

tesi di laurea

Analisi e sviluppo di un componente per un ESB open source

Anno Accademico **2010/2011**

relatore

Ch.mo prof. Porfirio Tramontana

correlatore

Ing. Ciro Romano

candidato

Rosario Celotto

Matr. 534/1459



Introduzione

L'attività svolta è stata l'analisi e lo sviluppo in linguaggio Java di un componente per la connessione con un database di tipo DBF

Il componente realizzato è stato concepito come un insieme di classi, nella forma di un progetto Java, utili per l'interfacciamento con Green Vulcano ESB.

L'azienda E@I Software

L'azienda ha una forte esperienza nell'IT Integration. Ha le sedi dei suoi uffici a Roma e Napoli. E' il partner principale di GreenVulcano. Il suo ESB permette un'integrazione semplice e flessibile con i servizi IT di un'azienda

www.eaisoftware.it

Che cos'è un ESB?

"Un Enterprise Service Bus (ESB) è un'architettura software che sfrutta, messaging, middleware, intelligent routing e trasformazione dati. Gli ESB si comportano come dorsali d'integrazione distribuite, attraverso le quali i servizi software e le applicazioni interagiscono."

Gartner



Tecnologie e software utilizzati

Java API: JDBC Java DOM GreenVulcano	Database DBF	XML	Eclipse	DBF Manager	Notepad++
Codice sviluppato	Tabelle Database	File di input e di configura- zione	Ambiente di sviluppo integrato	Creazione di tabelle di tipo DBF	Creazione dei file input .xml
<a href="http://java.sun.com/refere
nce/api/">http://java.sun.com/refere nce/api/			<a href="http://www.eclipse.
org/">http://www.eclipse. org/	www.astersoft.com	<a href="http://notepad-plus-
plus.org/">http://notepad-plus- plus.org/



Analisi dei requisiti

- ❑ Il progetto Java deve permettere l'accesso con diverse funzionalità al database stabilito.
- ❑ Per quanto riguarda il database la tabella deve essere composta da campi stabiliti al momento della creazione, ad esempio nome, cognome, data di nascita e identificativo (tabella anagrafica).
- ❑ Per la gestione delle funzionalità (inserimento, aggiornamento, eliminazione e selezione) bisogna creare diversi file di configurazione esterni al progetto Java.

Nome requisito: selezione

Descrizione funzionalità: selezione di nome, cognome e data per un identificativo (ID) assegnato

Motivazione: la funzionalità permette di visualizzare nome, cognome e data corrispondenti ad un dato ID

Specifica: SQL/DBF

Specifica dei requisiti

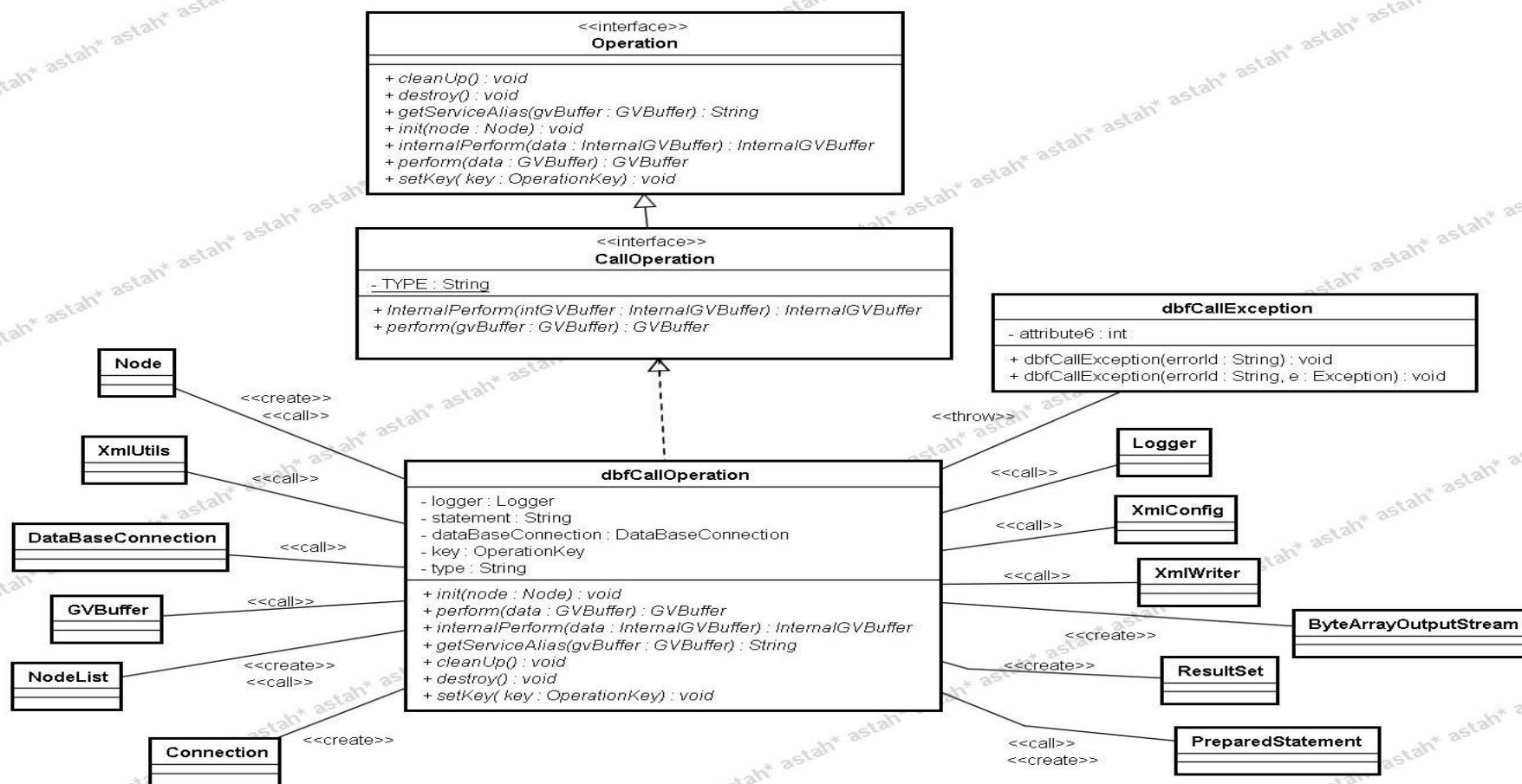
- ❑ Per i requisiti funzionali (funzionalità CRUD) è stato adottato un modello basato sullo standard IEEE 830-1998 per l'SRS (Software Requirements Specification, Documento di Specifica dei Requisiti), quindi per ogni requisito funzionali avremo: introduzione, input, elaborazione, output.
- ❑ Per quanto riguarda input e output è stato necessario scrivere alcuni dettagli implementativi poiché per l'input è necessario utilizzare un particolare tipo di formato di file, per l'output si utilizza lo standard di output di Eclipse però occorre presentarlo in un formato specifico richiesto dall'azienda



Fase di sviluppo 1/2

- ❑ Il progetto che ho realizzato è consistito in un progetto di prova che è stato implementato ai fini del testing, cioè per verificare l'esecuzione corretta delle funzionalità richieste.
- ❑ Il progetto Java, consiste di tre classi Java: DBFCallOperation, DBFCallException e testdb.
- ❑ DBFCallOperation è la classe che contiene i metodi necessari per l'implementazione delle funzionalità richieste, DBFCallException gestisce le eccezioni eventualmente sollevate nella classe precedente e testdb è la classe che contiene il metodo main e dalla quale il programma viene compilato.
- ❑ Non appena il progetto viene compilato lo standard di input chiede l'inserimento di una stringa corrispondente ad una funzionalità: select, update, insert, delete.
- ❑ La classe testdb invoca i metodi init e perform della classe DBFCallOperation e inizializza un oggetto di tipo GVBuffer che serve per memorizzare l'output del progetto

Fase di sviluppo 2/2

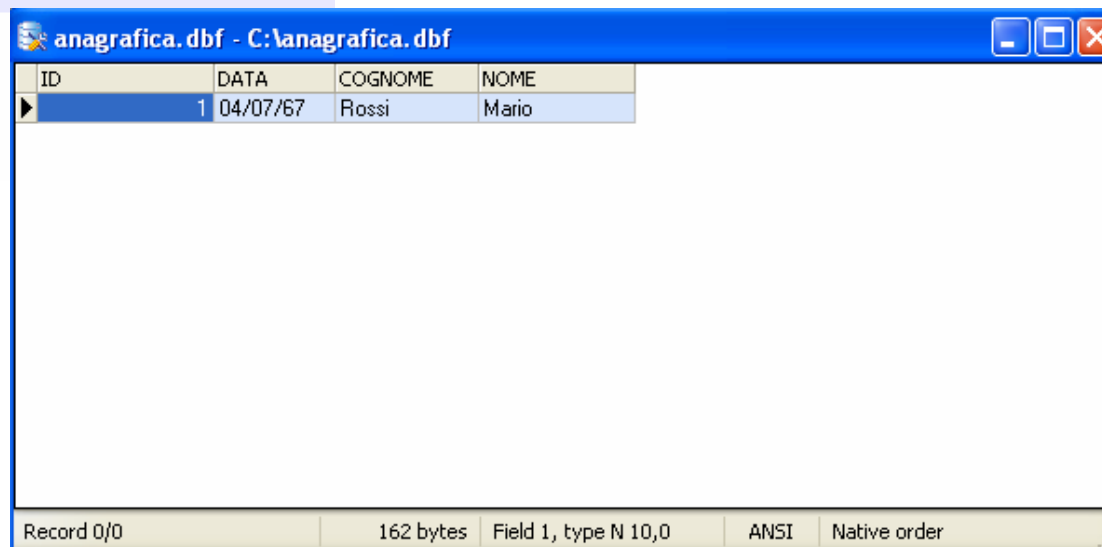


Testing

Post-condizione
successiva
all'inserimento
del record

```
inputIns.xml
1 <RowSet>
2   <row>
3     <col type="string">Mario</col>
4     <col type="string">Rossi</col>
5     <col type="string">1967-07-04</col>
6     <col type="integer">1</col>
7   </row>
8 </RowSet>
```

Input per la funzionalità di
inserimento con file
.xml



anagrafica.dbf - C:\anagrafica.dbf

ID	DATA	COGNOME	NOME
1	04/07/67	Rossi	Mario

Record 0/0 162 bytes Field 1, type N 10,0 ANSI Native order



Conclusioni

- ❑ **Risultati ottenuti:** realizzazione di un progetto di prova in grado di testare le funzionalità descritte nel documento di specifica dei requisiti
- ❑ **Conoscenze acquisite:** approfondimento del linguaggio JAVA e delle API JDBC, JAVA DOM, database di tipo SQL e DBF
- ❑ **Metodologia usata:** implementazione modulare seguendo il metodo bottom-up, ereditarietà, implementazione di interfacce e ridefinizione di metodi preesistenti
- ❑ **Sviluppi futuri:** è possibile modificare il progetto presentando le funzionalità in modalità webservices