

Il pattern facade serve a definire un'unica interfaccia d'ingresso ad un sistema complesso.

Per esempio i vari comandi di un sistema operativo agiscono sull'hardware attraverso il sistema operativo stesso che rappresenta l'interfaccia tra due strati e che maschera la difficoltà delle operazioni dello strato a più basso livello più vicino alla macchina hardware.

Da questo esempio si evince un altro vantaggio del pattern facade ossia il disaccoppiamento tra i livelli che permette un'evoluzione indipendente fra i vari strati, infatti nelle varie evoluzioni dei sistemi operativi i comandi ad alto livello rimangono invariati (copy, move, ecc..), mentre quelli a basso livello possono cambiare adattandosi al nuovo hardware.

Un esempio più vicino al mondo java può essere l'interfaccia JDBC. La connessione al db avviene tramite l'interfaccia `java.sql.Connection`, ma l'implementazione è realizzata dai vari vendor del driver (oracle, mysql, ecc.) .

Nella tecnologia J2EE è molto usato il `sessionFacade`.

Questo pattern si realizza con un `session ejb` posto tra il client e business object BO. Il facade espone verso il client i metodi che interagiscono con la BO mascherando tutte le difficoltà di questa interazione.

Marcello Pirazzoli