

Relazione Progetto Basi di Dati

A.A. 2021 - 2022

Antonutti Marco (142426), Candolo Vittorio Giorgio (141879),
Pipan Martin (151699), Pozzana Matteo (142250)

Ottobre 2022



Indice

1	Introduzione	4
2	Analisi dei requisiti	5
2.1	Requisiti forniti	5
2.2	Glossario	5
2.3	Strutturazione dei requisiti	6
2.4	Rappresentazione dei concetti	7
2.5	Integrazioni e riepilogo delle assunzioni fatte	7
3	Progettazione Concettuale	8
3.1	Diagramma ER	8
3.2	Vincoli aziendali	9
3.3	Regole di derivazione	9
4	Progettazione Logica	10
4.1	Analisi delle ridondanze	10
4.1.1	Analisi dei cicli	10
4.1.2	Requisiti operazionali	10
4.1.3	Tavola dei volumi	10
4.1.4	Analisi attributi derivabili: Numero Progetti	11
4.1.5	Analisi attributi derivabili: Numero Fornitori	12
4.2	Ristrutturazione del diagramma ER	13
4.2.1	Eliminazione delle generalizzazioni	13
4.2.2	Eliminazione degli attributi multivalore	13
4.2.3	Eliminazione degli attributi composti	13
4.3	Diagramma ER Ristrutturato	14
4.4	Traduzione al modello relazionale	15
4.4.1	Traduzione delle entità	15
4.4.2	Traduzione delle relazioni	15
4.5	Diagramma relazionale	16
4.6	Osservazioni	16
4.6.1	Vincolo sulle partecipazioni	16
4.6.2	Vincolo sugli impiegati segretari	16
5	Progettazione Fisica	17
5.1	Indici	17
5.1.1	Valutazione dell'indicizzazione di qualifica di impiegato	17
5.1.2	Valutazione dell'indicizzazione di data di assunzione di impiegato	20
5.1.3	Valutazione dell'indicizzazione di budget di progetto	23
6	Implementazione	26
6.1	Definizione delle enumerazioni	26
6.2	Definizione delle tabelle	26
6.3	Definizione delle viste	27
6.4	Definizione dei trigger	28
6.4.1	Vincolo aziendale relativo alla partecipazione ai progetti per città	28
6.4.2	Vincolo aziendale relativo agli impiegati con qualifica di segretario	29
6.4.3	Vincolo relativo ai matrimoni	31
6.4.4	Sincronizzazione attributo derivato numero fornitori di dipartimento	32
6.4.5	Sincronizzazione attributo derivato numero progetti di impiegato	33
6.5	Definizione degli indici	34
6.5.1	Indice su qualifica	34
6.5.2	Indice su data di assunzione	34
6.5.3	Indice su budget	34
6.6	Interrogazioni	34

6.6.1	Ricerca dei segretari che conoscono una determinata lingua	34
6.6.2	Ricerca degli impiegati con determinata competenza	34
6.6.3	Ricerca delle competenze di un impiegato	34
6.6.4	Ricerca degli impiegati coniugati	35
6.6.5	Ricerca dei laureati in una materia	35
6.6.6	Ricerca delle città in cui l'azienda opera	35
6.6.7	Ricerca del numero di dipendenti in una determinata città	35
6.6.8	Ricerca del numero di progetti a cui lavora un impiegato	35
6.6.9	Ricerca del numero di fornitori di un dipartimento	36
6.6.10	Assegnazione di un progetto ad un impiegato	36
6.6.11	Assegnazione di un fornitore a un dipartimento	36
6.6.12	Inserimento di un impiegato segretario	36
7	Analisi dei dati	37
7.1	Numero progetti per città	37
7.2	Distribuzione data di nascita per qualifica	38
7.3	Numero segretari per lingua	39
7.4	Distribuzione budget per città	40
7.5	Distribuzione partecipazioni a progetto con competenze	41
7.6	Distribuzione impiegati per dipartimento	42
7.7	Percentuale impiegati laureati	43
7.8	Percentuale impiegati per qualifica	44
8	Conclusione	45

1 Introduzione

L'elaborato tratta l'attività di progettazione e di implementazione di una base di dati relazionale per il sistema informativo di un'azienda.

Vengono analizzate in particolare le fasi di raccolta e analisi dei requisiti, progettazione concettuale, progettazione logica, progettazione fisica, implementazione e analisi dei dati.

2 Analisi dei requisiti

2.1 Requisiti forniti

Si è acquisita la descrizione del sistema e del dominio dalla consegna.

Vogliamo modellare le seguenti informazioni riguardanti gli impiegati di un'azienda, i dipartimenti a cui afferiscono, le competenze che possiedono e i progetti a cui partecipano.

Ogni impiegato ha una matricola, assegnatagli dalla società, che lo identifica univocamente. Di ogni impiegato interessano il nome e il cognome, la data di nascita e la data di assunzione. Se un impiegato è coniugato con un altro impiegato della stessa società, interessano la data del matrimonio e il coniuge. Ogni impiegato ha una qualifica (ad esempio, segretario, impiegato, programmatore, analista, progettista, ecc.). Dei laureati e dei segretari interessano anche altre informazioni. Dei laureati interessa il tipo di laurea e dei segretari le lingue conosciute.

La società è organizzata in dipartimenti. Ciascun dipartimento è identificato univocamente dal nome e possiede un recapito telefonico. Dipartimenti distinti hanno un diverso recapito telefonico. Ogni impiegato afferisce ad un unico dipartimento. Ogni dipartimento viene rifornito da vari fornitori e un fornitore può rifornire vari dipartimenti. Di ogni fornitore interessano il nome e l'indirizzo.

I progetti sono identificati da un numero e sono caratterizzati da una città e da un budget. Più impiegati possono essere coinvolti in uno stesso progetto. Un impiegato può partecipare a più progetti, ma può essere assegnato ad un unico progetto per città. Di ogni città con almeno un progetto, interessano il numero di residenti e la regione di appartenenza. Un impiegato può avere più competenze, ma usarne solo alcune per un particolare progetto. Un impiegato usa ogni sua competenza in almeno un progetto. Ad ogni competenza è assegnato un codice, che la identifica univocamente e una descrizione.

2.2 Glossario

Pur essendo le specifiche iniziali sostanzialmente univoche nell'uso dei termini, si è deciso di redigere comunque un glossario al fine di meglio analizzare le entità e le relazioni fra esse.

Termine	Descrizione	Sinonimi	Relazioni
Impiegato	Persona che svolge la propria attività professionale presso l'azienda		Dipartimento, Progetto, Competenza
Dipartimento	Struttura organizzativa interna all'azienda che svolge un determinato compito per l'azienda		Fornitore, Impiegato
Fornitore	Azienda esterna, rifornisce i dipartimenti		Dipartimento
Progetto	Serie di attività svolte dagli impiegati dell'azienda al fine di raggiungere un determinato obiettivo aziendale		Città, Competenza, Impiegato
Città	Centro abitato dove l'azienda svolge i suoi progetti		Progetto
Competenza	Abilità di un impiegato, viene usata per svolgere determinati compiti aziendali		Progetto, Impiegato
Laureato	Impiegato in possesso di una o più lauree		Impiegato (<i>Specializzazione</i>)
Segretario	Impiegato che svolge compiti amministrativi o di segreteria		Impiegato (<i>Specializzazione</i>)

2.3 Strutturazione dei requisiti

Eventuali integrazioni alla descrizione fornita o assunzioni fatte sono indicate in *corsivo*.

Impiegato
<ul style="list-style-type: none">• Identificato univocamente da una matricola e caratterizzato da nome, cognome, data di nascita, data di assunzione e qualifica.• Se coniugato con altro impiegato è ulteriormente caratterizzato dalla data di matrimonio col coniuge.• Se segretario è ulteriormente caratterizzato dalle lingue conosciute.• Se laureato è ulteriormente caratterizzato dal tipo di laurea.• Ogni impiegato afferisce ad un unico dipartimento.• Ogni impiegato può partecipare a più progetti ma ad uno solo per città.• Ogni impiegato usa ogni sua competenza in almeno un progetto.
Dipartimento
<ul style="list-style-type: none">• Identificato univocamente dal nome e caratterizzato da un recapito telefonico univoco.
Fornitore
<ul style="list-style-type: none">• <i>Identificato univocamente dalla partita IVA</i> e caratterizzato dal nome e dall'indirizzo.• Ogni fornitore può rifornire diversi dipartimenti.
Progetto
<ul style="list-style-type: none">• Identificato univocamente da un numero e caratterizzato dalla città dove si svolge e da un budget.
Città
<ul style="list-style-type: none">• <i>Identificata univocamente dal nome e dalla regione</i> e caratterizzata da numero di residenti e regione di appartenenza.
Competenza
<ul style="list-style-type: none">• Identificata univocamente da un codice e caratterizzata da una descrizione.

2.4 Rappresentazione dei concetti

Ogni **impiegato** ha una **matricola**, assegnatagli dalla società, che lo identifica univocamente.

Di ogni impiegato interessano il **nome** e il **cognome**, la **data di nascita** e la **data di assunzione**.

Se un impiegato è **coniugato** con un altro impiegato della stessa società, interessano la **data del matrimonio** e il coniuge.

Ogni impiegato ha una **qualifica** (ad esempio, segretario, impiegato, programmatore, analista, progettista, ecc.).

Dei **laureati** e dei **segretari** interessano anche altre informazioni.

Dei laureati interessa il **tipo di laurea** e dei segretari le **lingue conosciute**.

La società è organizzata in **dipartimenti**.

Ciascun dipartimento è identificato univocamente dal **nome** e possiede uno **recapito telefonico**.

Dipartimenti distinti hanno un diverso recapito telefonico.

Ogni impiegato **afferisce** ad un unico dipartimento.

Ogni dipartimento viene **rifornito** da vari **fornitori** e un fornitore può rifornire vari dipartimenti.

Di ogni fornitore interessano il **nome** e l'**indirizzo**.

I **progetti** sono identificati da un **numero** e sono caratterizzati da una **città** e da un **budget**.

Più impiegati possono essere coinvolti in uno stesso progetto.

Un impiegato può **partecipare** a più progetti, ma può essere assegnato ad un unico progetto per **città**.

Di ogni città con almeno un progetto, interessano il **numero** di residenti e la **regione** di appartenenza.

Un impiegato può **avere** più **competenze**, ma usarne solo alcune per un particolare progetto.

Un impiegato usa ogni sua competenza in almeno un progetto.

Ad ogni competenza è assegnato un **codice**, che la identifica univocamente, e una **descrizione**.

Legenda

Entità

Specializzazione

Attributo

Relazione

Attributo di relazione

2.5 Integrazioni e riepilogo delle assunzioni fatte

In conclusione della fase di analisi dei requisiti si è provveduto ad integrare la consegna di due attributi derivati, al fine di poter svolgere le opportune analisi e modellazioni nelle fasi successive del progetto:

Dipartimento - numero fornitori.

Impiegato - numero progetti.

Si vogliono inoltre ricapitolare le assunzioni e le scelte fatte per escludere eventuali ambiguità:

Le competenze degli impiegati costituiscono un'entità e sono da intendersi come applicazioni delle qualifiche.

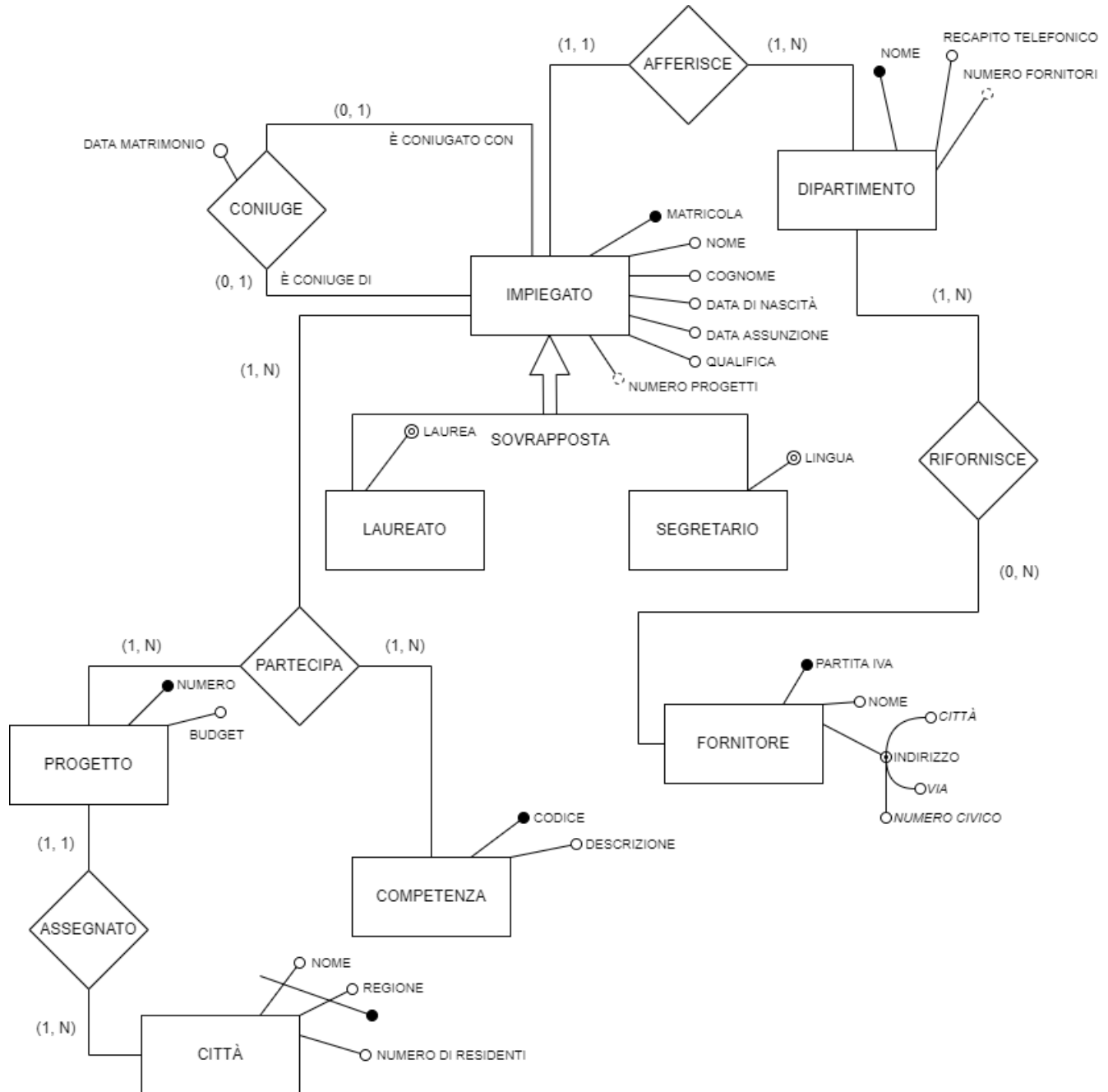
Le qualifiche invece rimangono attributo degli impiegati come richiesto dalla consegna.

Laureati e segretari sono specializzazioni dell'entità Impiegato.

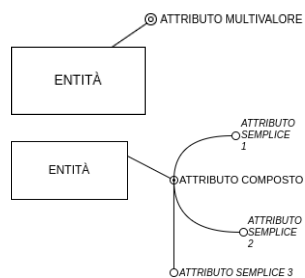
Tipo di laurea e lingue conosciute sono, rispettivamente, attributi della prima e della seconda specializzazione.

3 Progettazione Concettuale

3.1 Diagramma ER



Legenda



3.2 Vincoli aziendali

Al fine di codificare i vincoli imposti dai requisiti del progetto si è definito il seguente vincolo aziendale:

Un impiegato può partecipare a più progetti, ma non a più di uno per città.

3.3 Regole di derivazione

Si esplicitano poi le regole di derivazione per gli attributi derivati introdotti:

Dipendenti: numero progetti

Conta il numero di tuple della relazione partecipa in cui compare l'impiegato

Dipartimento: numero fornitori

Conta il numero di tuple della relazione rifornisce in cui compare il dipartimento

4 Progettazione Logica

4.1 Analisi delle ridondanze

4.1.1 Analisi dei cicli

Lo schema ER non presenta cicli fatta eccezione per la relazione ricorsiva. Tuttavia la configurazione scelta prevede il vincolo aziendale descritto in precedenza.

4.1.2 Requisiti operazionali

Partendo da quanto descritto nei requisiti e facendo delle assunzioni al fine di poter svolgere poi delle analisi significative, di seguito vengono formulate le principali operazioni, ciascuna con la rispettiva frequenza.

Operazione	Tipo	Frequenza
Ricerca dei fornitori di un dipartimento	Interattiva	100/mese
Ricerca dei segretari che conoscono una determinata lingua	Interattiva	15/mese
Ricerca degli impiegati con determinata competenza	Interattiva	200/mese
Inserimento di un nuovo progetto	Interattiva	20/mese
Ricerca delle competenze di un impiegato	Interattiva	200/mese
Ricerca di impiegati coniugati	Interattiva	10/mese
Ricerca di impiegati che partecipano ad un progetto nullo	Batch	4/mese
Assegnazione di un progetto ad un impiegato	Interattiva	200/mese
Assegnazione fornitore a dipartimento	Interattiva	20/mese
Ricerca tipo di laureato	Interattiva	50/settimana
Ricerca della città in cui l'azienda opera	Interattiva	25/mese
Ricerca del numero di dipendenti per una città	Interattiva	2/settimana
Ricerca del numero di progetti a cui lavora un impiegato	Batch	50/mese
Ricerca del numero di fornitori di un dipartimento	Batch	30/mese

4.1.3 Tavola dei volumi

Si definisce ora la tavola dei volumi al fine di determinare se mantenere o meno gli attributi derivati. Si assume lo stato della base di dati dopo 5 anni di utilizzo.

Concetto	Tipo	Volume
Impiegato	E	3000
Laureato	E	1000
Segretario	E	50
Dipartimento	E	60
Fornitore	E	100
Progetto	E	100
Competenza	E	30
Città	E	15
Coniugato	R	20
Afferisce	R	3000
Rifornisce	R	250
Partecipa	R	9000
Assegnato	R	100

4.1.4 Analisi attributi derivabili: Numero Progetti

Le operazioni frequenti che coinvolgono questo attributo sono la ricerca del numero di progetti a cui lavora un impiegato e l'assegnazione di un progetto ad un impiegato.

Lettura senza attributo derivato

Concetto	Tipo	Accessi	Tipo accesso
Partecipa	R	9000	R

Lettura con attributo derivato

Concetto	Tipo	Accessi	Tipo accesso
Impiegato	E	1	R

Scrittura senza attributo derivato

Concetto	Tipo	Accessi	Tipo accesso
Partecipa	R	3	W

Scrittura con attributo derivato

Concetto	Tipo	Accessi	Tipo accesso
Partecipa	R	3	W
Impiegato	E	1	W

Applicando alle scritture un peso doppio rispetto alle letture si ottengono i valori di seguito riportati.

	Costo senza attributo	Costo con attributo
Costo lettura	450000	50
Costo scrittura	1200	1600
Costo totale	45120	1650

Si conclude quindi di mantenere l'attributo derivato vista la netta prevalenza delle operazioni di lettura.

4.1.5 Analisi attributi derivabili: Numero Fornitori

Le operazioni frequenti che coinvolgono questo attributo sono la ricerca del numero di fornitori di un dipartimento e l'assegnazione di un fornitore ad un dipartimento.

Lettura senza attributo derivato

Concetto	Tipo	Accessi	Tipo accesso
Rifornisce	R	250	R

Lettura con attributo derivato

Concetto	Tipo	Accessi	Tipo accesso
Rifornisce	R	1	W
Dipartimento	E	1	W

Scrittura senza attributo derivato

Concetto	Tipo	Accessi	Tipo accesso
Rifornisce	R	1	W

Scrittura con attributo derivato

Concetto	Tipo	Accessi	Tipo accesso
Dipartimento	E	1	R

Applicando alle scritture un peso doppio rispetto alle letture si ottengono i valori di seguito riportati.

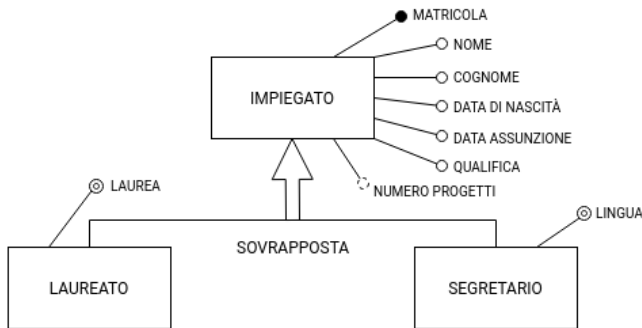
	Costo senza attributo	Costo con attributo
Costo lettura	7500	30
Costo scrittura	40	80
Costo totale	7540	110

Si conclude quindi di mantenere l'attributo derivato vista la netta prevalenza delle operazioni di lettura.

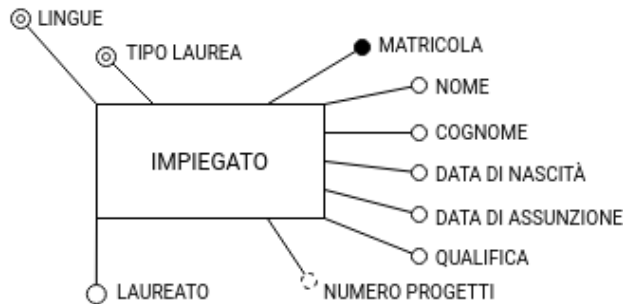
4.2 Ristrutturazione del diagramma ER

4.2.1 Eliminazione delle generalizzazioni

Il diagramma ER presenta due specializzazioni per l'entità impiegato:



Gli attributi caratteristici delle due classi di specializzazione vengono assegnati all'entità padre. Si rinomina l'attributo laurea in tipo laurea.

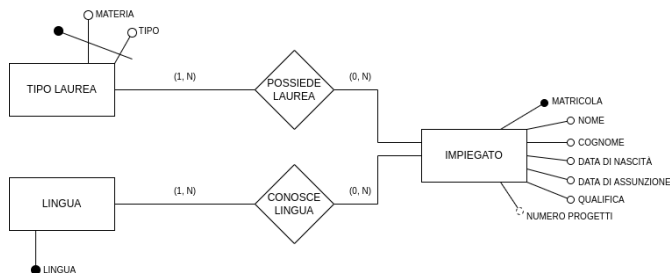


4.2.2 Eliminazione degli attributi multivalore

È ora necessario eliminare gli attributi multivalore.

Gli unici presenti sono gli attributi "lingue" e "tipo laurea" di impiegato. Si procede ad una reificazione.

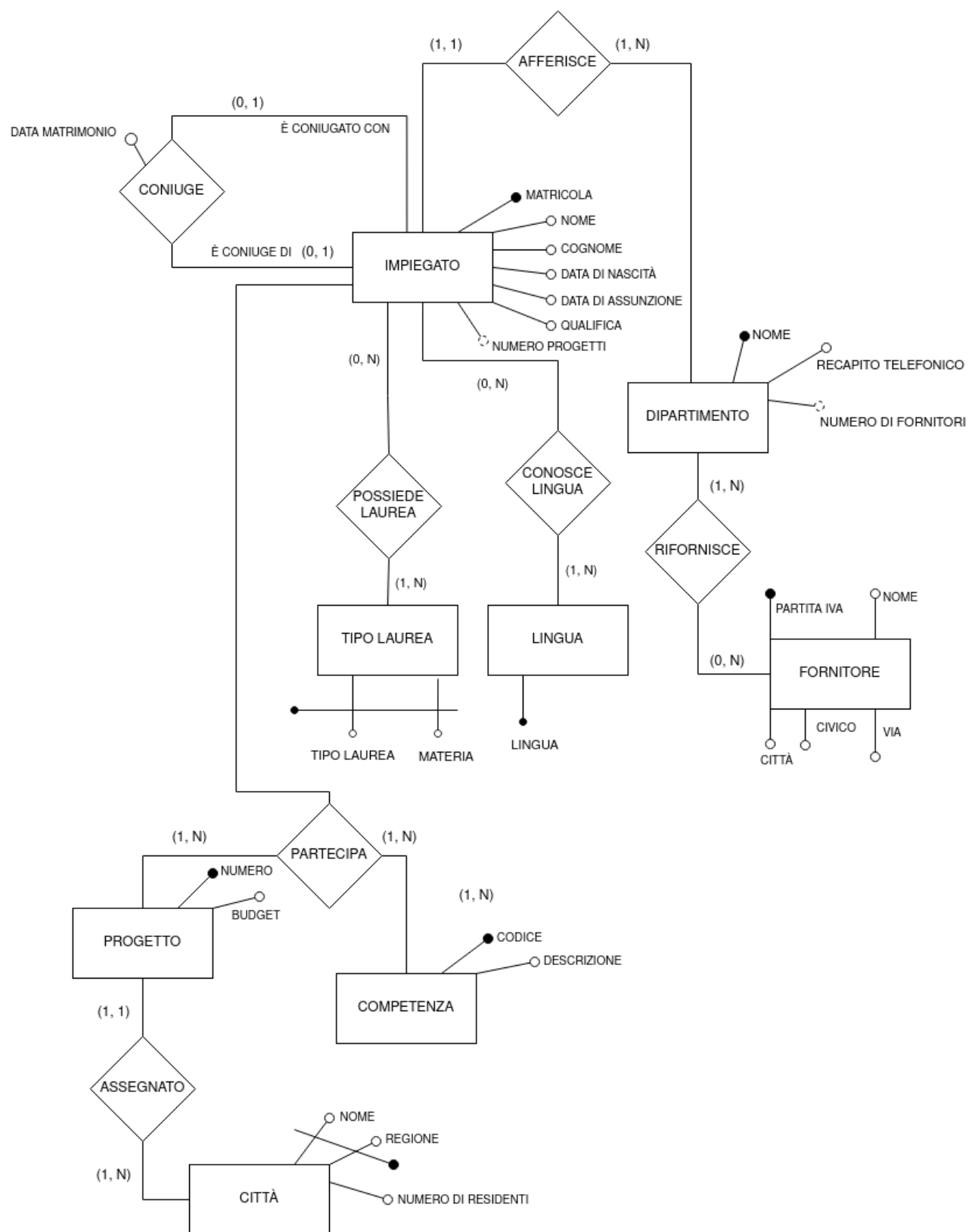
Nel caso di tipo laurea si è deciso inoltre di separare la materia dal livello (tipo).



4.2.3 Eliminazione degli attributi composti

L'unico attributo composto che compare nel diagramma ER è l'indirizzo dell'entità fornitore. Si è scelto di scorporare il summenzionato attributo nei campi componenti.

4.3 Diagramma ER Ristrutturato



4.4 Traduzione al modello relazionale

4.4.1 Traduzione delle entità

Fornitore (PartitaIva, Nome, Via, Civico, Città)

Dipartimento (Nome, RecapitoTelefonico, NumeroFornitori)

- RecapitoTelefonico attributo dal valore unico
- NumeroFornitori attributo derivato

Impiegato (Matricola, Nome, Cognome, DataDiNascita, DataDiAssunzione, Dipartimento, Qualifica, NumeroProgetti)

- Dipartimento chiave esterna (riferimento alla chiave primaria dell'entità Dipartimento)
- NumeroProgetti attributo derivato

Competenza (Codice, Descrizione)

- Descrizione unico

Città (Nome, Regione, NumeroDiResidenti)

Progetto (Numero, Budget, Città, Regione)

- Città chiave esterna (riferimento alla chiave primaria dell'entità Città)
- Regione chiave esterna (riferimento alla chiave primaria dell'entità Città)

Tipo Laurea

Si è deciso di codificare l'informazione direttamente come attributi della relazione Laureato.

Lingua

Si è deciso di codificare l'informazione direttamente come attributo della relazione Segretario.

4.4.2 Traduzione delle relazioni

Rifornisce (Dipartimento, Fornitore)

- Dipartimento chiave esterna (riferimento alla chiave primaria dell'entità Dipartimento)
- Fornitore chiave esterna (riferimento alla chiave primaria dell'entità Fornitore)

Segretario (Impiegato, Lingua) [Rinomina di Conosce Lingua]

- Impiegato (riferimento alla chiave primaria dell'entità Impiegato)
- Lingua chiave esterna (riferimento alla chiave primaria dell'entità Lingua)

Matrimonio (Marito, Moglie, DataDiMatrimonio) [Rinomina di Coniuge]

- Marito chiave esterna (riferimento alla chiave primaria dell'entità Impiegato)
- Moglie chiave esterna (riferimento alla chiave primaria dell'entità Impiegato)
- Moglie attributo dal valore unico
- Marito diverso da Moglie

Laureato (Impiegato, TipoLaurea, Materia) [Rinomina di Possiede Laurea]

- Impiegato chiave esterna (riferimento alla chiave primaria dell'entità Impiegato)

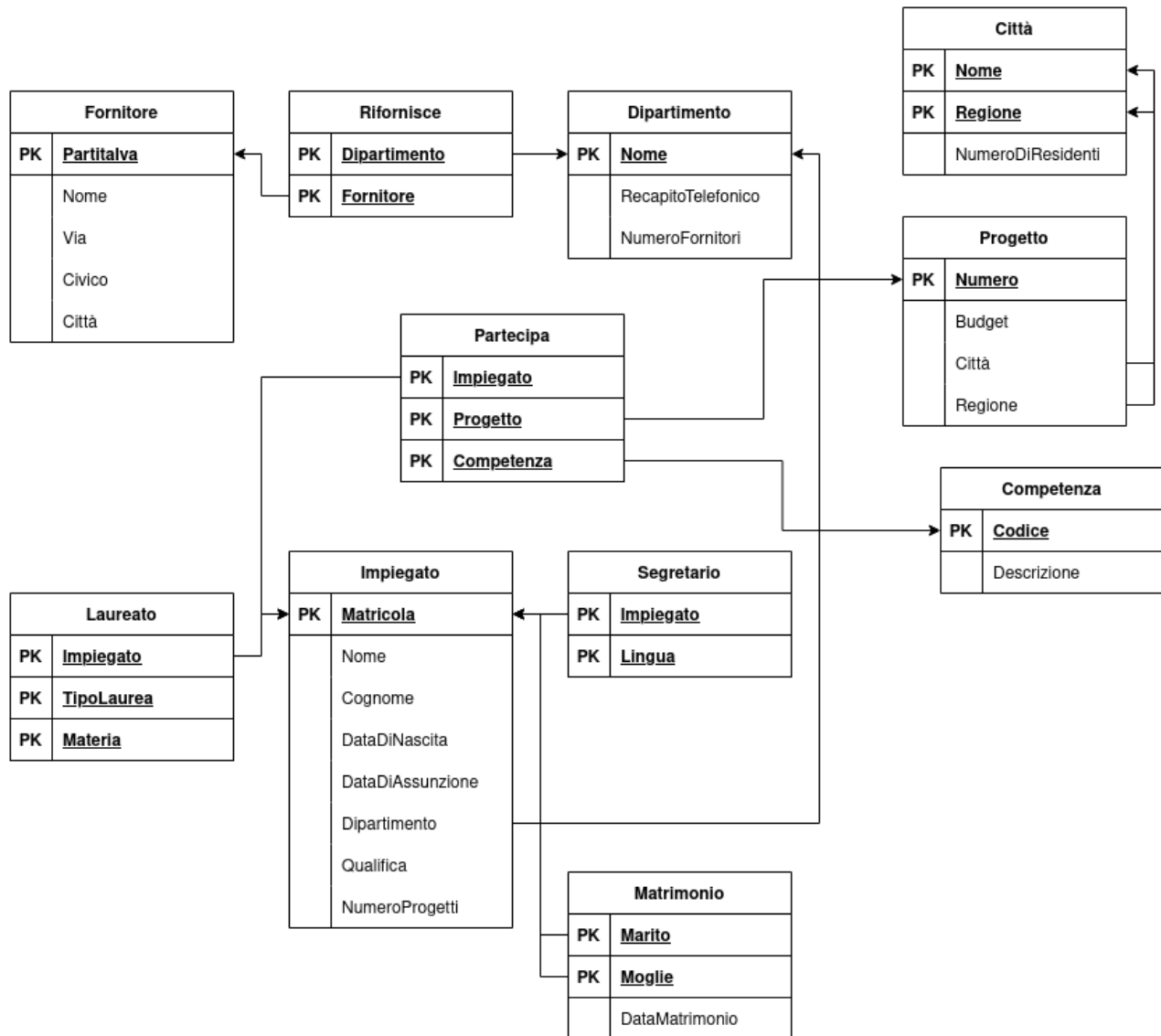
Partecipa (Impiegato, Competenza, Progetto)

- Impiegato chiave esterna (riferimento alla chiave primaria dell'entità Impiegato)
- Competenza chiave esterna (riferimento alla chiave primaria dell'entità Competenza)
- Progetto chiave esterna (riferimento alla chiave primaria dell'entità Progetto)

Afferisce

Si è deciso di codificare l'informazione direttamente come attributo dell'entità Impiegato.

4.5 Diagramma relazionale



4.6 Osservazioni

4.6.1 Vincolo sulle partecipazioni

Il vincolo aziendale relativo a progetti, impiegati e città non è codificabile nello schema.

4.6.2 Vincolo sugli impiegati segretari

Dato che a differenza di quanto accade per gli impiegati laureati, i requisiti forniti richiedono che segretario sia uno dei possibili valori dell'attributo *Qualifica* di **Impiegato**, è necessario implementare un ulteriore vincolo che garantisca che gli impiegati con *Qualifica* "Segretario" compaiano nella tabella **Segretario**.

Sono indicati in corsivo i nomi di attributo e in grassetto i nomi delle tabelle.

5 Progettazione Fisica

5.1 Indici

Si valuta in questa fase l'introduzione di indici per ottimizzare le prestazioni della base di dati.

Noto che le chiavi primarie e gli attributi unici delle tabelle sono già indicizzati dal DBMS PostgreSQL si è ritenuto di studiare l'uso di indici sugli attributi *Competenza* e *DataDiAssunzione* della tabella **Impiegato** e sull'attributo *Budget* della tabella **Progetto**.

Per valutare l'impatto degli indici sulle operazioni di lettura e scrittura si è fatto uso come tool di profiling delle interrogazioni del comando EXPLAIN ANALISE.

Tramite una semplice funzione vengono raccolti 50 tempi, con e senza indice.

Per motivi di presentazione nelle tabelle ne vengono mostrati 30 mentre tutti e 50 contribuiscono ai grafici.

A margine saranno presenti anche alcune valutazioni sulla dimensione delle tabelle.

5.1.1 Valutazione dell'indicizzazione di qualifica di impiegato

Operazioni di selezione

Interrogazione: `explain analyse select * from impiegato where qualifica='Programmatore'`

Planning senza indice	Execution senza indice	Planning con indice	Execution con indice
0.048	0.449	0.057	0.373
0.022	0.366	0.029	0.342
0.017	0.299	0.03	0.328
0.017	0.329	0.021	0.285
0.017	0.324	0.023	0.293
0.019	0.287	0.028	0.31
0.02	0.318	0.028	0.282
0.035	0.367	0.02	0.285
0.019	0.299	0.022	0.337
0.023	0.344	0.03	0.287
0.018	0.304	0.028	0.291
0.045	0.494	0.029	0.288
0.023	0.344	0.059	0.431
0.02	0.333	0.033	0.312
0.02	0.313	0.021	0.288
0.026	0.302	0.02	0.292
0.025	0.317	0.029	0.307
0.017	0.285	0.028	0.28
0.017	0.283	0.02	0.288
0.017	0.279	0.023	0.31
0.016	0.278	0.031	0.294
0.017	0.277	0.02	0.29
0.017	0.311	0.026	0.29
0.017	0.284	0.029	0.285
0.016	0.277	0.02	0.291
0.017	0.306	0.02	0.288
0.023	0.27	0.021	0.306
0.047	0.433	0.02	0.293
0.029	0.365	0.021	0.289
0.021	0.367	0.027	0.303

Operazioni di modifica

Interrogazione:

```
explain analyse update impiegato set qualifica='Analista' where numero_progetti between 2 and 3
```

Planning senza indice	Execution senza indice	Planning con indice	Execution con indice
0.07	5.044	0.072	4.841
0.069	6.875	0.074	4.988
0.064	5.414	0.062	4.992
0.094	6.242	0.065	4.801
0.067	4.806	0.111	5.125
0.086	4.958	0.055	5.225
0.058	4.586	0.046	4.821
0.04	4.538	0.143	5.447
0.077	5.114	0.086	5.777
0.049	4.981	0.077	5.946
0.065	4.731	0.058	5.284
0.048	4.92	0.051	5.815
0.065	4.819	0.152	5.433
0.042	4.507	0.071	5.827
0.043	4.755	0.05	5.552
0.035	4.589	0.041	5.236
0.039	4.669	0.05	5.247
0.043	4.574	0.049	5.308
0.032	4.493	0.05	5.039
0.039	4.474	0.045	5.247
0.04	4.771	0.035	5.23
0.04	4.49	0.059	5.107
0.031	4.547	0.036	4.863
0.031	4.625	0.033	5.094
0.031	4.571	0.035	4.873
0.039	4.441	0.033	4.976
0.04	4.622	0.033	4.998
0.041	4.485	0.033	4.88
0.032	4.657	0.033	4.853
0.043	4.61	0.044	5.175

Considerazioni sullo spazio occupato

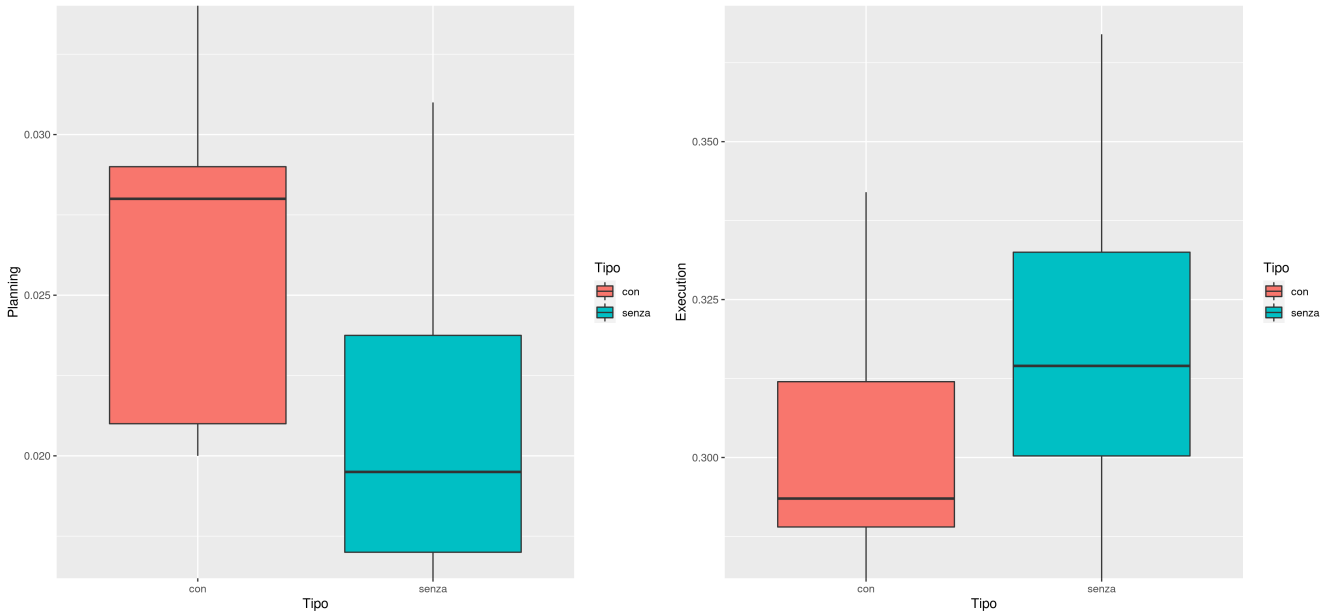
Sfruttando le seguenti interrogazioni è possibile conoscere lo spazio occupato dalla tabella e dagli indici:

```
SELECT pg_size_pretty(pg_total_relation_size ('impiegato'));  
SELECT pg_size_pretty(pg_indexes_size ('impiegato'));
```

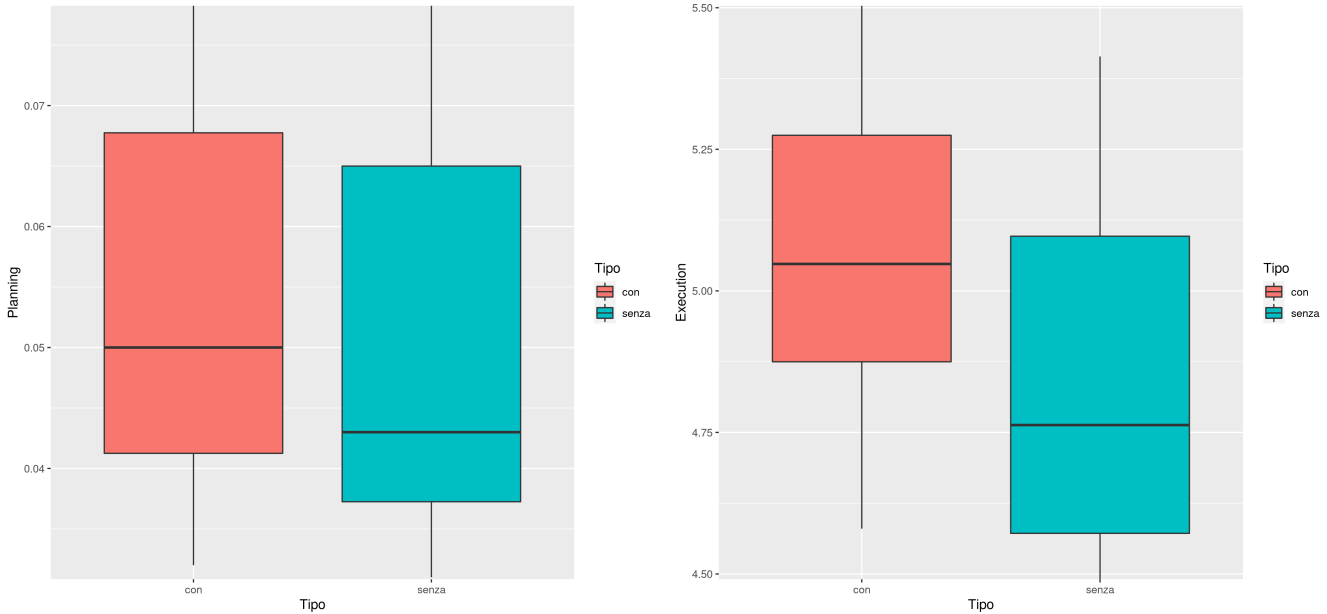
Per impiegato il risultato è di **2128kB** prima dell'indicizzazione e **2224kB** a seguito di essa con una dimensione dell'indice di **96kB**.

Concludiamo visualizzando i risultati della profilazione tramite box plot realizzati in R:

Operazioni di selezione



Operazioni di modifica



Visto l'esito del profiling si ritiene di implementare l'indice.

5.1.2 Valutazione dell'indicizzazione di data di assunzione di impiegato

Operazioni di selezione

Interrogazione: `explain analyse select * from impiegato where data_di_assunzione<='20000101'`

Planning senza indice	Execution senza indice	Planning con indice	Execution con indice
0.084	0.896	0.092	0.014
0.04	0.778	0.053	0.019
0.026	0.356	0.06	0.01
0.036	0.377	0.061	0.01
0.024	0.33	0.061	0.01
0.036	0.338	0.059	0.01
0.052	0.452	0.064	0.012
0.026	0.548	0.068	0.011
0.116	1.23	0.117	0.014
0.054	0.471	0.08	0.012
0.03	0.324	0.061	0.01
0.028	0.293	0.081	0.027
0.018	0.277	0.061	0.011
0.019	0.294	0.09	0.029
0.018	0.271	0.072	0.013
0.029	0.386	0.06	0.01
0.019	0.27	0.059	0.011
0.036	0.465	0.112	0.021
0.02	0.331	0.089	0.022
0.03	0.316	0.065	0.011
0.04	0.524	0.064	0.014
0.027	0.366	0.103	0.013
0.033	0.539	0.107	0.015
0.029	0.341	0.095	0.023
0.027	0.352	0.046	0.01
0.037	0.317	0.051	0.009
0.034	0.384	0.076	0.01
0.052	0.45	0.075	0.016
0.027	0.428	0.071	0.012
0.045	0.711	0.08	0.012

Operazioni di modifica

Interrogazione:

```
explain analyse
```

```
update impiegato
```

```
set data_di_assunzione='20220829' where data_di_nascita between '19820101' and '20050101'
```

Planning senza indice	Execution senza indice	Planning con indice	Execution con indice
0.069	7.221	0.073	7.519
0.143	7.546	0.066	6.17
0.097	7.623	0.055	6.281
0.124	7.475	0.107	7.559
0.072	7.268	0.068	7.164
0.068	7.978	0.072	6.133
0.113	12.945	0.101	6.604
0.065	6.818	0.051	6.122
0.046	6.371	0.046	6.782
0.043	6.328	0.048	6.263
0.044	6.414	0.035	6.486
0.074	8.316	0.037	6.531
0.081	6.918	0.037	6.479
0.062	7.582	0.047	6.468
0.081	6.874	0.036	6.444
0.046	6.597	0.034	6.547
0.037	6.201	0.049	6.548
0.046	6.235	0.043	6.966
0.083	4.914	0.071	6.882
0.029	4.792	0.084	6.668
0.028	4.691	0.083	6.987
0.027	4.609	0.098	6.878
0.034	4.585	0.08	6.891
0.035	4.651	0.079	6.878
0.026	4.521	0.072	7.138
0.035	4.61	0.091	7.488
0.035	4.653	0.069	7.252
0.045	5.002	0.072	6.607
0.029	4.568	0.073	6.82
0.052	5.783	0.118	7.071

Considerazioni sullo spazio occupato

Sfruttando le seguenti interrogazioni è possibile conoscere lo spazio occupato dalla tabella e dagli indici:

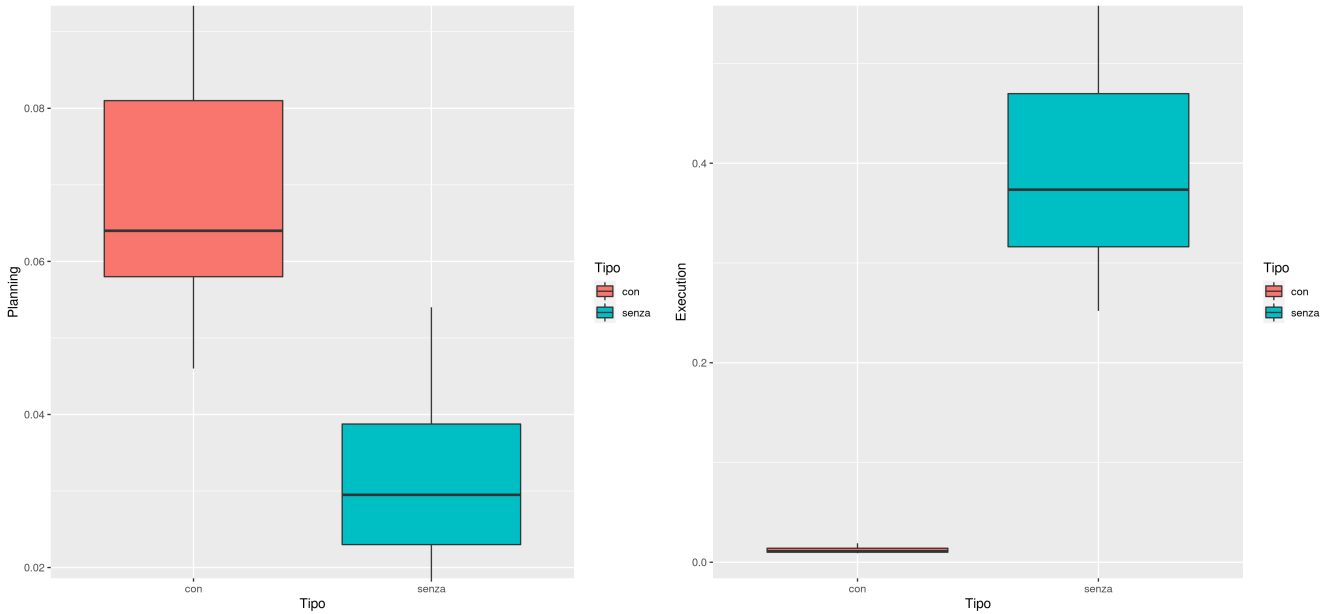
```
SELECT pg_size_pretty(pg_total_relation_size ('impiegato'));
```

```
SELECT pg_size_pretty(pg_indexes_size ('impiegato'));
```

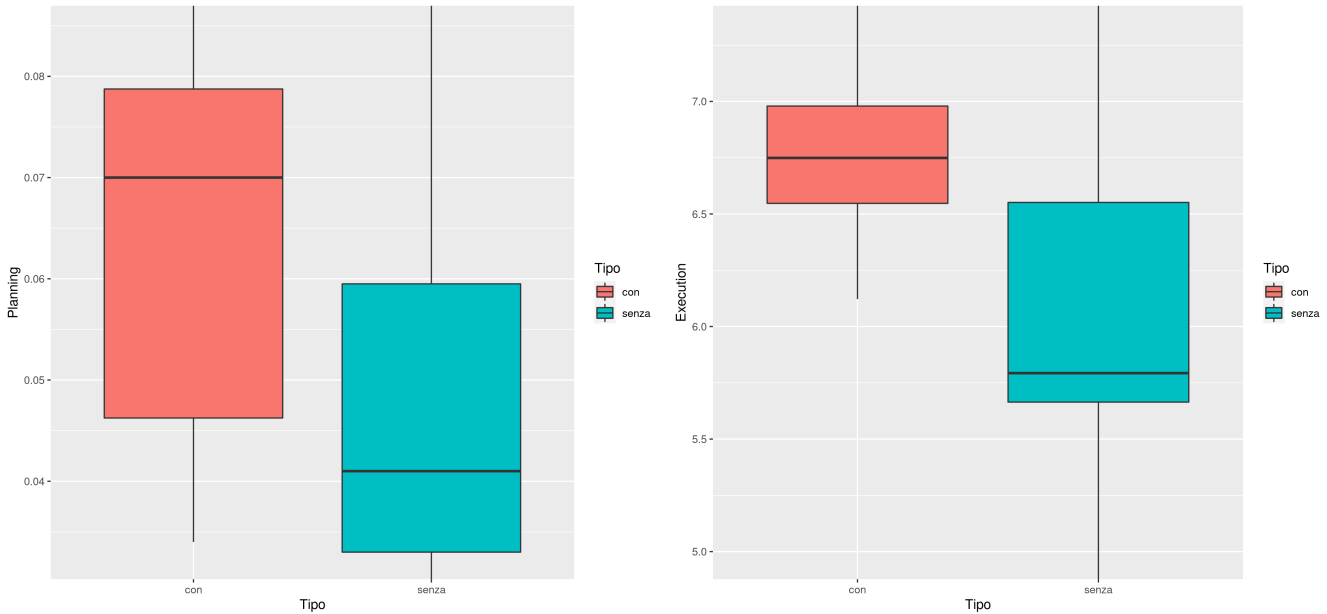
Per impiegato il risultato è di **2224kB** prima dell'indicizzazione e **2312kB** a seguito di essa con una dimensione dell'indice di **88kB**.

Concludiamo visualizzando i risultati della profilazione tramite box plot realizzati in R:

Operazioni di selezione



Operazioni di modifica



Visto l'esito del profiling e data la natura dell'attributo su cui si è costruito l'indice si ritiene di implementarlo.

5.1.3 Valutazione dell'indicizzazione di budget di progetto

Operazioni di selezione

Interrogazione: `explain analyse select * from progetto where budget>=100000`

Planning senza indice	Execution senza indice	Planning con indice	Execution con indice
0.048	0.042	0.078	0.036
0.041	0.048	0.037	0.03
0.024	0.041	0.05	0.022
0.027	0.035	0.041	0.022
0.019	0.033	0.044	0.024
0.02	0.033	0.059	0.046
0.019	0.023	0.069	0.041
0.02	0.03	0.041	0.022
0.019	0.031	0.041	0.022
0.022	0.031	0.032	0.029
0.037	0.041	0.04	0.022
0.026	0.041	0.046	0.026
0.027	0.023	0.038	0.027
0.027	0.042	0.031	0.031
0.072	0.052	0.035	0.018
0.034	0.049	0.089	0.051
0.04	0.042	0.035	0.03
0.02	0.023	0.045	0.033
0.027	0.023	0.049	0.021
0.027	0.047	0.043	0.029
0.034	0.04	0.044	0.033
0.021	0.032	0.027	0.02
0.02	0.031	0.03	0.025
0.03	0.026	0.034	0.018
0.02	0.025	0.049	0.031
0.032	0.028	0.056	0.031
0.027	0.032	0.044	0.026
0.022	0.025	0.041	0.023
0.027	0.037	0.047	0.029
0.036	0.042	0.061	0.045

Operazioni di modifica

Interrogazione: `explain analyse update progetto set budget=budget*0.75 where città='Atlanta'`

Planning senza indice	Execution senza indice	Planning con indice	Execution con indice
0.065	0.101	0.055	0.088
0.063	0.08	0.035	0.07
0.046	0.079	0.029	0.062
0.034	0.072	0.027	0.061
0.04	0.07	0.035	0.062
0.036	0.064	0.034	0.06
0.038	0.064	0.027	0.06
0.041	0.071	0.035	0.063
0.045	0.085	0.033	0.061
0.039	0.075	0.033	0.061
0.043	0.091	0.034	0.061
0.034	0.084	0.033	0.06
0.03	0.076	0.035	0.062
0.035	0.096	0.051	0.074
0.026	0.057	0.051	0.125
0.039	0.054	0.046	0.077
0.025	0.058	0.057	0.112
0.024	0.054	0.037	0.075
0.03	0.049	0.037	0.067
0.03	0.053	0.039	0.072
0.024	0.054	0.042	0.073
0.026	0.056	0.103	0.072
0.056	0.093	0.056	0.095
0.038	0.074	0.078	0.136
0.04	0.066	0.052	0.125
0.056	0.093	0.035	0.062
0.042	0.075	0.035	0.061
0.047	0.091	0.026	0.064
0.073	0.13	0.035	0.064
0.05	0.112	0.034	0.101

Considerazioni sullo spazio occupato

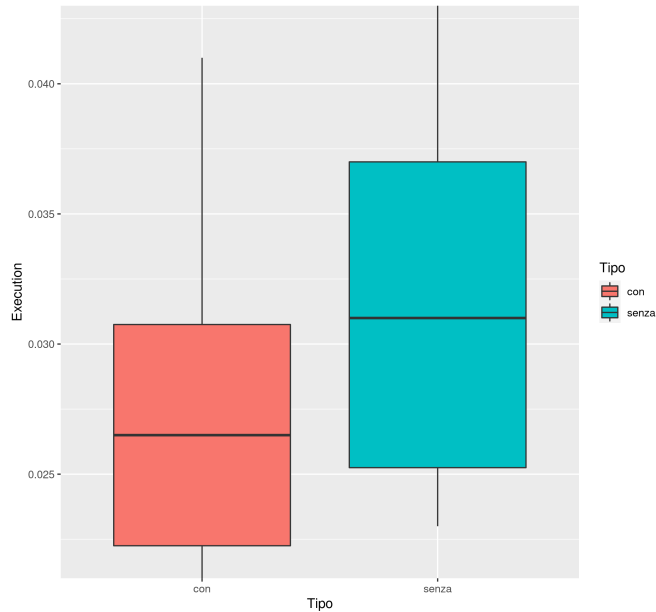
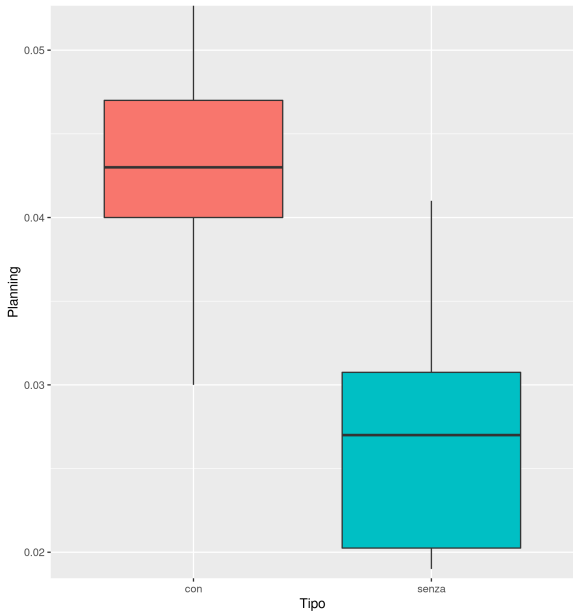
Sfruttando le seguenti interrogazioni è possibile conoscere lo spazio occupato dalla tabella e dagli indici:

```
SELECT pg_size_pretty(pg_total_relation_size ('progetto'));  
SELECT pg_size_pretty(pg_indexes_size ('progetto'));
```

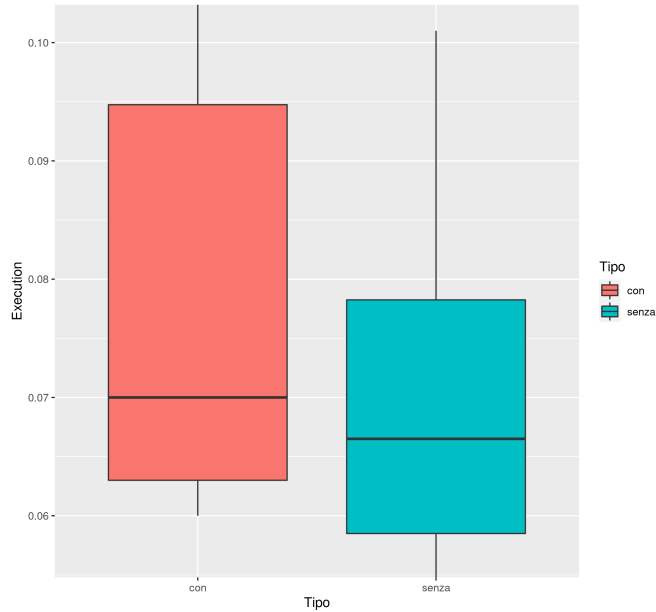
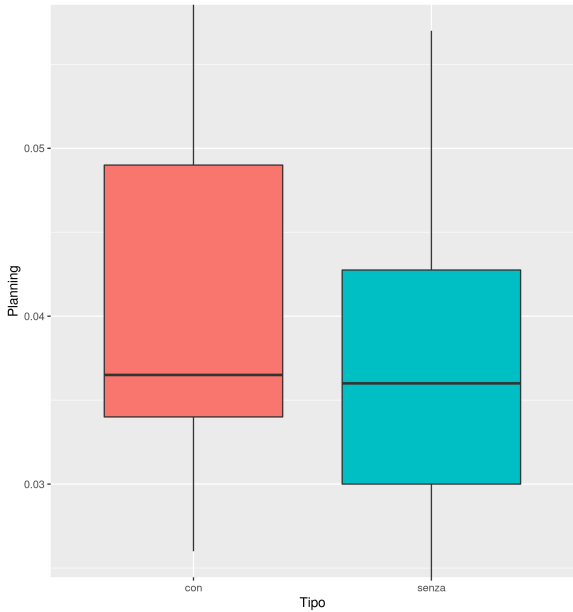
Per impiegato il risultato è di **72kB** prima dell'indicizzazione e **88kB** a seguito di essa con una dimensione dell'indice di **16kB**.

Concludiamo visualizzando i risultati della profilazione tramite box plot

Operazioni di selezione



Operazioni di modifica



Visto l'esito del profiling si ritiene di implementare l'indice.

6 Implementazione

6.1 Definizione delle enumerazioni

Si è valutato di sfruttare un'enumerazione per la codifica delle lingue conosciuta dal segretario. Questa soluzione garantisce che il valore dell'attributo appartenga ad un insieme prestabilito.

I comandi per gestire il tipo enumerazione sono semplici ma è importante notare che PostgreSQL permette l'aggiunta di nuovi valori all'enumerazione e la modifica di quelli esistenti ma non permette di rimuovere valori.

```
CREATE TYPE lingua AS enum (  
    'Inglese', 'Tedesco', 'Francese',  
    'Polacco', 'Russo', 'Arabo', 'Cinese',  
    'Giapponese', 'Coreano', 'Portoghese',  
    'Svedese', 'Indiano', 'Italiano'  
);
```

6.2 Definizione delle tabelle

Seguono le tabelle definite sulla base dello schema relazionale.

```
CREATE TABLE fornitore (  
    partitaiva numeric(11, 0) PRIMARY KEY,  
    nome varchar(50),  
    via varchar(50),  
    civico integer,  
    citta varchar(30)  
);
```

```
CREATE TABLE dipartimento (  
    nome varchar(50) PRIMARY KEY,  
    recapito_telefonico numeric(10, 0) UNIQUE,  
    numero_fornitori integer NOT NULL DEFAULT 0  
);
```

```
CREATE TABLE rifornisce (  
    dipartimento varchar(50) REFERENCES dipartimento(nome) ON UPDATE CASCADE ON DELETE CASCADE,  
    fornitore numeric(11, 0) REFERENCES fornitore(partitaiva) ON UPDATE CASCADE ON DELETE CASCADE,  
    PRIMARY KEY (dipartimento, fornitore)  
);
```

```
CREATE TABLE impiegato (  
    matricola integer PRIMARY KEY,  
    nome varchar(20),  
    cognome varchar(20),  
    data_di_nascita date,  
    data_di_assunzione date,  
    dipartimento varchar(50) REFERENCES dipartimento(nome) ON UPDATE CASCADE ON DELETE RESTRICT,  
    qualifica varchar(20),  
    numero_progetti integer NOT null DEFAULT 0  
);
```

```
CREATE TABLE segretario (  
    impiegato integer REFERENCES impiegato(matricola) ON UPDATE CASCADE ON DELETE CASCADE,  
    lingua lingua,  
    PRIMARY KEY (impiegato, lingua)  
);
```

```

CREATE TABLE competenza (
    codice integer PRIMARY KEY,
    descrizione varchar (50) UNIQUE
);

CREATE TABLE citta (
    nome varchar (20),
    regione varchar (20),
    numero_di_residenti integer,
    PRIMARY KEY (nome, regione)
);

CREATE TABLE progetto (
    numero integer PRIMARY KEY,
    budget integer,
    citta varchar(20),
    regione varchar(20),
    FOREIGN KEY(citta, regione) REFERENCES citta(nome, regione) ON UPDATE CASCADE ON DELETE RESTRICT
);

CREATE TABLE matrimonio (
    marito integer REFERENCES impiegato (matricola) ON UPDATE CASCADE ON DELETE CASCADE PRIMARY KEY,
    moglie integer REFERENCES impiegato(matricola) ON UPDATE CASCADE ON DELETE CASCADE UNIQUE NOT null,
    data_di_matrimonio date,
    CONSTRAINT marito_moglie_diversi CHECK(marito <> moglie)
);

CREATE TABLE laureato (
    impiegato integer REFERENCES impiegato(matricola) ON UPDATE CASCADE ON DELETE CASCADE,
    tipo_laurea varchar (20),
    materia varchar (50),
    PRIMARY KEY (impiegato, tipo_laurea, materia)
);

CREATE TABLE partecipa (
    impiegato integer REFERENCES impiegato(matricola) ON UPDATE CASCADE ON DELETE CASCADE,
    competenza integer REFERENCES competenza(codice) ON UPDATE CASCADE ON DELETE RESTRICT,
    progetto integer REFERENCES progetto(numero) ON UPDATE CASCADE ON DELETE RESTRICT,
    PRIMARY KEY(impiegato, competenza, progetto)
);

```

6.3 Definizione delle viste

Segue la definizione di una vista che è stata formulata per facilitare alcune interrogazioni e l'implementazione del vincolo aziendale relativo alla partecipazione ai progetti da parte degli impiegati.

```

CREATE VIEW cittaprogetto(progetto, citta, regione) AS
SELECT
    p.numero,
    ct.nome,
    ct.regione
FROM
    progetto p
    JOIN citta ct ON p.citta = ct.nome
    AND p.regione = ct.regione;

```

6.4 Definizione dei trigger

Segue la definizione dei trigger relativi ai vincoli aziendali e alla consistenza della base di dati.

6.4.1 Vincolo aziendale relativo alla partecipazione ai progetti per città

La funzione con trigger implementa il primo vincolo aziendale.

```
CREATE
OR REPLACE FUNCTION max_progetto_citta() RETURNS TRIGGER LANGUAGE plpgsql AS $$ BEGIN IF (
    new.competenza <> old.competenza
    AND (
        new.progetto = old.progetto
        AND new.impiegato = old.impiegato
    )
) THEN RETURN new;
END IF;
IF (
    NOT EXISTS (
        SELECT
            *
        FROM
            progetto,
            partecipa,
            cittaprogetto
        WHERE
            new.progetto = progetto.numero
            and new.impiegato = partecipa.impiegato
            and partecipa.progetto = cittaprogetto.progetto
            and progetto.citta = cittaprogetto.citta
    )
) THEN RETURN new;
END IF;
Return old;
END $$;

CREATE TRIGGER max_progetto_citta before
INSERT
OR
UPDATE
ON partecipa FOR each ROW execute procedure max_progetto_citta();
```

6.4.2 Vincolo aziendale relativo agli impiegati con qualifica di segretario

Le funzioni seguenti implementano il vincolo aziendale sugli impiegati con qualifica di segretari con un'ulteriore garanzia relativa alla tabella **segretario**.

Vincolo su impiegato

Un impiegato segretario deve conoscere almeno una lingua quindi la seguente funzione con trigger impedisce l'inserimento di impiegati con qualifica "Segretario", o l'attribuzione della stessa a impiegati che ne possedevano una diversa, se l'impiegato non è già presente nella tabella **segretario** dove la lingua fa parte della chiave primaria.

```
CREATE
OR REPLACE FUNCTION impiegato_segretario() RETURNS TRIGGER LANGUAGE plpgsql AS $$ BEGIN IF (
    new.qualifica = 'Segretario'
    AND (
        NOT EXISTS (
            SELECT
                *
            FROM
                segretario
            WHERE
                impiegato = new.matricola
        )
    )
) THEN RETURN old;
END IF;
RETURN new;
END $$;

CREATE TRIGGER impiegato_segretario BEFORE
INSERT
OR
UPDATE
ON impiegato FOR EACH ROW EXECUTE PROCEDURE impiegato_segretario();
```

Vincolo su segretario

La funzione, questa volta con un check integrato nella tabella, implementa un ulteriore vincolo la cui utilità è associata a quella del precedente. Questo vincolo è necessario a garantire che nella tabella segretario siano presenti solo impiegati con tale qualifica.

```
CREATE
OR REPLACE FUNCTION vincolo_segretario(m integer) RETURNS boolean LANGUAGE plpgsql AS $$ BEGIN IF (
    m NOT IN (
        SELECT
            matricola
        FROM
            impiegato
        WHERE
            qualifica = 'Segretario'
        OR
            qualifica IS NULL
    )
) THEN RETURN false;
END IF;
RETURN true;
END $$;

ALTER TABLE segretario
    ADD CONSTRAINT vincolo_segretario
        CHECK (vincolo_segretario(impiegato));
```

Come è facile intuire la coesistenza dei due trigger precedenti non permette l'inserimento di impiegati con qualifica di segretario o di segretari.

Noto che PostgreSQL non permette il deferimento di vincoli di tipo CHECK (e NOT NULL ma non è questo il caso) e volendo rispettare a pieno le specifiche che prevedevano "Segretario" come possibile valore dell'attributo qualifica, si è optato per l'utilizzo del valore temporaneo null come "jolly" per permettere gli inserimenti senza rinunciare a nessuno dei due vincoli.

Il comportamento della base di dati può essere chiarito visionando la transazione relativa all'inserimento di impiegati segretari.

6.4.3 Vincolo relativo ai matrimoni

La funzione con trigger implementa un vincolo sulla tabella matrimonio che previene ad esempio che un impiegato risulti contemporaneamente sposato con più impiegati diversi o che sia possibile inserire coppie permutate. L'eventualità dell'impiegato sposato con se stesso era già gestita con un constraint a livello di tabella

```
CREATE
OR REPLACE FUNCTION matrimonio_unico() RETURNS TRIGGER LANGUAGE plpgsql AS $$ BEGIN IF (
    EXISTS (
        SELECT
            *
        FROM
            matrimonio
        WHERE
            marito = new.marito
            OR marito = new.moglie
            OR moglie = new.marito
            OR moglie = new.moglie
    )
) THEN RETURN old;
END IF;
RETURN new;
END $$;
```

```
CREATE TRIGGER matrimonio_unico before
INSERT
OR
UPDATE
ON matrimonio FOR each ROW execute procedure matrimonio_unico();
```

6.4.4 Sincronizzazione attributo derivato numero fornitori di dipartimento

Seguono funzioni e trigger necessari per la gestione del primo attributo derivato.

```
CREATE
OR replace FUNCTION numero_fornitori_inc() RETURNS TRIGGER LANGUAGE plpgsql AS $$ BEGIN
UPDATE
    dipartimento
SET
    numero_fornitori = numero_fornitori + 1
WHERE
    nome = new.dipartimento;
RETURN new;
END;
$$;
```

```
CREATE TRIGGER numero_fornitori_inc before
INSERT
    ON rifornisce FOR each ROW execute procedure numero_fornitori_inc();
```

```
CREATE
OR replace FUNCTION numero_fornitori_dec() RETURNS TRIGGER LANGUAGE plpgsql AS $$ BEGIN
UPDATE
    dipartimento
SET
    numero_fornitori = numero_fornitori - 1
WHERE
    nome = old.dipartimento;
RETURN old;
END;
$$;
```

```
CREATE TRIGGER numero_fornitori_dec before
DELETE
    ON rifornisce FOR each ROW execute procedure numero_fornitori_dec();
```

```
CREATE
OR replace FUNCTION numero_fornitori_update() RETURNS TRIGGER LANGUAGE plpgsql AS $$ BEGIN
UPDATE
    dipartimento
SET
    numero_fornitori = numero_fornitori + 1
WHERE
    nome = new.dipartimento;
UPDATE
    dipartimento
SET
    numero_fornitori = numero_fornitori - 1
WHERE
    nome = old.dipartimento;
RETURN new;
END;
$$;
```

```
CREATE TRIGGER numero_fornitori_update before
UPDATE
    ON rifornisce FOR each ROW execute procedure numero_fornitori_update();
```


6.4.5 Sincronizzazione attributo derivato numero progetti di impiegato

Seguono funzioni e trigger necessari per la gestione del secondo attributo derivato.

```
CREATE
OR replace FUNCTION numero_progetti_inc() RETURNS TRIGGER LANGUAGE plpgsql AS $$ BEGIN
UPDATE
    impiegato
SET
    numero_progetti = numero_progetti + 1
WHERE
    matricola = new.impiegato;
RETURN new;
END;
$$;

CREATE TRIGGER numero_progetti_inc before
INSERT
    ON partecipa FOR each ROW execute procedure numero_progetti_inc();

CREATE
OR replace FUNCTION numero_progetti_dec() RETURNS TRIGGER LANGUAGE plpgsql AS $$ BEGIN
UPDATE
    impiegato
SET
    numero_progetti = numero_progetti - 1
WHERE
    matricola = old.impiegato;
RETURN old;
END;
$$;

CREATE TRIGGER numero_progetti_dec before
DELETE
    ON partecipa FOR each ROW execute procedure numero_progetti_dec();

CREATE
OR replace FUNCTION numero_progetti_update() RETURNS TRIGGER LANGUAGE plpgsql AS $$ BEGIN
UPDATE
    impiegato
SET
    numero_progetti = numero_progetti + 1
WHERE
    matricola = new.impiegato;
UPDATE
    impiegato
SET
    numero_progetti = numero_progetti - 1
WHERE
    matricola = old.impiegato;
RETURN new;
END;
$$;
CREATE TRIGGER numero_progetti_update before
UPDATE
    ON partecipa FOR each ROW execute procedure numero_progetti_update();
```

6.5 Definizione degli indici

Segue la definizione degli indici analizzati in fase di **progettazione fisica**

6.5.1 Indice su qualifica

```
CREATE INDEX index_impiegato_qualifica ON impiegato(qualifica);
```

6.5.2 Indice su data di assunzione

```
CREATE INDEX index_impiegato_dataAssunzione ON impiegato(data_di_assunzione);
```

6.5.3 Indice su budget

```
CREATE INDEX index_progetto_budget ON progetto(budget);
```

6.6 Interrogazioni

Vengono presentate alcune interrogazioni rilevanti costruite sulla base dei **requisiti operazionali**

6.6.1 Ricerca dei segretari che conoscono una determinata lingua

```
SELECT
    impiegato.matricola,
    impiegato.nome,
    impiegato.cognome
FROM
    impiegato,
    segretario
WHERE
    impiegato.matricola = segretario.impiegato
    AND lingua = 'Russo';
```

6.6.2 Ricerca degli impiegati con determinata competenza

```
SELECT
    DISTINCT matricola
FROM
    impiegato
    JOIN partecipa ON impiegato.matricola = partecipa.impiegato
WHERE
    partecipa.competenza = 824;
```

6.6.3 Ricerca delle competenze di un impiegato

```
SELECT
    competenza
FROM
    partecipa
WHERE
    impiegato = 69992;
```

6.6.4 Ricerca degli impiegati coniugati

```
SELECT
    matricola
FROM
    impiegato
WHERE
    matricola IN (
        SELECT
            marito
        FROM
            matrimonio
    )
    OR matricola IN (
        SELECT
            moglie
        FROM
            matrimonio
    );
```

6.6.5 Ricerca dei laureati in una materia

```
SELECT
    impiegato
FROM
    laureato
WHERE
    materia = 'Physics';
```

6.6.6 Ricerca delle città in cui l'azienda opera

```
SELECT
    DISTINCT citta
FROM
    progetto;
```

6.6.7 Ricerca del numero di dipendenti in una determinata città

```
SELECT
    COUNT(DISTINCT impiegato)
FROM
    partecipa AS p
    JOIN progetto AS ptt ON p.progetto = ptt.numero
WHERE
    citta = 'El Paso';
```

6.6.8 Ricerca del numero di progetti a cui lavora un impiegato

```
SELECT
    numero_progetti
FROM
    impiegato
WHERE
    matricola = 69992;
```

6.6.9 Ricerca del numero di fornitori di un dipartimento

```
SELECT
    numero_fornitori
FROM
    dipartimento
WHERE
    nome = 'B028';
```

6.6.10 Assegnazione di un progetto ad un impiegato

```
INSERT INTO partecipa(impiegato, competenza, progetto)
VALUES
    (69992, 3247, 13);
```

6.6.11 Assegnazione di un fornitore a un dipartimento

```
INSERT INTO rifornisce(dipartimento, fornitore)
VALUES
    ('V8644XS', 23489419167);
```

6.6.12 Inserimento di un impiegato segretario

A causa dei vincoli, per inserirne un segretario è necessario inserire prima un impiegato con qualifica nulla. Dopodiché sarà possibile l'inserimento di un record riferito a quella matricola nella tabella segretario, corredato di una lingua.

Solo in seguito è possibile aggiornare la qualifica dell'impiegato in segretario.

Si prevede quindi l'utilizzo di una transazione:

```
START TRANSACTION;

INSERT INTO impiegato
    (matricola, nome, cognome, data_di_nascita, data_di_assunzione, dipartimento, qualifica)
VALUES
    (28172, 'Lianna', 'Vitler', '1980/04/25', '2019/05/05', 'C221', null);

INSERT INTO segretario (impiegato, lingua)
VALUES
    (28172, 'Russo');

UPDATE
    impiegato
SET
    qualifica = 'Segretario'
WHERE
    matricola = 28172;

COMMIT;
```

Non è possibile automatizzare oltre il processo perché l'informazione relativa alla lingua deve necessariamente essere inserita a mano.

7 Analisi dei dati

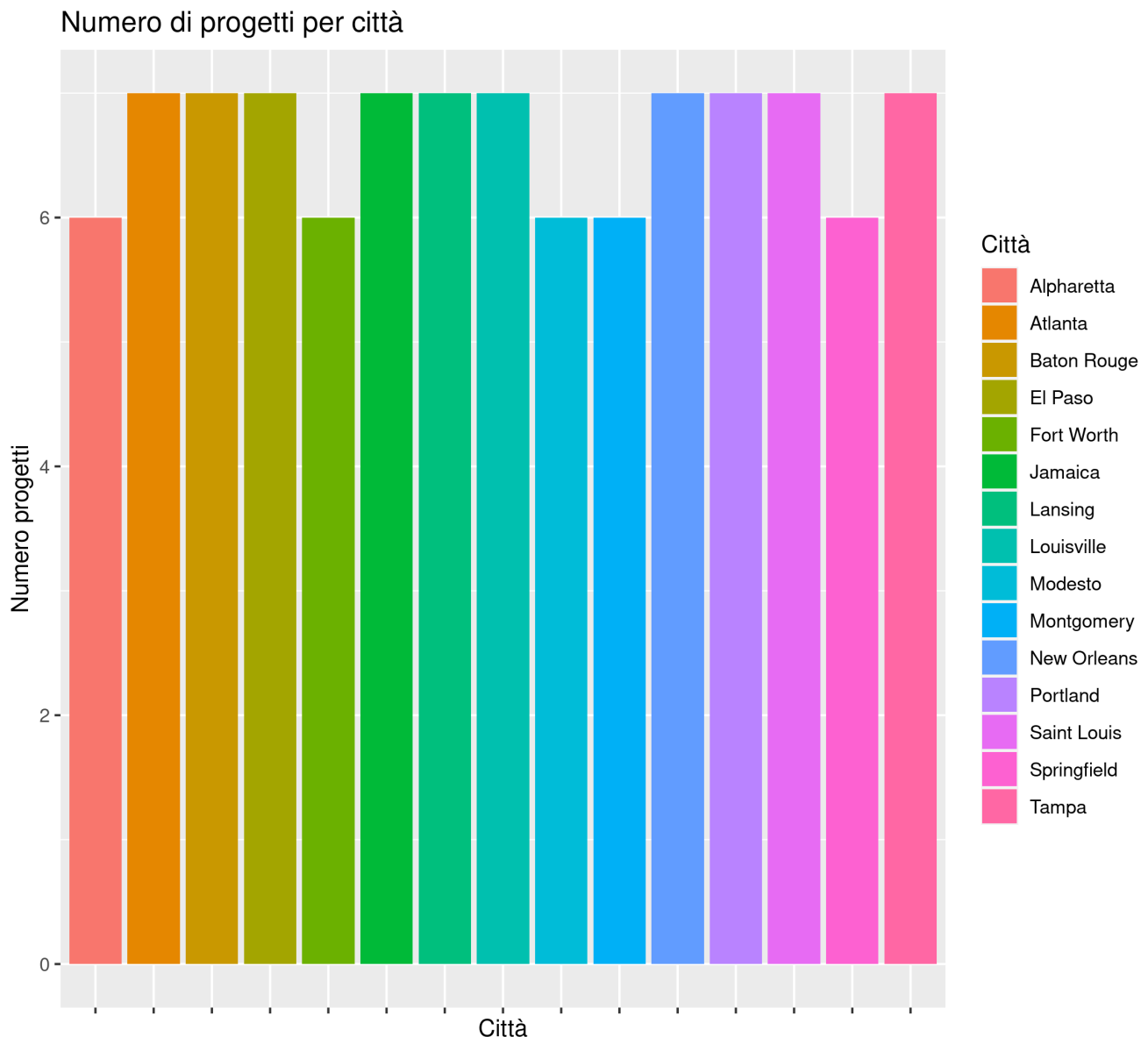
Una volta completata l'implementazione e inseriti i dati di mockup è stato possibile procedere all'analisi dei dati in R. Viene presentato il codice SQL usato per raccogliere i dati seguito da un opportuno grafico prodotto a partire dall'elaborazione degli stessi.

Per questo fine sono state utilizzate le librerie RPostgreSQL, dplyr e ggplot2, rispettivamente per la connessione alla base di dati, per la manipolazione dei risultati delle query e per produrre e salvare le visualizzazioni.

7.1 Numero progetti per città

A seguito dell'interrogazione sono stati prodotti dei bar plot raffiguranti, per ciascuna città, il numero di progetti attivi.

```
SELECT citta, COUNT(*)  
FROM cittaprogetto  
GROUP BY citta  
ORDER BY citta;
```

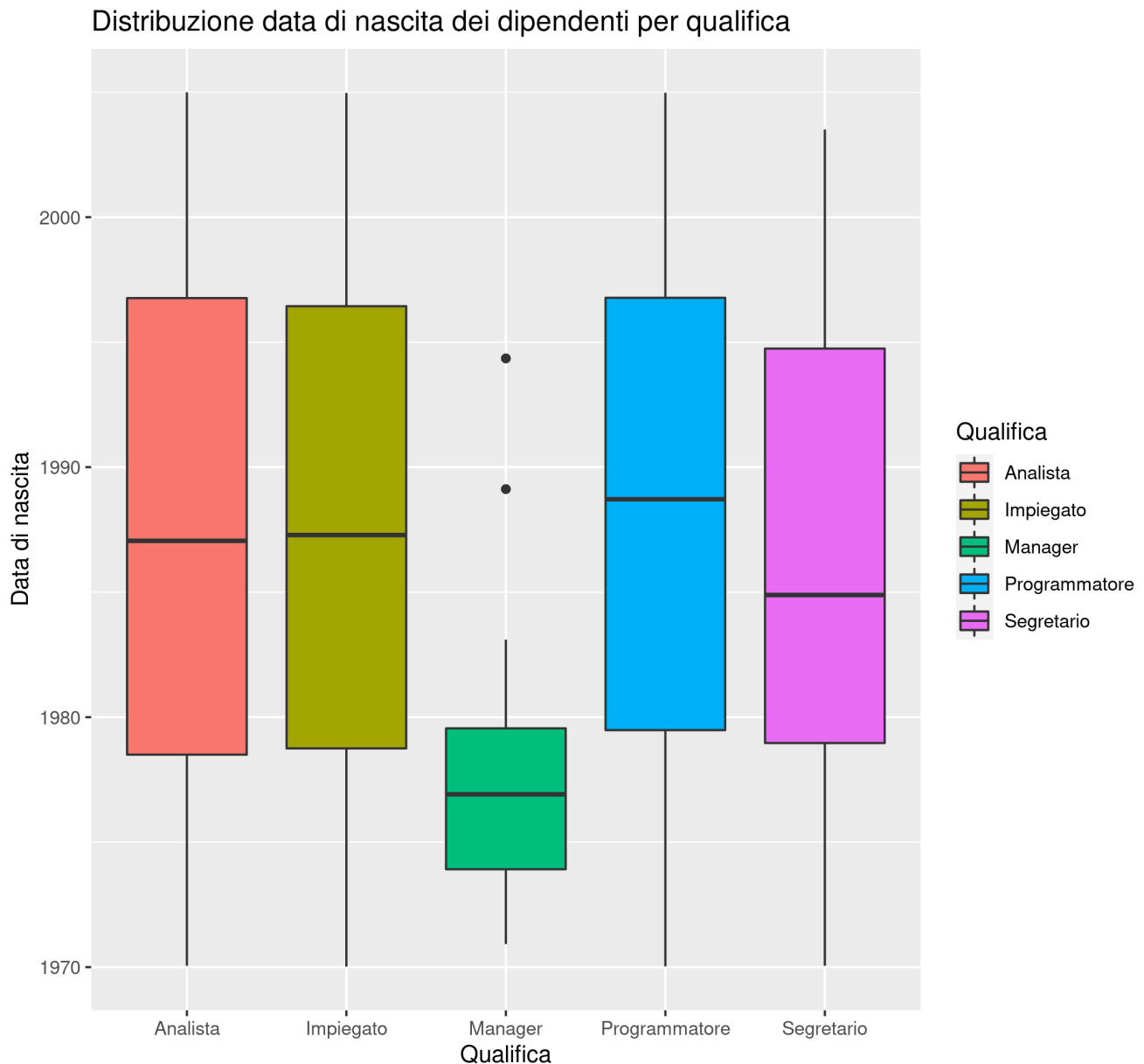


7.2 Distribuzione data di nascita per qualifica

A seguito dell'interrogazione sono stati prodotti dei box plot raffiguranti, per ciascuna qualifica, la distribuzione della data di nascita degli impiegati.

Come si può notare la mediana delle date di nascita degli analisti, impiegati, programmatori e segretari si attesta tra gli anni 1980 e 1990 mentre la mediana delle date di nascita dei manager si attesta tra gli anni 1970 e 1980 con la presenza di 2 outlier.

```
SELECT data_di_nascita, qualifica
FROM impiegato;
```

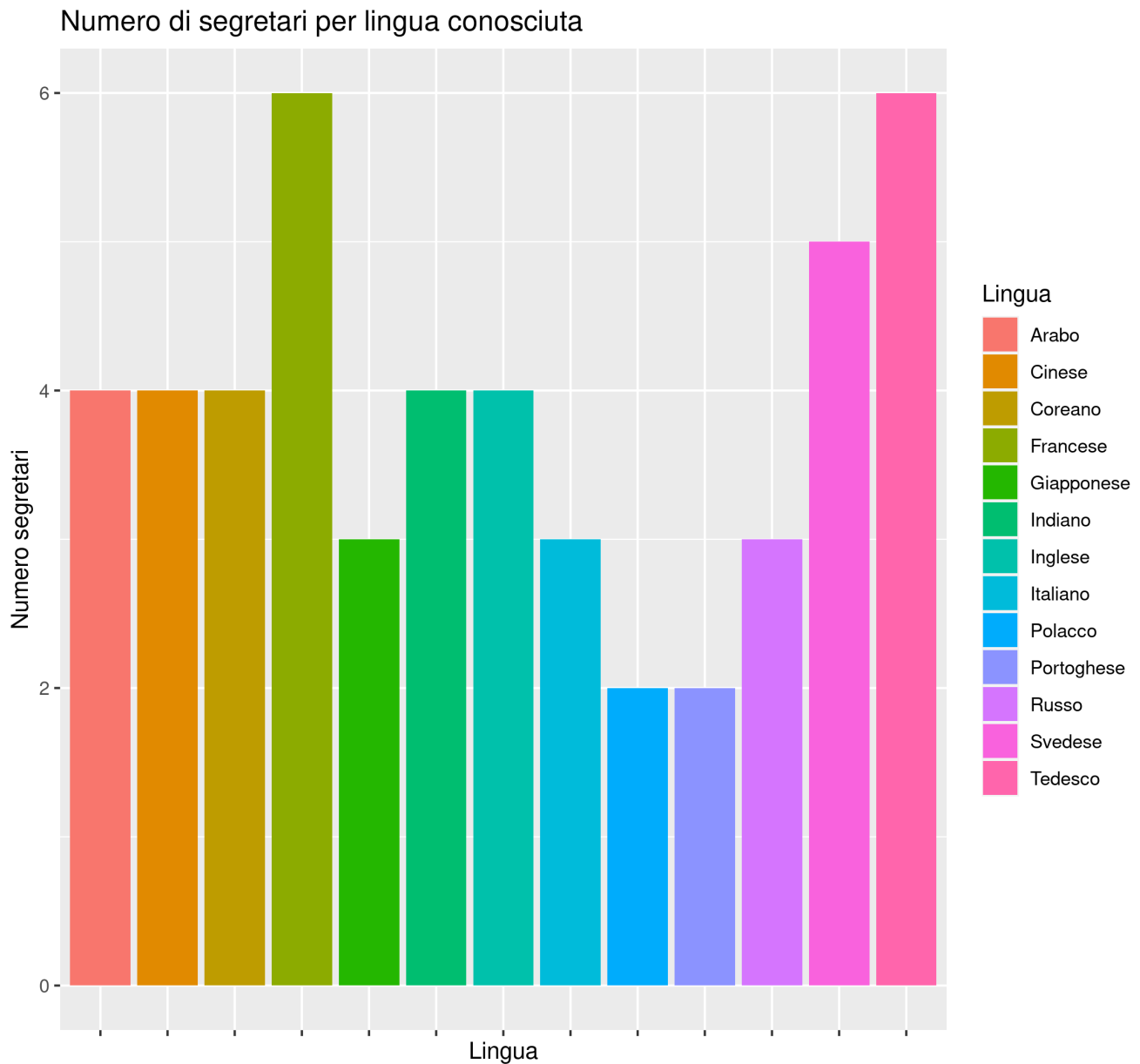


7.3 Numero segretari per lingua

A seguito dell'interrogazione sono stati prodotti dei bar plot raffiguranti, per ciascuna lingua, il numero di segretari che conosce quella determinata lingua.

Come si può notare i segretari dell'azienda rappresentata nella nostra banca dati conoscono un ampio spettro di lingue straniere, con una preminenza di conoscitori della lingua francese, svedese e tedesca.

```
SELECT lingua, count(*)  
FROM segretario  
GROUP BY lingua  
ORDER BY lingua;
```

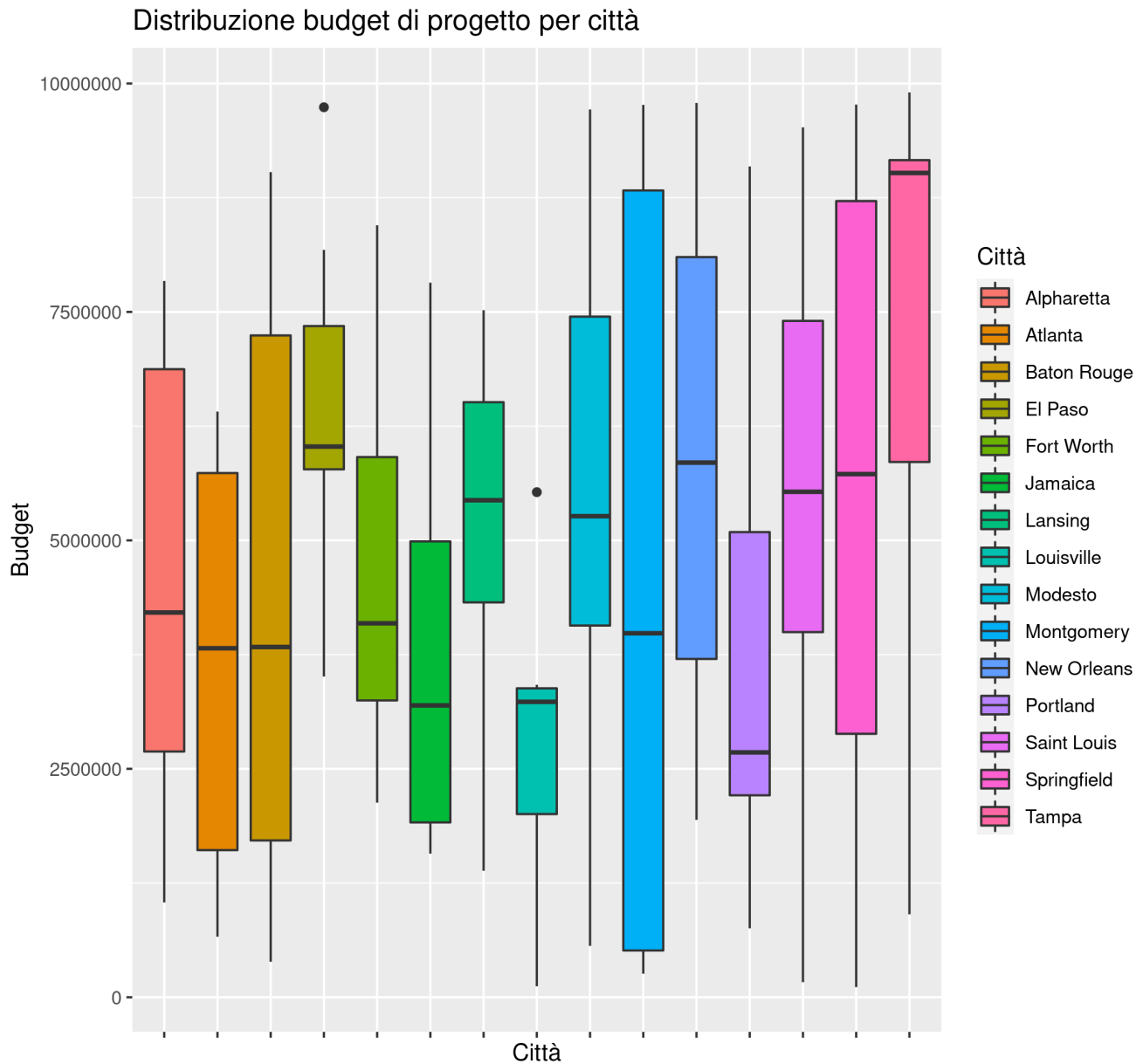


7.4 Distribuzione budget per città

A seguito dell'interrogazione sono stati prodotti dei box plot raffiguranti, per ciascuna città, la distribuzione del budget dei progetti lì svolti.

Come si può notare la mediana dei budget di tutte le città, esclusa Tampa, si attesta in un intervallo compreso tra i 2,5 milioni e tra i 6 milioni.

```
SELECT budget, città  
FROM progetto;
```



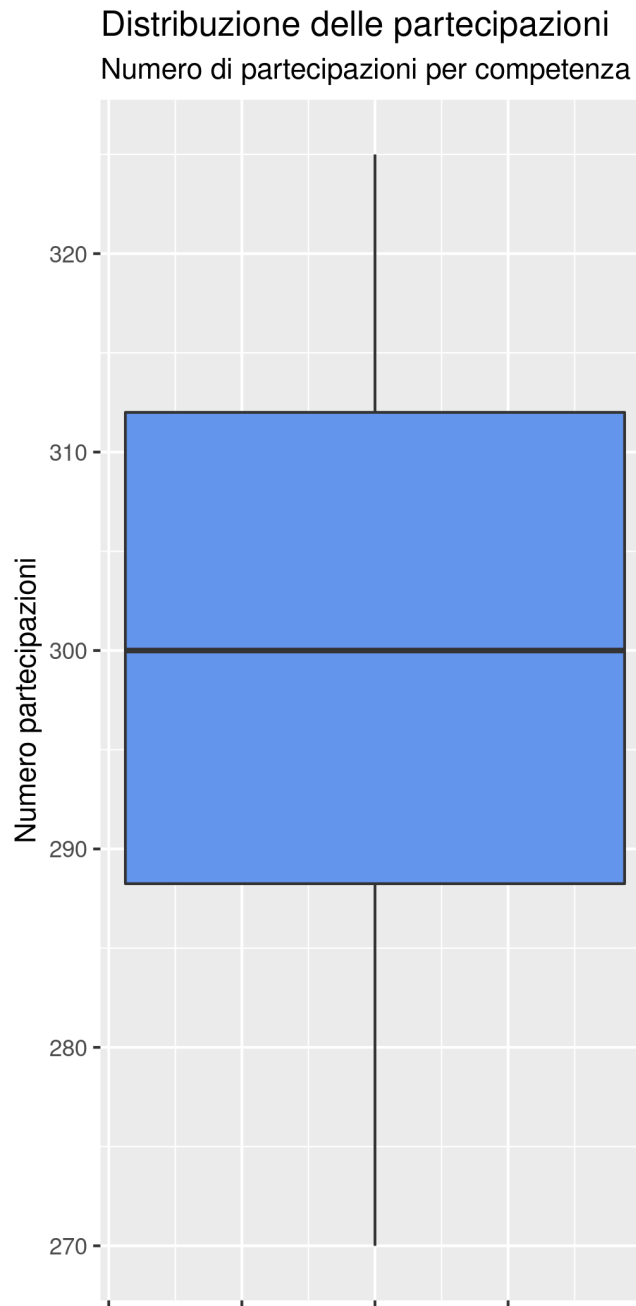
7.5 Distribuzione partecipazioni a progetto con competenze

A seguito dell'interrogazione è stato prodotto un box plot raffigurante la distribuzione delle partecipazioni per competenza.

Come si può notare la mediana di partecipazioni in cui viene applicata una competenza è 300 con una varianza compresa tra 290 e 310.

Non compaiono dei veri e propri outlier ma alcune competenze sono impiegate meno (270) o più spesso (325).

```
SELECT descrizione, COUNT(*) FROM competenza, partecipa
WHERE codice = partecipa.competenza
GROUP BY codice
ORDER BY descrizione;
```

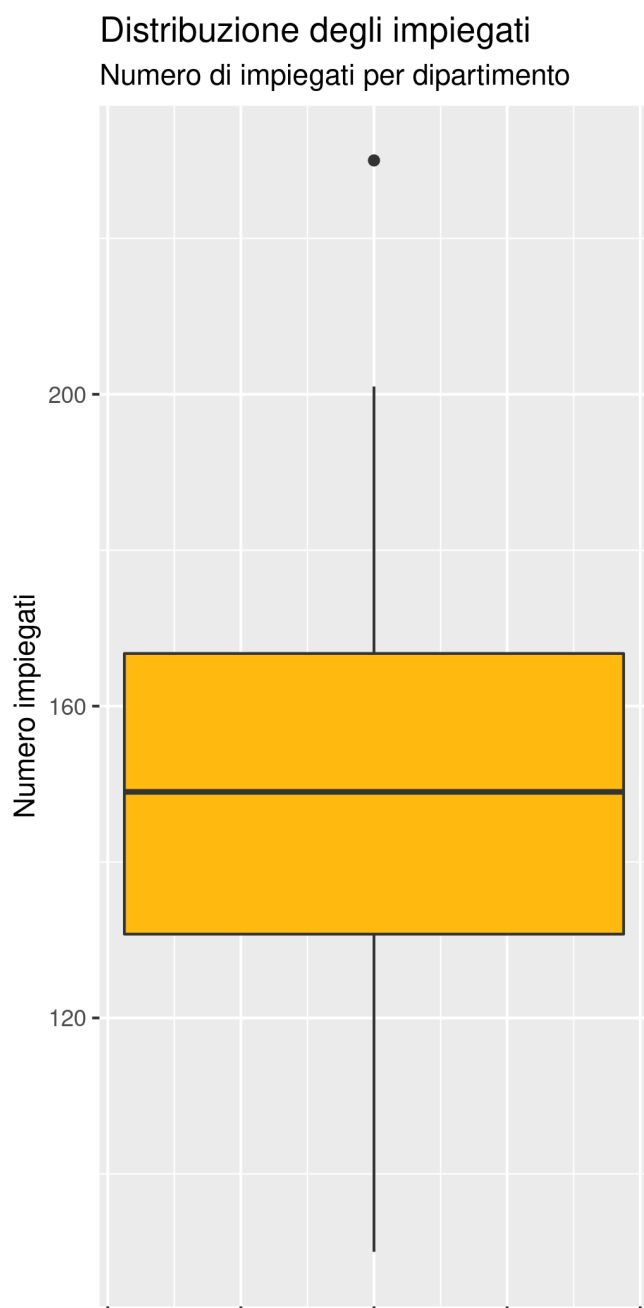


7.6 Distribuzione impiegati per dipartimento

A seguito dell'interrogazione è stato prodotto un box plot raffigurante la distribuzione degli impiegati per dipartimento.

Come si può notare si ha una mediana di circa 150 impiegati per dipartimento con un outlier prossimo ai 230.

```
SELECT dipartimento, COUNT(*)  
FROM impiegato, partecipa  
WHERE matricola = partecipa.impiegato  
GROUP BY dipartimento  
ORDER BY dipartimento
```



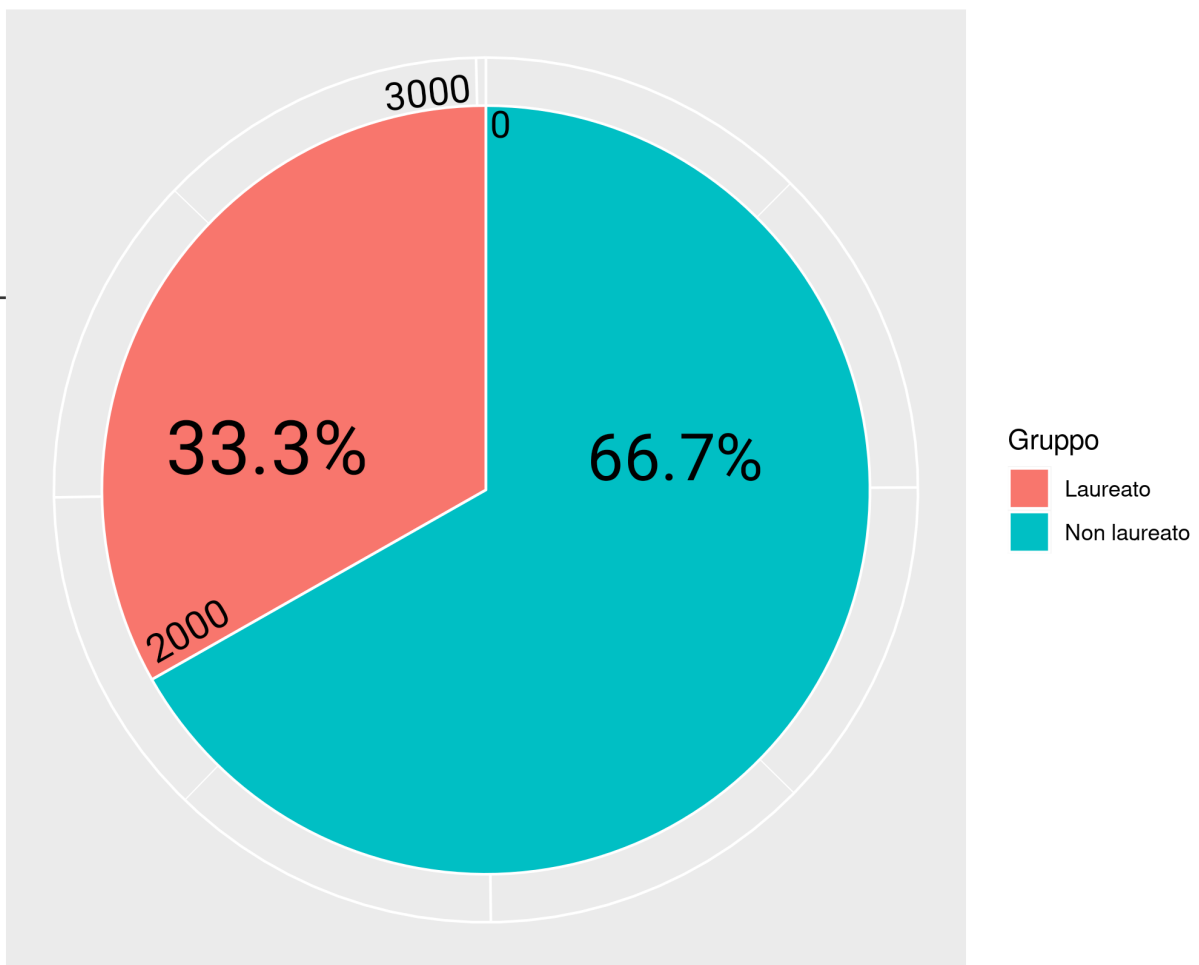
7.7 Percentuale impiegati laureati

A seguito dell'interrogazione è stato prodotto un diagramma a torta raffigurante il numero di impiegati laureati e non.

Come si può notare dal grafico il 66.7% degli impiegati non ha conseguito una laurea, mentre il 33% degli impiegati risulta laureato.

```
SELECT COUNT(*)  
FROM laureato  
UNION  
SELECT COUNT(*)  
FROM impiegato  
WHERE NOT EXISTS(  
    SELECT *  
    FROM laureato  
    WHERE matricola = laureato.impiegato  
);
```

Percentuale di impiegati laureati



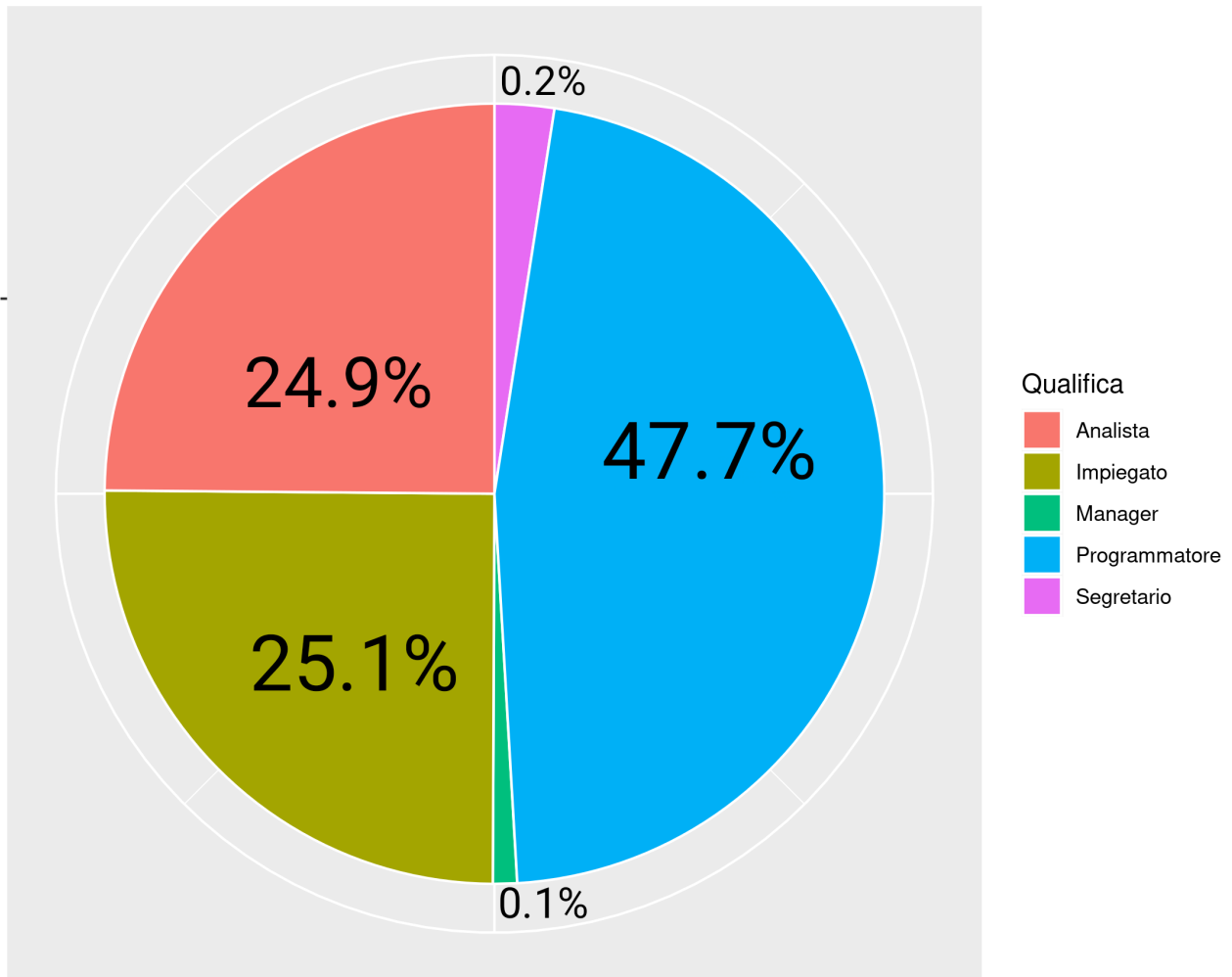
7.8 Percentuale impiegati per qualifica

A seguito dell'interrogazione è stato prodotto un diagramma a torta raffigurante, per ciascuna qualifica, la percentuale di dipendenti impiegati con quella determinata qualifica.

Come si può notare dal grafico quasi il 50% degli impiegati sono dei programmatori, mentre i manager ed i segretari costituiscono una parte marginale dell'organico dell'azienda.

```
SELECT qualifica, COUNT(*)/(SUM(COUNT(*)) OVER()) AS frequenza
FROM impiegato
GROUP BY qualifica;
```

Percentuale di impiegati per qualifica



8 Conclusione

La relazione descrive la progettazione e implementazione di una base di dati relazionale per un'ipotetica azienda.

Il processo è partito dai requisiti forniti poi formalizzati in specifiche.

Particolare attenzione è stata rivolta alla fase di progettazione concettuale.

Per la realizzazione degli schemi è stato usato [Diagrams.net](#).

Si è sfruttata la libreria [Mockaroo](#) per generare dati di mockup il più possibile adatti al dominio e significativi.

Ulteriori operazioni relative ai dati da inserire sono state svolte con espressioni regolari, bash e Python.

Per la collaborazione e i test sulla base di dati si sono sfruttati git, [Overleaf](#), container e virtualizzazione.

L'analisi dei dati in R è stata per quanto possibile automatizzata nella generazione di csv e grafici.

In tutte le fasi del progetto si è fatto uso delle conoscenze acquisite durante il corso.