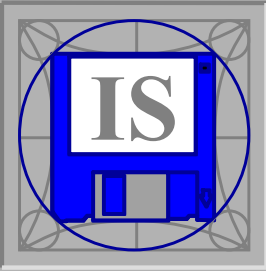
ATTENZIONE:

Vi è *bias* nei dati assoluti,
ma alla base vi sono forti
elementi di realtà



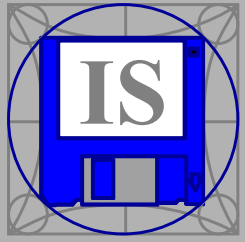
Fattori di successo

- | | |
|---|-------|
| <input type="checkbox"/> Coinvolgimento del cliente | 15.9% |
| <input type="checkbox"/> Supporto della direzione esecutiva | 13.9% |
| <input type="checkbox"/> Definizione chiara dei requisiti | 13.0% |
| <input type="checkbox"/> Pianificazione corretta | 9.6% |
| <input type="checkbox"/> Aspettative realistiche | 8.2% |
| <input type="checkbox"/> Personale competente | 7.2% |



Fattori di fallimento

- | | |
|---|-------|
| <input type="checkbox"/> Requisiti incompleti | 13.1% |
| <input type="checkbox"/> Mancato coinvolgimento del cliente | 12.4% |
| <input type="checkbox"/> Mancanza di risorse | 10.6% |
| <input type="checkbox"/> Aspettative non realistiche | 9.9% |
| <input type="checkbox"/> Mancanza di supporto esecutivo | 9.3% |
| <input type="checkbox"/> Fluttuazione dei requisiti | 8.7% |



Secondo Standish Group nel 2004

☐ Dieci anni dopo

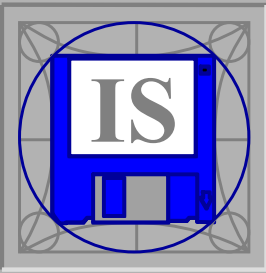
- Oltre 40.000 progetti USA studiati nel decennio
- Valore complessivo : 255 miliardi \$ (erano 250 nel 1994)

☐ Progetti finiti con successo : 34% (era 16,2%)

- Importante miglioramento nelle tecniche di gestione

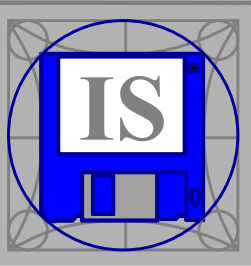
☐ Progetti falliti : 15% (era 31,1%)

- Danno economico : 55 miliardi \$ (140 nel 1994)
- Peggior eccesso di costo : 43% (189% nel 1994)



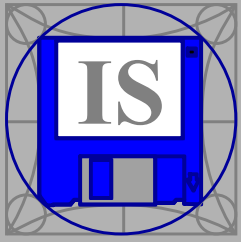
Consuntivo di periodo e retrospettiva

- ❑ Le buone prassi suggeriscono che lo sviluppo sia una successione di periodi brevi
- ❑ Ogni singolo periodo ha una struttura circolare, cioè finisce con una riflessione che guarda dietro, al rapporto tra attese e risultati
- ❑ Questa riflessione usa due strumenti
- ❑ Il “**consuntivo di periodo**”, che è quantitativo e serve a migliorare la pianificazione futura
- ❑ La “**retrospettiva**”, che è qualitativa e serve a riconoscere ciò che ha funzionato e ciò che no – e perché
- ❑ Quando la gestione di progetto segue prassi allo stato dell’arte, quei due strumenti si integrano perfettamente



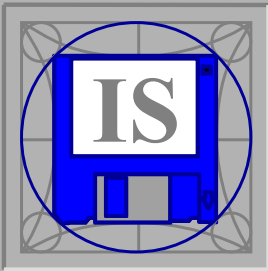
Cicli di avanzamento





Potere della retrospettiva





- ❑ *Software Project Managenment Technology Report*,
www.slideshare.net/Samuel90/project-management-technology-report
- ❑ La stima dei costi dei sistemi informativi automatizzati,
www.researchgate.net/publication/265986910 **LA STIMA DEI COSTI DEI SISTEMI INFORMATIVI AUTOMATIZZATI**
- ❑ B. Boehm et al., “Cost Models for Future Software Life Cycle Processes: CoCoMo II”, USC CSSE,
sunset.usc.edu/csse/research/COCOMOII/cocomo_main.html
- ❑ Standish Group, “The CHAOS Report”
[vedi calendario del corso]