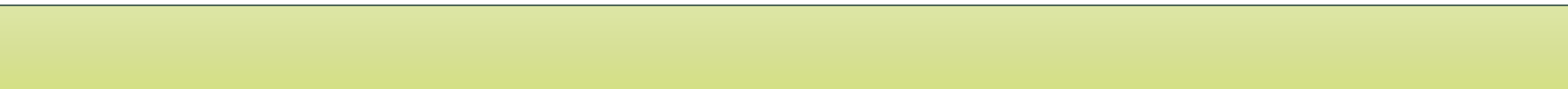




Basi Di Dati e di conoscenza

Progettazione di una base di dati



Progettazione di una base di dati

Progettare una base di dati significa definirne struttura, caratteristiche e contenuto. E quindi prevede l'uso di opportune metodologie. In base al grado di astrazione, la progettazione prevede:

- **Modello concettuale**: rappresenta la realtà dei dati e le relazioni tra essi attraverso uno schema
- **Modello logico**: descrive il modo attraverso il quale i dati sono organizzati negli archivi del calcolatore
- **Modello fisico**: descrive come i dati sono registrati nelle memorie di massa

Modello logico

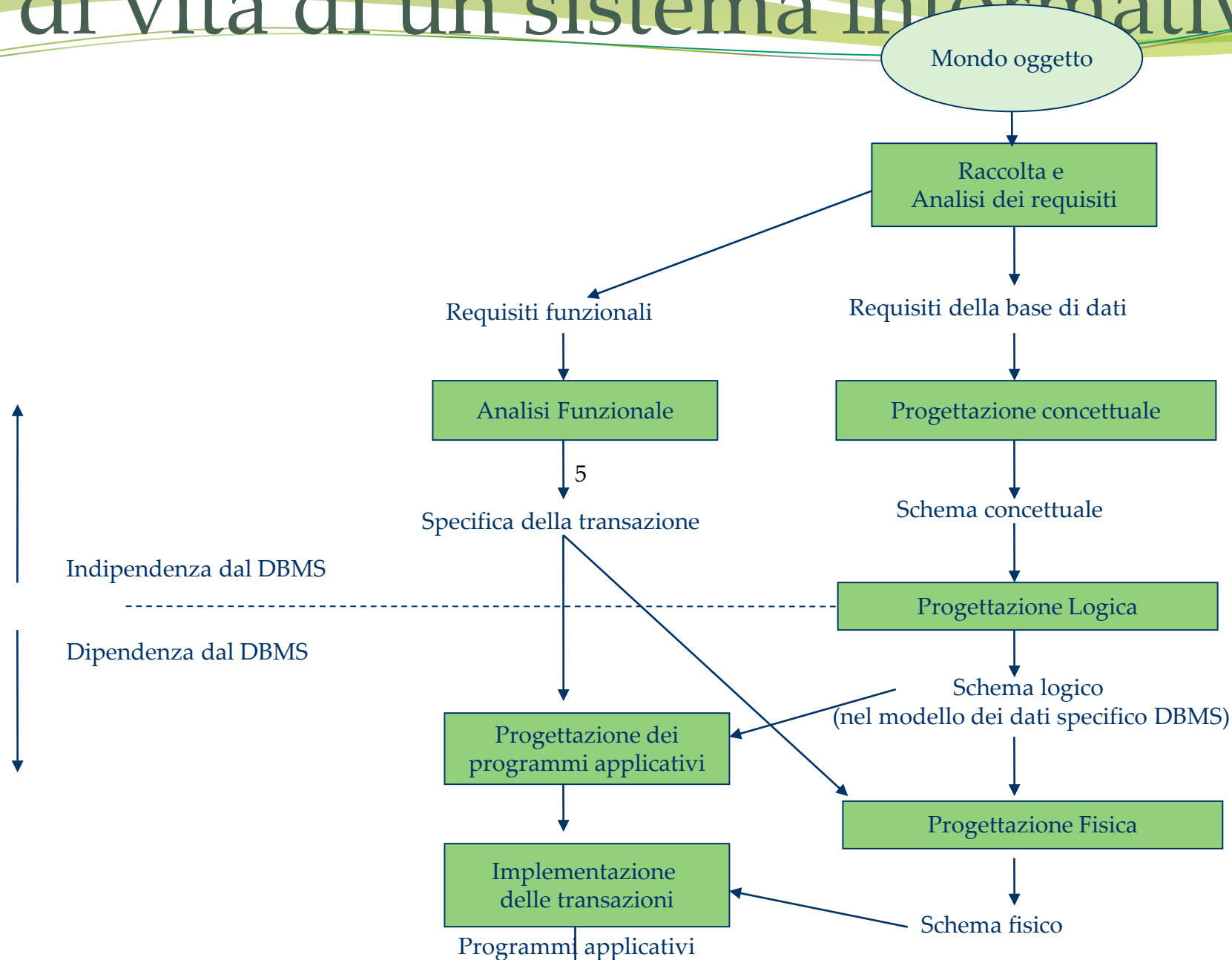
Obiettivo della fase di progettazione logica è pervenire, **a partire dallo schema concettuale**, a uno **schema logico** che lo rappresenti in modo fedele, “efficiente” e **indipendente** dal particolare **DBMS adottato**. A tal fine questa fase di progettazione può essere suddivisa in 2 passi:

1. **Ristrutturazione dello schema Entità-Relazione:** è una fase indipendente dal modello logico e si basa su criteri di ottimizzazione dello schema;
2. **Traduzione verso il modello logico:** fa riferimento ad uno specifico modello logico (**modello relazionale**).

Ciclo di vita di un sistema informativo

1. **Studio di fattibilità** definisce le varie alternative possibili, i relativi costi e le priorità di realizzazione.
2. **Raccolta e analisi dei requisiti** individua proprietà e funzionalità del sistema tramite interazione con gli utenti e definizione informale dei dati e delle operazioni.
3. **Progettazione** divisa in **progettazione dei dati** e **progettazione delle applicazioni**. Individua struttura e organizzazione dei dati e caratteristiche degli applicativi che vi dovranno accedere.
4. **Implementazione** realizza la base di dati e il codice dei programmi conformemente alle specifiche
5. **Validazione e collaudo** verifica il corretto funzionamento del sistema informativo
6. **Funzionamento** il sistema informativo diviene operativo

Ciclo di vita di un sistema informativo



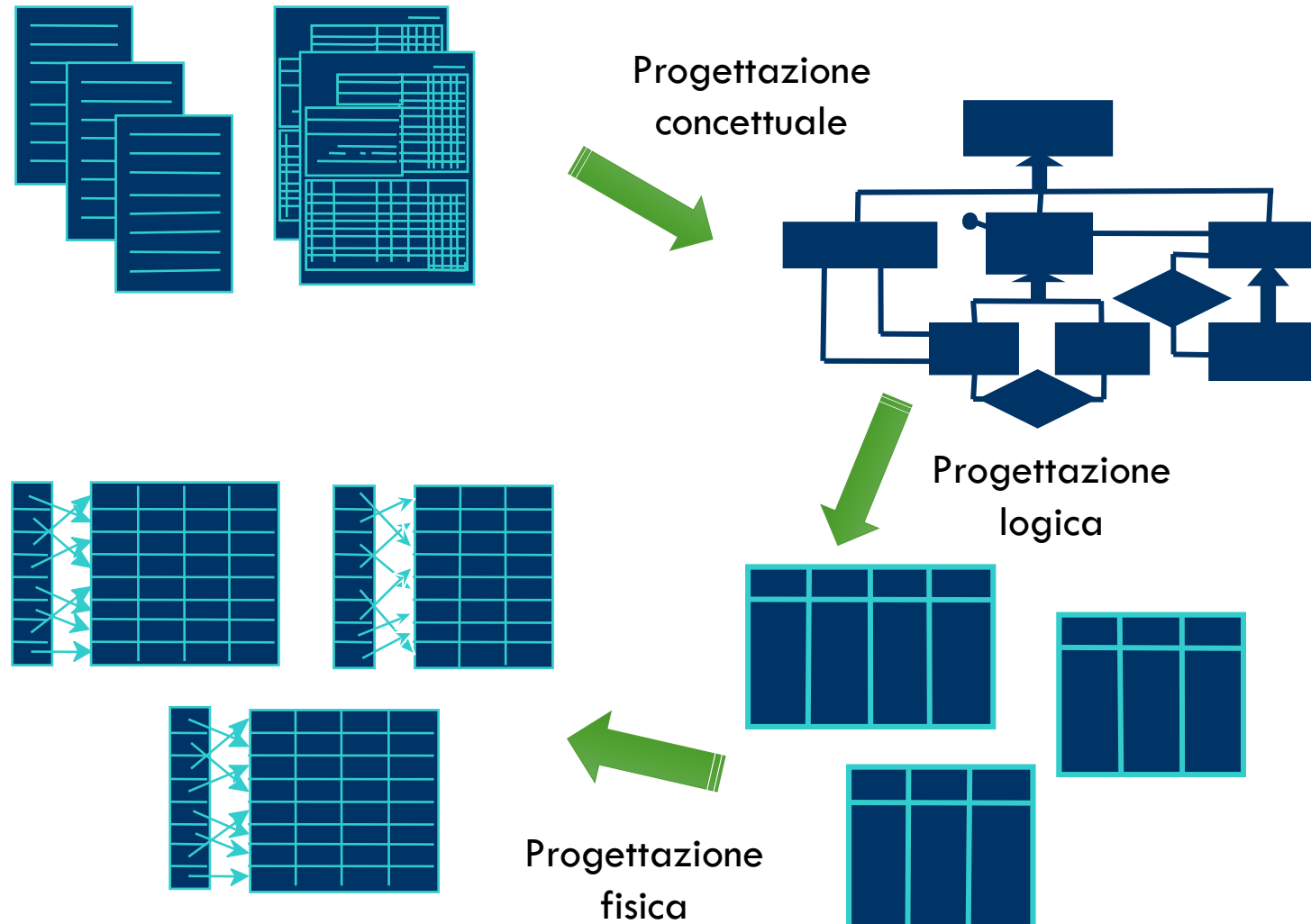
Fasi della progettazione

Progettazione concettuale: rappresenta le specifiche informali in modo formale e completo, ma indipendente dalla rappresentazione usata nei DBMS. Produce lo schema concettuale e fa riferimento ad un modello concettuale dei dati. Rappresenta il contenuto informativo e non la codifica.

Progettazione logica: traduce lo schema concettuale nello schema logico basato su un modello logico (es. modello relazionale) ancora indipendente dalla realizzazione fisica della base di dati.

Progettazione fisica: completa lo schema logico con la specifica dei parametri fisici di memorizzazione dei dati. Produce uno schema fisico e fa riferimento ad un modello fisico dei dati, dipendente dal DBMS.

Fasi della progettazione



Specifiche

Si distinguono le **specifiche sui dati** e le **specifiche sulle operazioni**

Progettazione concettuale: si usano le specifiche sui dati; quelle sulle operazioni servono a verificare che lo schema concettuale contenga tutte le informazioni necessarie alla loro esecuzione

Progettazione logica: lo schema concettuale riassume le specifiche sui dati, le specifiche sulle operazioni si usano per ottenere uno schema logico che permetta di eseguirle in modo efficiente. Bisogna conoscere il modello logico ma non il DBMS adottato.

Progettazione fisica: si usano lo schema logico e le specifiche sulle operazioni per implementare il sistema in modo efficiente

Progettazione concettuale

Il prodotto è uno schema E-R che descrive le specifiche sui dati relative ad una applicazione.

Si può suddividere in due fasi:

- **Raccolta e analisi dei requisiti**
 - Individuazione dei problemi da risolvere
 - Chiarimento ed organizzazione delle specifiche
- **Definizione dello schema E-R**

Progettazione concettuale

Il prodotto è uno schema E-R che descrive le specifiche sui dati relative ad una applicazione.

Si può suddividere in due fasi:

- **Raccolta e analisi dei requisiti**
 - Individuazione dei problemi da risolvere
 - Chiarimento ed organizzazione delle specifiche
- **Definizione dello schema E-R**

Raccolta e analisi dei requisiti

- I **requisiti** definiscono le caratteristiche dell'applicazione da realizzare (dati, operazioni).
- I requisiti spesso sono espressi con frasi in linguaggio naturale spesso ambigue e disorganizzate.
- L'analisi consiste nel chiarimento e nell'organizzazione delle specifiche
 - Il reperimento dei requisiti è un'attività difficile e non standardizzabile;
 - l'attività di analisi inizia con i primi requisiti raccolti e spesso indirizza verso altre acquisizioni.

Raccolta e analisi dei requisiti

Realizzazione di una descrizione del problema in linguaggio naturale che rispetti criteri di completezza e non ambiguità:

- corretto livello di astrazione
- frasi standardizzate
- semplicità delle specifiche (evitare frasi contorte)
- eliminazione di sinonimi o omonimi
- esplicitazione dei riferimenti fra termini
- glossario dei termini

Per i **dati**, specificare il numero delle **occorrenze** previste.

Per le **operazioni**, specificare il **numero di volte** che si prevede che debbano essere eseguite in un certo arco di tempo.

Fonti dei requisiti

- **Utenti** (interviste, documentazione scritta)
- **Documentazione esistente** (normative, leggi e regolamenti del settore, regolamenti interni, procedure aziendali, moduli)
- **Realizzazioni preesistenti** (applicativi da rimpiazzare, applicazioni che dovranno interagire col sistema da realizzare)

Interazione con gli utenti

- utenti diversi possono fornire informazioni diverse
- utenti a livello più alto hanno spesso una visione più ampia ma meno dettagliata
- le interviste portano spesso ad una acquisizione dei requisiti «per raffinamenti successivi»
 - effettuare spesso verifiche di comprensione e coerenza
 - verificare anche per mezzo di esempi (generalisti e relativi a casi limite)
 - richiedere definizioni e classificazioni
 - far evidenziare gli aspetti essenziali rispetto a quelli marginali

Raccolta e analisi dei requisiti

Esempio 1

Base di dati bibliografica

Si vogliono organizzare i dati di interesse per automatizzare la gestione dei riferimenti bibliografici

Raccolta e analisi dei requisiti

Esempio 1

Base di dati bibliografica

Si vogliono organizzare i dati di interesse per automatizzare la gestione dei riferimenti bibliografici, con tutte le informazioni da riportarsi in una bibliografia.

Per ogni pubblicazione deve esistere un codice identificante costituito da sette caratteri, indicanti le iniziali degli autori, l'anno di pubblicazione e un carattere aggiuntivo per la discriminazione delle collisioni.

Raccolta e analisi dei requisiti

Esempio 1

Base di dati bibliografica

Si vogliono organizzare i dati di interesse per automatizzare la gestione dei riferimenti bibliografici, con tutte le informazioni da riportarsi in una bibliografia. Le pubblicazioni sono di due tipi, monografie (per le quali interessano editore, data e luogo di pubblicazione) e articoli su rivista (con nome della rivista, volume, numero, pagine e anno di pubblicazione); per entrambi i tipi si debbono ovviamente riportare i nomi degli autori. Per ogni pubblicazione deve esistere un codice identificante costituito da sette caratteri, indicanti le iniziali degli autori, l'anno di pubblicazione e un carattere aggiuntivo per la discriminazione delle collisioni.

Raccolta e analisi dei requisiti

Esempio 2- Più articolato

Società di formazione (continua)

Si vuole realizzare una base di dati per una società che eroga corsi, di cui vogliamo rappresentare i dati dei partecipanti ai corsi e dei docenti. Per gli studenti (circa 5000), identificati da un codice, si vuole memorizzare il codice fiscale, il cognome, l'età, il sesso, il luogo di nascita, il nome dei loro attuali datori di lavoro, i posti dove hanno lavorato in precedenza insieme al periodo, l'indirizzo e il numero di telefono, i corsi che hanno frequentato (i corsi sono in tutto circa 200) e il giudizio finale.

Raccolta e analisi dei requisiti

Esempio 2- Più articolato

Società di formazione (continua)

Rappresentiamo anche i seminari che stanno attualmente frequentando e, per ogni giorno, i luoghi e le ore dove sono tenute le lezioni. I corsi hanno un codice, un titolo e possono avere varie edizioni con date di inizio e fine e numero di partecipanti. Se gli studenti sono liberi professionisti, vogliamo conoscere l'area di interesse e, se lo possiedono, il titolo. Per quelli che lavorano alle dipendenze di altri, vogliamo conoscere invece il loro livello e la posizione ricoperta.

Raccolta e analisi dei requisiti

Esempio 2- Più articolato

Società di formazione

Per gli insegnanti (circa 300), rappresentiamo il cognome, l'età, il posto dove sono nati, il nome del corso che insegnano, quelli che hanno insegnato nel passato e quelli che possono insegnare. Rappresentiamo anche tutti i loro recapiti telefonici. I docenti possono essere dipendenti interni della società o collaboratori esterni.

Raccolta e analisi dei requisiti

Esempio 2- Più articolato

Glossario dei termini

Termine	Descrizione	Sinonimi	Collegamenti
Partecipante	Persona che partecipa ai corsi	Studente	Corso, Società
Docente	Docente dei corsi. Può essere esterno	Insegnante	Corso
Corso	Corso organizzato dalla società. Può avere più edizioni.	Seminario	Docente, Partecipante
Società	Ente presso cui i partecipanti lavorano o hanno lavorato	Posto	Partecipante

Raccolta e analisi dei requisiti

Esempio 2- Più articolato

Strutturazione requisiti in gruppi di frasi omogenee

Frase di carattere generale

Si vuole realizzare una base di dati per una società che eroga corsi, di cui vogliamo rappresentare i dati dei partecipanti ai corsi e dei docenti.

Raccolta e analisi dei requisiti

Esempio 2- Più articolato

Strutturazione requisiti in gruppi di frasi omogenee

Frase relative ai partecipanti

Per i partecipanti (circa 5000), identificati da un codice, rappresentiamo il codice fiscale, il cognome, l'età, il sesso, la città di nascita, i nomi dei loro attuali datori di lavoro e di quelli precedenti (insieme alle date di inizio e fine rapporto), le edizioni dei corsi che stanno attualmente frequentando e quelli che hanno frequentato nel passato, con la relativa votazione finale in decimi.

Frase relative a tipi specifici di partecipanti

Per i partecipanti che sono liberi professionisti, rappresentiamo l'area di interesse e, se lo possiedono, il titolo professionale. Per i partecipanti che sono dipendenti, rappresentiamo invece il loro livello e la posizione ricoperta

Raccolta e analisi dei requisiti

Esempio 2- Più articolato

Strutturazione requisiti in gruppi di frasi omogenee

Frase relative ai datori di lavoro

Relativamente ai datori di lavoro presenti e passati dei partecipanti, rappresentiamo il nome, l'indirizzo e il numero di telefono.

Frase relative ai corsi

Per i corsi (circa 200), rappresentiamo il titolo e il codice, le varie edizioni con date di inizio e fine e, per ogni edizione, rappresentiamo il numero di partecipanti e il giorno della settimana, le aule e le ore dove sono tenute le lezioni.

Raccolta e analisi dei requisiti

Esempio 2- Più articolato

Strutturazione requisiti in gruppi di frasi omogenee

Frase relative ai docenti

Per i docenti (circa 300), rappresentiamo il cognome, l'età, la città di nascita, tutti i numeri di telefono, il titolo del corso che insegnano, di quelli che hanno insegnato nel passato e di quelli che possono insegnare. I docenti possono essere dipendenti interni della società di formazione o collaboratori esterni.

Progettazione concettuale

Criteri di rappresentazione

- rappresentare mediante **entità** classi di oggetti con esistenza autonoma e caratterizzati da proprietà significative
- rappresentare mediante **attributi** gli oggetti con struttura semplice e che non presentino proprietà rilevanti
- rappresentare mediante **relazioni** concetti che associano più entità precedentemente identificate
- rappresentare mediante **generalizzazioni** concetti che risultino essere casi particolari di altri.

Strategie di progetto

Si possono utilizzare le strategie tipiche dello sviluppo di un processo di ingegnerizzazione (es. ingegneria del software).

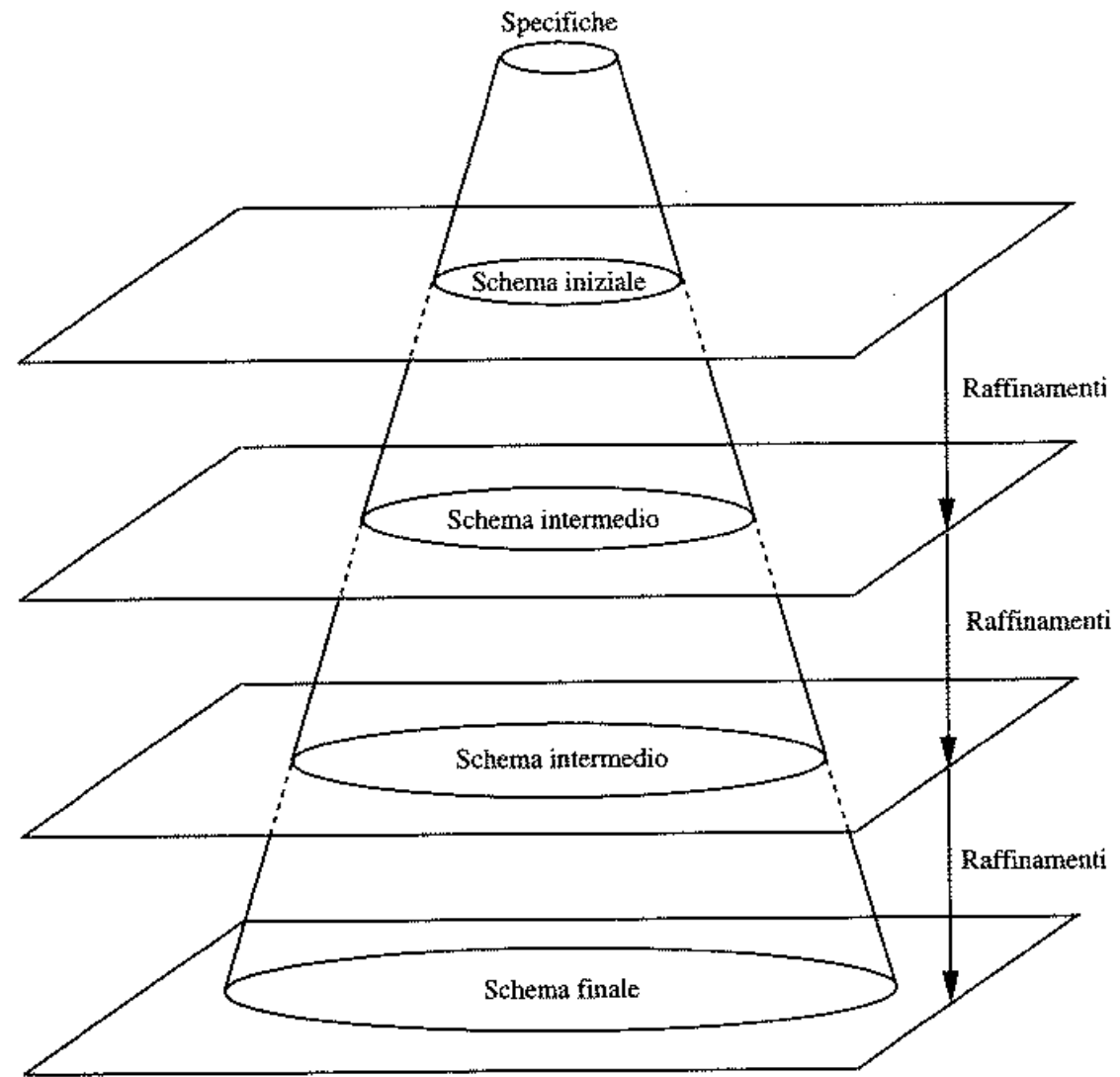
- Strategie top-down
- Strategie bottom-up
- Strategie inside-out
- Approcci misti

Progettazione concettuale:

Strategia top-down

A partire da uno schema che descrive le specifiche mediante pochi concetti molto astratti, si produce uno schema concettuale mediante raffinamenti successivi che aggiungono via via più dettagli.

I raffinamenti successivi, come accade anche per altre strategie, vengono realizzati mediante trasformazioni elementari, denominate *primitive di trasformazione top-down* che trasformano un concetto di uno schema in una struttura più complessa che descrive la realtà in maggiore dettaglio.



Progettazione concettuale:

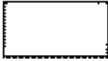
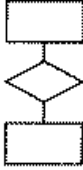
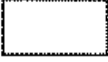
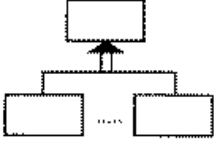

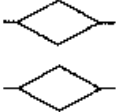
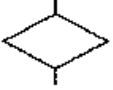
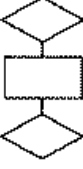



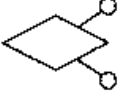
Primitive di trasformazione top-down

- **T1, da entità a relazione fra entità:** si applica quando si verifica che una entità descrive due concetti diversi legati logicamente fra loro
- **T2, da entità a generalizzazione:** si applica quando una entità è composta da sotto-entità distinte o, più in generale, che uno stesso concetto può in realtà comprendere più concetti.
- **T3, da relazione a insieme di relazioni:** si applica quando una relazione descrive in realtà due o più relazioni fra le medesime entità; tipicamente le due relazioni sono due specializzazioni di una relazione più generale.

Progettazione concettuale:

Primitive di trasformazione top-down

- **T4, da relazione ad entità con relazioni:** si applica quando una relazione descrive un concetto con esistenza autonoma ai fini dell'applicazione o concetti di cui si possono avere più occorrenze.
- **T5, introduzione di attributi in una entità:** si applica per introdurre nuovi attributi ad una entità, che aiutino a descriverla in modo più completo.
- **T6, introduzione di attributi su relazioni:** si applica per aggiungere proprietà a relazioni.

Primitiva di trasformazione	Concetto iniziale	Risultato
T ₁ : Da entità a relazione tra entità		
T ₂ : Da entità a generalizzazione		
T ₃ : Da relazione a insieme di relazioni		
T ₄ : Da relazione a entità con relazioni		
T ₅ : Introduzione di attributi su entità		
T ₆ : Introduzione di attributi su relazioni		

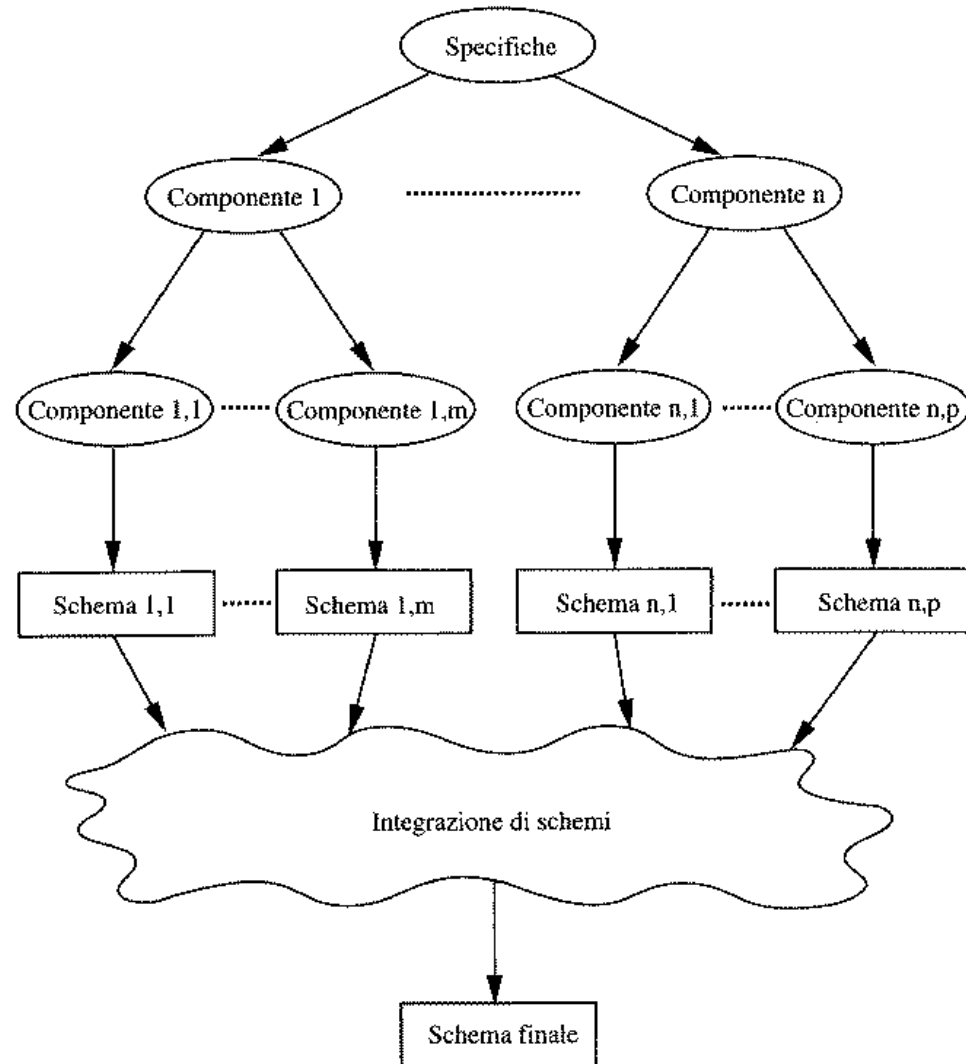
Progettazione concettuale: Strategia Bottom-up

Nella strategia bottom-up le specifiche iniziali sono suddivise in componenti via via sempre più piccole, fino a descrivere frammenti elementari della realtà.

Le componenti vengono poi fuse con trasformazioni successive (primitive di trasformazione bottom-up) per giungere allo schema concettuale finale.

Ogni trasformazione introduce nuovi concetti non presenti al livello precedente.

Progettazione concettuale: Strategia Bottom-up





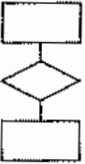

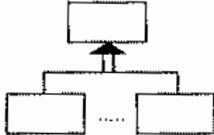



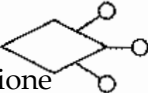
Progettazione concettuale:

Primitive di trasformazione bottom-up

- **T1, generazione di entità:** si applica quando si individua nelle specifiche una classe di oggetti caratterizzata da proprietà comuni.
- **T2, generazione di relazione:** si applica quando nelle specifiche si individua un legame logico fra entità.
- **T3, generazione di generalizzazione:** si applica quando si individua un legame fra diverse entità riconducibile ad una generalizzazione, quando cioè le diverse entità possono essere istanze di una stessa classe.
- **T4, aggregazione di attributi su entità:** si applica quando si individua una entità che può essere rappresentata come aggregazione di attributi presenti nelle specifiche.
- **T5, aggregazione di attributi su relazione:** analoga a T4, ma relativa ad una relazione.

Progettazione concettuale:

Primitive di trasformazione bottom-up

Primitiva di trasformazione	Concetto iniziale	Risultato
T ₁ : Generazione di entità		
T ₂ : Generazione di relazione		
T ₃ : Generazione di generalizzazione		
T ₄ : Aggregazione di attributi su entità		
T ₅ : Aggregazione di attributi su relazioni		

Progettazione concettuale:

Strategia inside-out

Può essere vista come un caso particolare della strategia bottom-up: individua solo alcuni concetti importanti, per poi procedere *a macchia d'olio*.

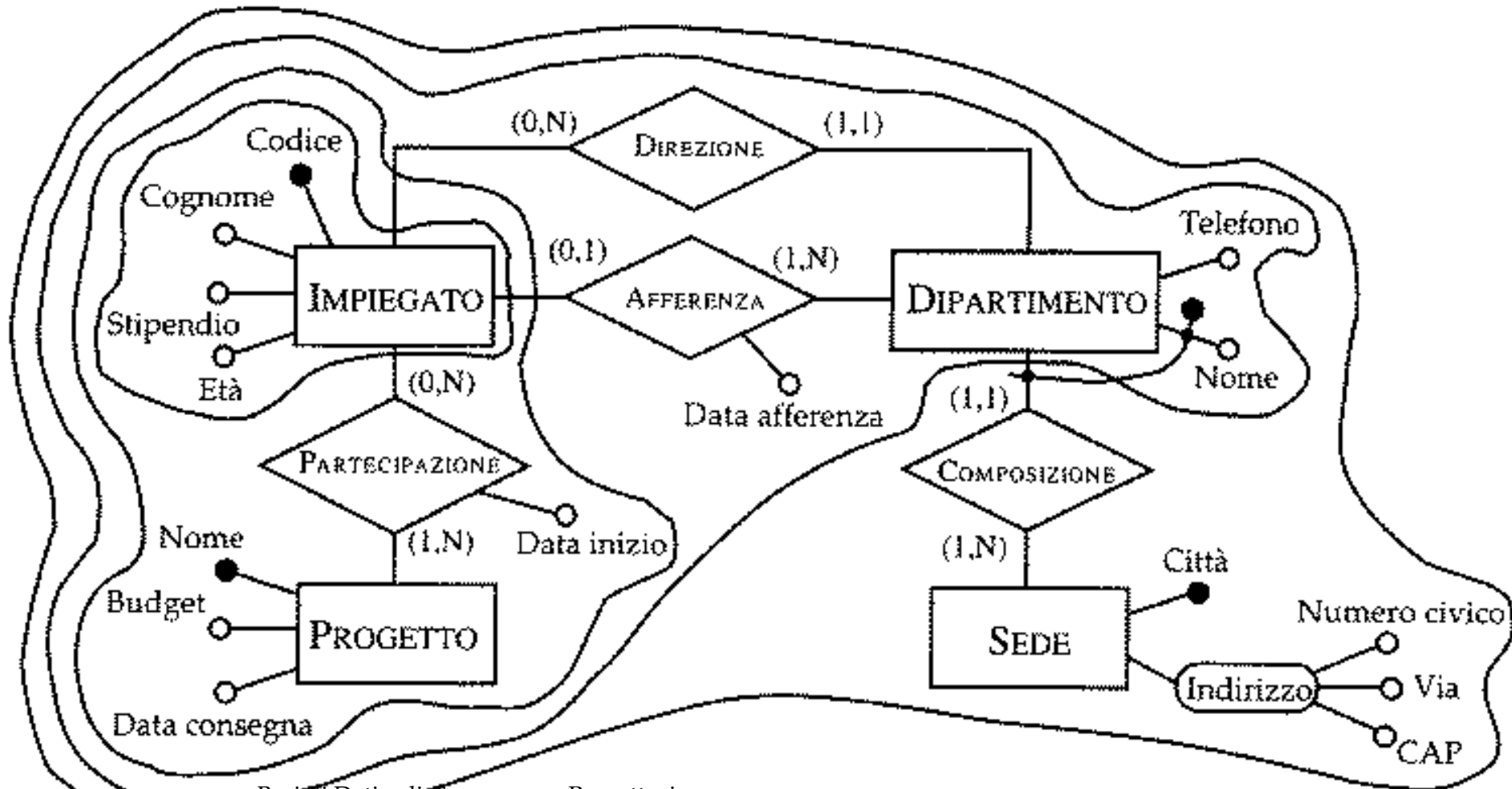
Si rappresentano prima i concetti più vicini a quelli di partenza per poi sviluppare quelli più lontani attraverso una *navigazione* nelle specifiche.

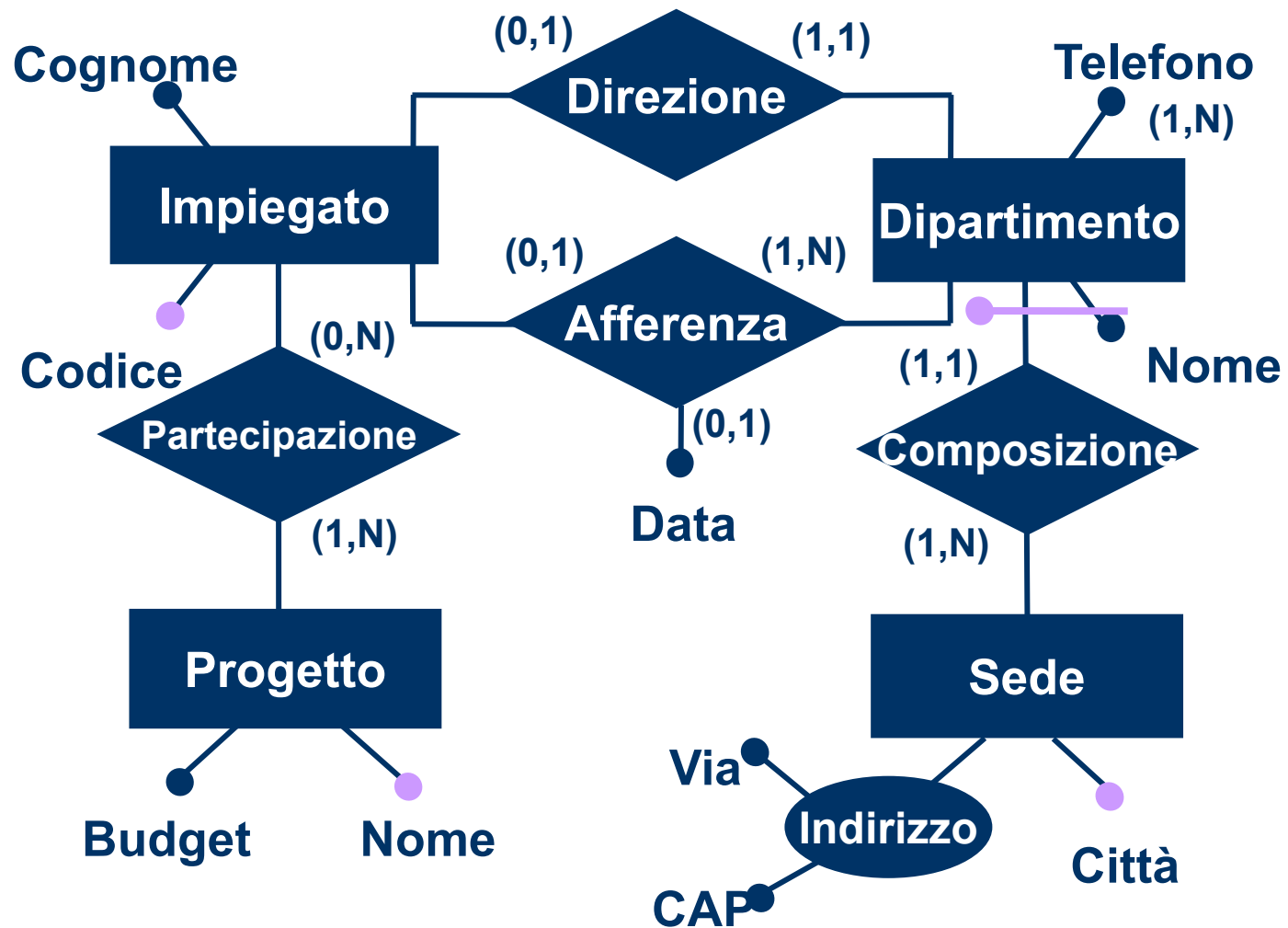
Vantaggi: non richiede passi di integrazione.

Svantaggi: è necessario di volta in volta esaminare tutte le specifiche e descrivere i nuovi concetti nel dettaglio.

Non è possibile procedere per livelli di astrazione.

Progettazione concettuale: Strategia inside-out





Progettazione concettuale:

Strategia mista

- Cerca di unire i vantaggi delle strategie top-down e bottom-up. A un estremo, si individuano **componenti elementari**, all'altro si crea uno **schema scheletro** contenente concetti di base da espandere, con raffinamenti successivi, in modo top-down.
- Contemporaneamente, dalle specifiche, si creano in modo bottom-up i concetti non presenti nello schema scheletro.
- Si adatta ad esigenze opposte di suddivisione di un problema complesso in sottoproblemi, e di procedura per raffinamenti successivi.
- Ingloba in pratica anche la strategia inside-out

Spesso è l'unica strategia utilizzabile

Progettazione concettuale:

Strategia mista- In pratica

- si procede di solito con una strategia ibrida (**mista**):
 - si individuano i concetti principali e si realizza uno **schema scheletro**
 - sulla base di questo si può decomporre
 - poi si **raffina**, si espande, si integra
- **Schema Scheletro**: Si individuano i concetti più importanti, ad esempio perché più citati o perché indicati esplicitamente come cruciali e li si organizza in un semplice schema concettuale

Qualità di uno schema concettuale

Nel definire uno schema concettuale ci sono alcune proprietà che bisogna cercare di garantire:

- **Correttezza:** lo schema utilizza propriamente i costrutti del modello di riferimento.
- **Completezza:** tutti i dati di interesse sono rappresentati e tutte le operazioni sono eseguibili a partire dai concetti descritti nello schema.
- **Leggibilità:** i requisiti sono rappresentati in modo naturale e comprensibile. Estetica dello schema.
- **Minimalità:** le specifiche sono rappresentate una sola volta. Non sempre, tuttavia, eventuali ridondanze sono indesiderate.

Progettazione concettuale:

Metodologia

- **Analisi dei requisiti**
 - Analizzare i requisiti ed eliminare le ambiguità
 - Costruire un glossario dei termini
 - Raggruppare i requisiti in insiemi omogenei
- **Passo base:** Definire uno schema scheletro con i concetti più rilevanti
- **Passo decomposizione:** Effettuare una decomposizione dei requisiti con riferimento ai concetti presenti nello schema scheletro

Progettazione concettuale:

Metodologia

- **Passo iterativo:** da ripetere per tutti i sottoschemi finchè ogni specifica è stata rappresentata:
 - Raffinare i concetti presenti in base alle specifiche
 - Aggiungere nuovi concetti
- **Passo di integrazione:** Integrare i vari sottoschemi utilizzando lo schema scheletro
- **Analisi di qualità:**
 - Verifica correttezza
 - Verifica completezza
 - Verifica leggibilità
 - Verifica la minimalità e documentare eventuali ridondanze volute