

Laboratorio di Basi di dati - Progettazione

Checklist per evitare errori nella relazione

Contrassegnare tutti i controlli effettuati.

1 Progettazione concettuale

○ Requisiti iniziali:

- ☐ I requisiti iniziali sono riportati integralmente, ovvero comprendono anche eventuali schermate o informazioni fornite in un formato non testuale.

○ Requisiti rivisti:

- ☐ I requisiti rivisti non comprendono sinonimi e omonimi.
- ☐ I requisiti rivisti comprendono anche le informazioni date dalle schermate che sono state fornite come parte dei requisiti iniziali.
- ☐ I requisiti rivisti sono formulati utilizzando uno stile sintattico semplice e uniforme per tutte le frasi (ad esempio: "per <dato> rappresentiamo <proprietà>").

○ Schema ER principale:

▪ **Entità:**

- ☐ Le entità hanno solo attributi nominati nei requisiti rivisti.
- ☐ Ogni entità ha un identificatore.
- ☐ Gli identificatori delle entità **non** sono ID o codici (con l'eccezione dei rari casi in cui ID e codici sono stati nominati nei requisiti).
- ☐ Gli identificatori delle entità hanno valori che presumibilmente **non** variano.
- ☐ Le entità (e le associazioni) **non** hanno attributi corrispondenti a identificatori di *altre* entità (in altre parole, a differenza del modello relazionale, i "collegamenti" tra entità sono rappresentati tramite associazioni e **non** attraverso attributi). N.B., gli identificatori esterni, non rientrando in questo caso, sono ammessi.

▪ **Associazioni:**

- ☐ Le associazioni **non** hanno identificatori.
- ☐ Le associazioni **non** rappresentano operazioni sul sistema ma legami tra i dati.
- ☐ Le associazioni ternarie indicano che potenzialmente è possibile avere ogni combinazione di occorrenze delle tre entità collegate. Se non è quello che si desidera, usare associazioni binarie.

▪ **Generalizzazioni:**

- ☐ Per ogni generalizzazione è stato indicato il tipo.
- ☐ Nelle generalizzazioni i figli sono, come i genitori, entità e **non** occorrenze di entità.

▪ **Ridondanze**

- ☐ Lo schema ER principale **contiene** ridondanze che saranno poi analizzate nella fase di progettazione logica.

○ Regole aziendali:

- ☐ Le regole aziendali definiscono solo regole non esprimibili con lo schema ER.
- ☐ Le regole aziendali sono effettivamente controllabili utilizzando i dati rappresentati nell'ER.
- ☐ Le regole aziendali sono sufficientemente precise da essere implementabili da un programmatore.

○ Schema ER principale+regole aziendali:

- ☐ Per ogni ridondanza esiste o una regola aziendale che indica come mantenere la coerenza o una regola aziendale di derivazione che indica come derivare la ridondanza.
- ☐ Lo schema ER principale + regole aziendali sono equivalenti ai requisiti rivisti.

2 Progettazione logica

○ Tavola dei volumi:

- ☐ La tavola dei volumi contiene tutte le entità e le associazioni presenti nello schema ER principale.
- ☐ La tavola dei volumi contiene valori coerenti con lo schema ER principale e con il funzionamento a regime del sistema.

○ Tavola delle operazioni

- ☐ La tavola delle operazioni contiene operazioni coerenti con i requisiti: sono state considerate sia le operazioni citate esplicitamente nei requisiti iniziali sia le schermate fornite.
- ☐ La tavola delle operazioni contiene sia operazioni che “leggono” i dati che operazioni che modificano e inseriscono dati.
- ☐ La tavola delle operazioni contiene numeri coerenti con il funzionamento a regime del sistema.
- ☐ Gli schemi delle operazioni e le tavole degli accessi **non** sono riportati qui, ma nell’analisi delle ridondanze perché sono specifici per la singola ridondanza.

○ Analisi delle ridondanze:

- ☐ Separatamente per ogni ridondanza analizzata:
 1. Sono state elencate **sia** le operazioni di **lettura** dei dati **che** quelle di **modifica/inserimento** più significative che modificano/utilizzano la ridondanza.
 2. *Per ogni relativa operazione* sono stati riportati gli schemi delle operazioni in presenza e assenza della ridondanza.
 3. *Per ogni relativa operazione* sono state riportate le tavole degli accessi in presenza e assenza della ridondanza.
 4. È stato riportato lo spazio occupato dalla ridondanza.
 5. È stato confrontato lo spazio e il numero di accessi in presenza di ridondanza con lo spazio e il numero di accessi in assenza di ridondanza e si è deciso se tenere la ridondanza o no.

○ Schema ER ristrutturato+regole aziendali:

- ☐ Lo schema ER ristrutturato + regole aziendali è *equivalente* allo schema ER principale + regole aziendali: nello schema ER ristrutturato non si possono introdurre nuovi attributi/entità/associazioni se non quelli che derivano dalla ristrutturazione. Se si scopre un errore o una lacuna, occorre correggere prima l’ER principale.
- ☐ Gli identificatori delle entità **sono** stati sostituiti con ID o codici nei casi in cui gli identificatori dell’ER principale sarebbero risultati inefficienti.
- ☐ Le entità **non** hanno attributi che corrispondono a “chiavi esterne” e che possono essere rappresentati tramite associazioni.
- ☐ Le associazioni **non** hanno identificatori.
- ☐ Nell’ER ristrutturato, a differenza dell’ER principale, è **possibile** usare come identificatori delle entità ID o codici per motivi di efficienza.
- ☐ Ogni generalizzazione è stata eliminata motivando la scelta e rispettandone la semantica (parziale/totale e sovrapposta/esclusiva).
- ☐ Le associazioni/attributi sui figli/genitori eliminati in una generalizzazione sono state sostituite rispettando la semantica della generalizzazione (eventualmente introducendo nuove regole aziendali).

○ Schema relazionale:

- ☐ Lo schema relazionale è *equivalente* allo schema ER ristrutturato: non si possono introdurre nuovi attributi/relazioni/vincoli se non quelli derivanti dalla traduzione dello schema ER ristrutturato. Se si scopre un errore o una lacuna, occorre correggere prima l’ER principale e l’ER ristrutturato.
- ☐ Per ogni relazione è indicata la chiave primaria.
- ☐ Per ogni relazione sono stati indicati i vincoli di integrità referenziale.
- ☐ Ogni associazione dello schema ER ristrutturato è stata tradotta nello schema relazionale rispettandone il tipo (uno a uno, uno a molti, molti a molti, ...).

- **Implementazione in SQL:**

- ☐ L'implementazione in SQL ha lo scopo di verificare l'effettiva utilizzabilità dello schema relazionale: non è richiesto lo sviluppo di un sistema completo e funzionante. Inoltre non è richiesta l'implementazione delle regole aziendali.