



Studio di fattibilità per l'integrazione di un driver C in PoLiDBMS

Paolo Tagliaferri
matr.646468

Una vista introduttiva

- PoLIDBMS, un DBMS particolare
 - Leggero e ottimizzato per l'uso su dispositivi portatili
 - Politiche di memorizzazione "intelligenti"
 - Implementato in *Java* e in linguaggio C





Fasi di lavoro

■ **Obiettivi e vincoli**

- Stato di sviluppo, vincoli di sistema, obiettivi del lavoro

■ **Analisi**

- Analisi architettura esistente
- Definizione del problema
- Analisi dei possibili strumenti risolutivi
- Prototipazione

■ **Proposta di soluzione**

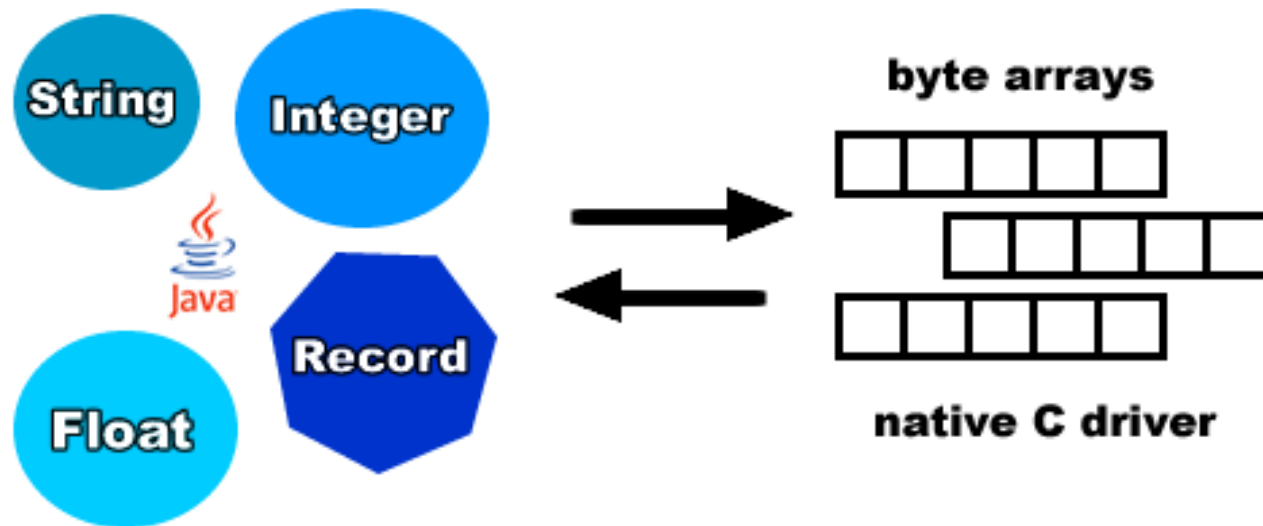
- Implementazione scelta

■ **Collaudo e valutazione**

■ **Conclusioni**

Integrazione Java \leftrightarrow C

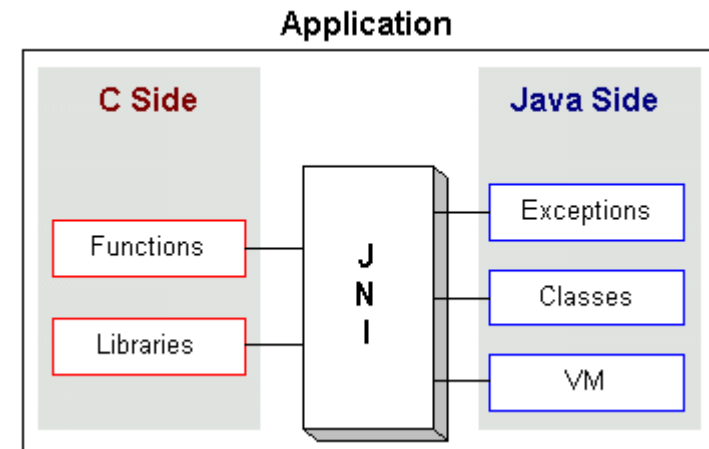
- **PROBLEMA:** il driver che gestisce la memoria flash è implementato in C, il resto dell'applicazione in *Java*.
 - Come rendere il driver accessibile?
 - Come gestire la differente rappresentazione delle informazioni?
 - Come implementare un efficiente protocollo di comunicazione?



Proposta A: Java Native Interface

- **Proposta (A)**: utilizzare il *framework* JNI incluso nella Java Virtual Machine

- ✓ Fornisce un'interfaccia per rendere accessibili da *Java* le funzioni e le librerie native.
- ✓ Viceversa rende utilizzabili in C meccanismi propri di *Java* (eccezioni, creazione oggetti...)



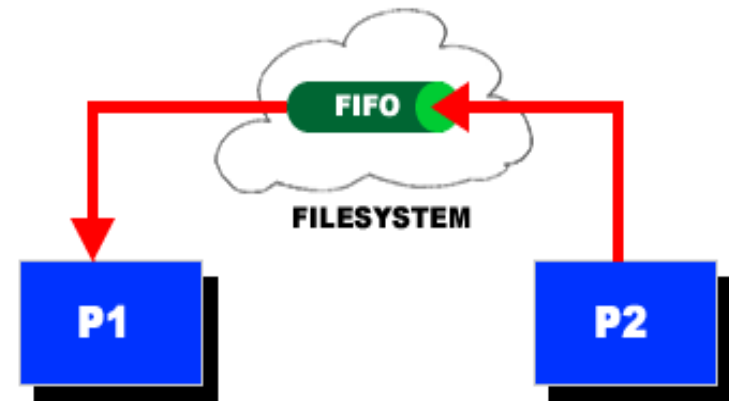
- **Problemi**

- Garantire le prestazioni e la robustezza dell'applicazione
- Circoscrivere le routine basate su JNI
- Mantenere la funzionalità del driver nativo originale

Proposta B: Metodi IPC

- **Proposta (B):** Utilizzare metodi di *Comunicazione Interprocesso* (IPC)

- ✓ Driver e PoLiDBMS sono visti come due processi separati che comunicano tra di loro
- ✓ Ci si è orientati sull'utilizzo delle *named pipes* (FIFO)

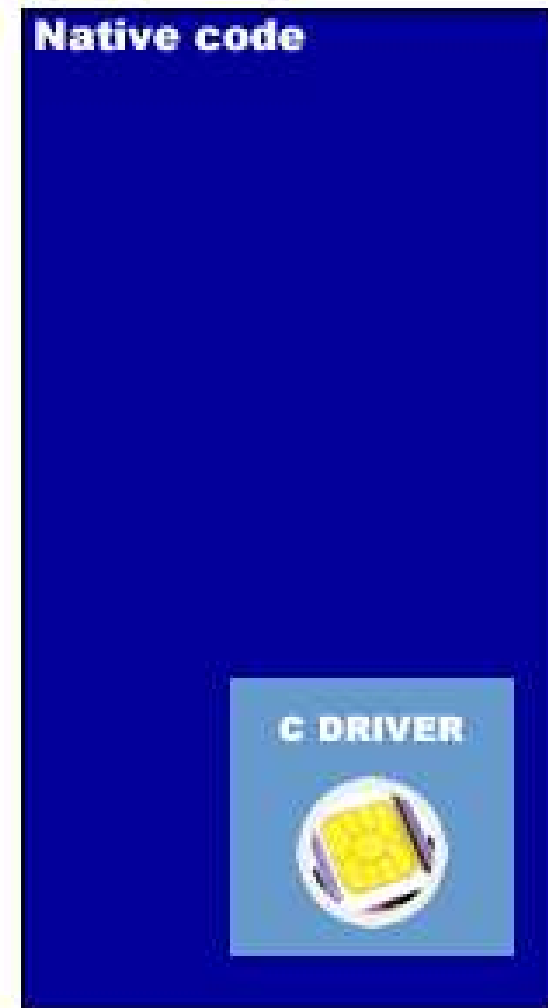


- **Problemi**

- Complessità del protocollo di comunicazione
- Sequenzialità intrinseca del meccanismo

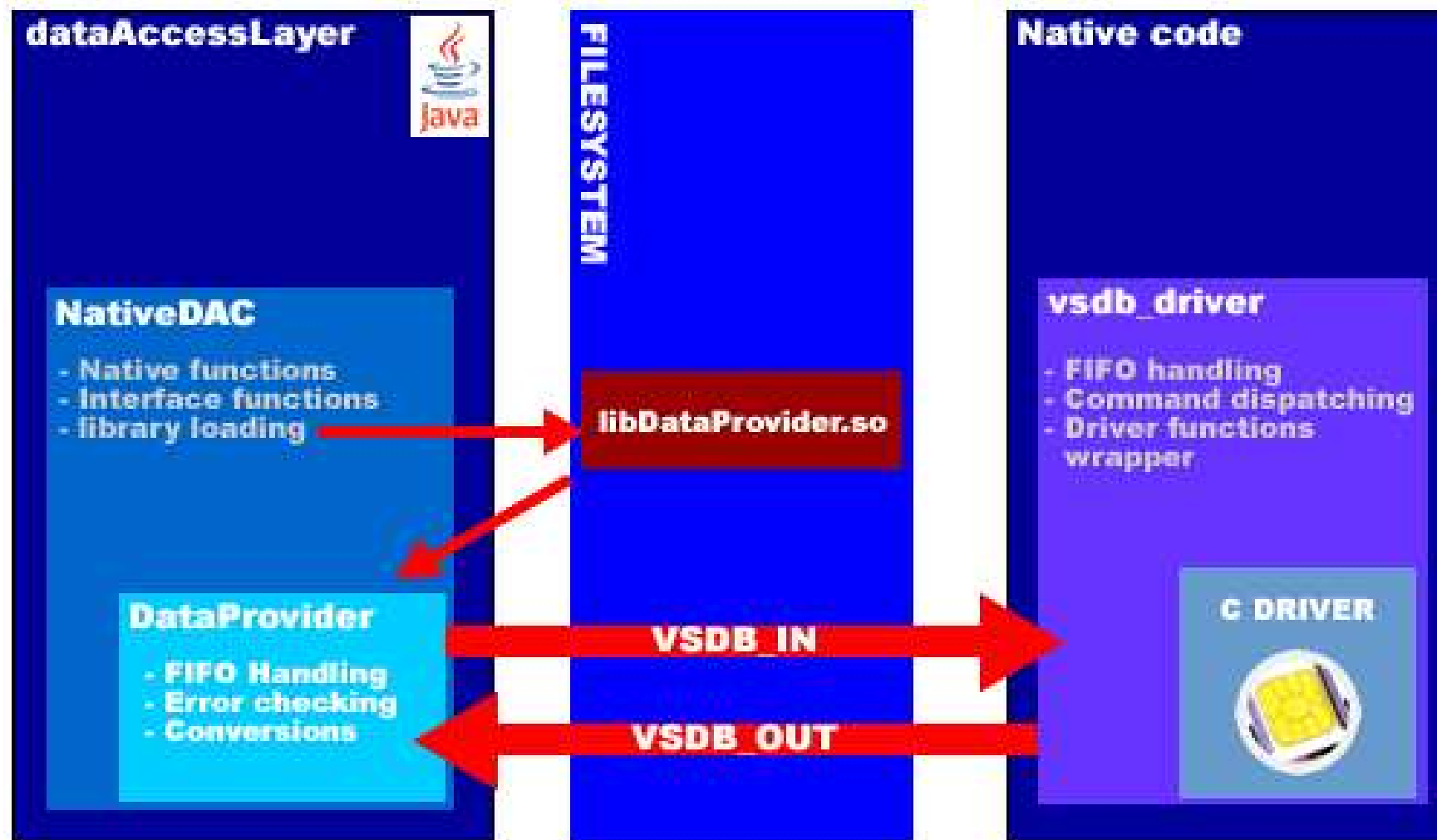
La soluzione adottata

1/2



La soluzione adottata

2/2





Caratteristiche della soluzione scelta

- Un “mix” delle proposte valutate in precedenza
 - Vantaggi dell’uso di JNI:
 - Semplificazione del protocollo di comunicazione tra i due componenti
 - Gestione errori tramite signal handler e lancio di eccezioni direttamente dal codice nativo
 - Facilità di conversione dei dati da oggetti a byte array e viceversa, tramite le funzioni del framework
 - Vantaggi dell’uso delle FIFO:
 - Disaccoppiamento del driver vero e proprio dalla struttura di comunicazione
 - Facilità di scambio dati tra i due componenti sotto forma di array di byte

- Preparazione della piattaforma di test
 - Palmare iPAQ H3900 della *Compaq*
 - Sistema Operativo Unix: *Familiar*
 - Versione **bootstrap**: controllata via shell (seriale o ssh), adatta allo sviluppo di applicazioni perchè occupa poco spazio
 - JVM Blackdown 1.3.1 (*lightweight*)
- Compilazione del sottosistema
 - Classi Java compilate sulla piattaforma di sviluppo e trasferite sul palmare, sfruttando la portabilità del bytecode
 - Codice nativo compilato su cluster pubblico di compilazione per processori ARM (<http://www.handhelds.org>) e trasferito sul palmare

- Utilizzo dell'applicazione di test *NativeDACTest*
 - Simula l'utilizzatore di NativeDAC (PoLiDBMS)
 - Applicazione testuale scritta in *Java*

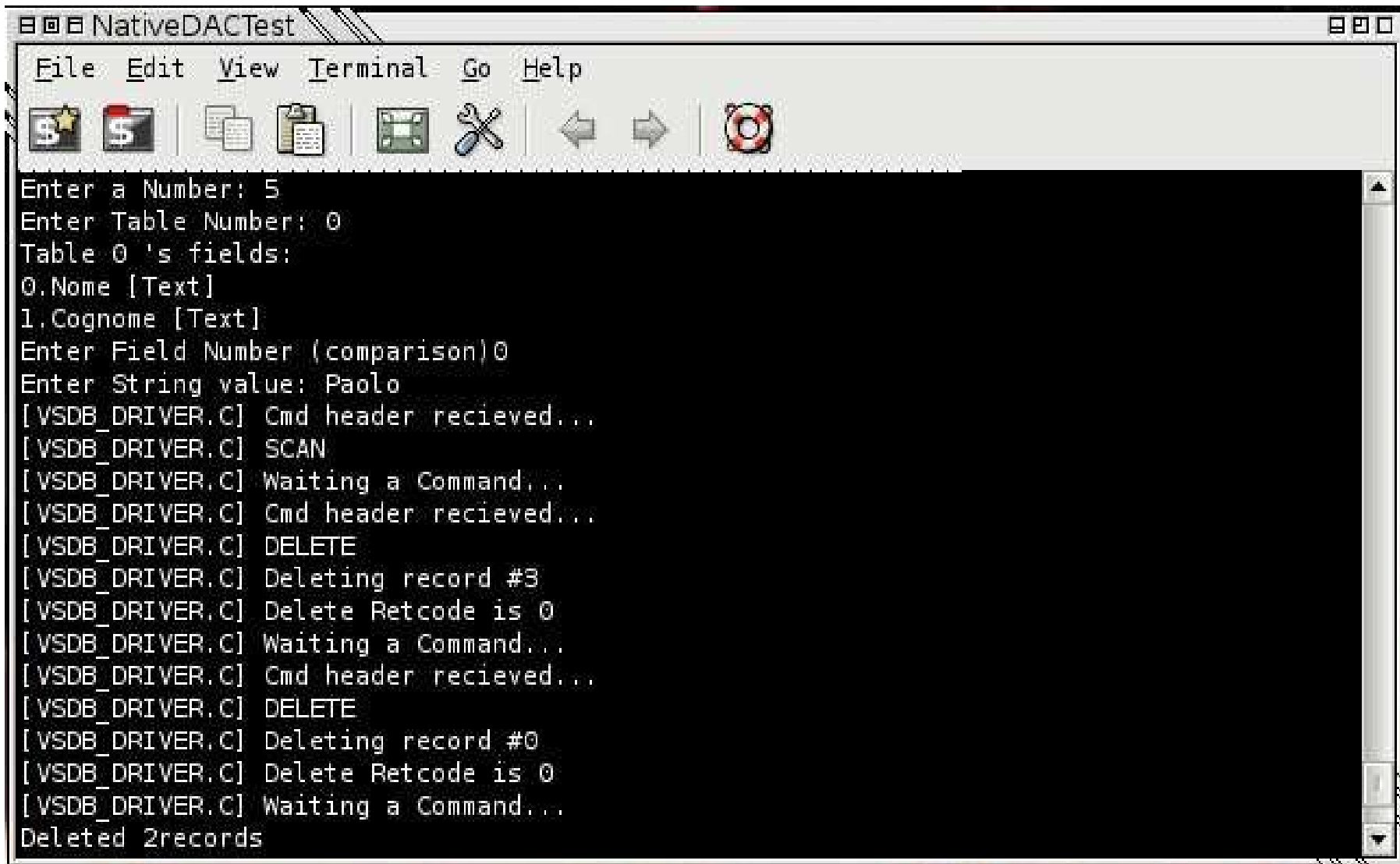
 - Tramite un menù, si scelgono le operazioni disponibili
 - Insert
 - Scan
 - Delete
 - Update
 - Equality search
 - Range search
 - Dispose

```
NativeDACTest
File Edit View Terminal Go Help
[Icons: Dollar sign, Document, Database, Scissors, Left arrow, Right arrow, Lifebuoy]
Enter a Number: 1
Enter Table Number: 1
[VSDB_DRIVER.C] Cmd header recieved...
[VSDB_DRIVER.C] SCAN
[VSDB_DRIVER.C] Waiting a Command...

-----
[RESULTS ON TABLE 1]
.Record 0: [Citt? : Torino] [Abitanti : 123000] [Tasso Inquinamento : 42.12]
.Record 1: [Citt? : Milano] [Abitanti : 90505] [Tasso Inquinamento : 123.23]
.Record 2: [Citt? : Piacenza] [Abitanti : 20040] [Tasso Inquinamento : 424.4]
.Record 3: [Citt? : Borgo Antico] [Abitanti : 901110] [Tasso Inquinamento : 41000.0]
[END RESULTS ON TABLE 1]

-----
Command Menu (Choose a Number):
1. Scan
2. Equality Search
3. Range Search
4. Insert
5. Delete
6. Update
7. Dispose

-----
Enter a Number: 
```



```
NativeDACTest
File Edit View Terminal Go Help
[Icons: Home, Dollar, Documents, Terminal, Scissors, Left Arrow, Right Arrow, Refresh]
Enter a Number: 5
Enter Table Number: 0
Table 0 's fields:
0.Nome [Text]
1.Cognome [Text]
Enter Field Number (comparison)0
Enter String value: Paolo
[VSDB_DRIVER.C] Cmd header recieved...
[VSDB_DRIVER.C] SCAN
[VSDB_DRIVER.C] Waiting a Command...
[VSDB_DRIVER.C] Cmd header recieved...
[VSDB_DRIVER.C] DELETE
[VSDB_DRIVER.C] Deleting record #3
[VSDB_DRIVER.C] Delete Retcode is 0
[VSDB_DRIVER.C] Waiting a Command...
[VSDB_DRIVER.C] Cmd header recieved...
[VSDB_DRIVER.C] DELETE
[VSDB_DRIVER.C] Deleting record #0
[VSDB_DRIVER.C] Delete Retcode is 0
[VSDB_DRIVER.C] Waiting a Command...
Deleted 2 records
```

```
NativeDACTest
File Edit View Terminal Go Help
[VSDB_DRIVER.C] GETTABLENAME
[VSDB_DRIVER.C] Waiting a Command...
[VSDB_DRIVER.C] Cmd header recieved...
[VSDB_DRIVER.C] GETDBNAME
Got the schema!

-----
Command Menu (Choose a Number):
1. Scan
2. Equality Search
3. Range Search
4. Insert
5. Delete
6. Update
7. Dispose

-----
Enter a Number: [VSDB_DRIVER.C] Waiting a Command..
VSDB_Driver 10240 terminated with status f
Driver was terminated with signal 15
SIGCHLD caught but vsdb_driver exit status is not OK!
.Raising SIGTERM!
This is shutdown hook thread! VM is shutting down ... Trying to shutdown vsdb_driver...
Shutting down vsdb_driver.
taglia@vort3xmind dbms $
```

```
kill
File Edit View Terminal Go Help
taglia@vort3xmind taglia $ ps -A | grep vsdb
10240 pts/0    00:00:00 vsdb_driver
taglia@vort3xmind taglia $ kill 10240
taglia@vort3xmind taglia $
```



Conclusioni e sviluppi futuri

- E' stata dimostrata la validità della soluzione proposta
 - Struttura abbastanza flessibile per future espansioni
 - Funzionalità dimostrata sul dispositivo portatile
 - Capacità di reagire agli errori tramite eccezioni

- Il lavoro non è terminato
 - E' necessaria una revisione del driver per forzare il rispetto delle specifiche
 - Possibilità di futuri miglioramenti della struttura
 - Rendere il driver multi-threaded (maggior parallelismo)
 - Migliorare la gestione degli errori (recovery automatico)
 - Rendere "rigoroso" il protocollo di comunicazione (parser)
 - Ampliare il set di dati servito dal database