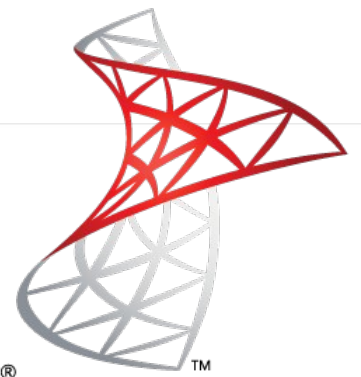




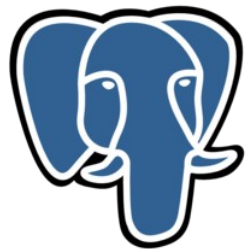
mongoDB



Microsoft®
SQL Server®

ORACLE®

DATA BASE
I DBMS



PostgreSQL

MySQL™



Database

Un **Database** o **base di dati** è una collezione di dati non necessariamente omogenei ma tra loro correlati. Può essere visto come un sistema per rappresentare dei dati e accedere alle informazioni su di essi.

DBMS – Database Management System

- È il software per la gestione di uno o più database.
- Definisce gli utenti e gli amministratori dei database che accedono ai dati a vari livelli.
- Fornisce meccanismi di sicurezza, protezione e controllo dell'integrità dei dati.

Attori

- **DBA, Database Administrator**: gestisce gli accessi, predispone tabelle, indici, viste, ottimizza lo spazio disco, effettua i backup e i restore.
- **Programmatori**: scrivono applicazioni che utilizzano i dati del DB.
- **Utenti finali**: utilizzano le applicazioni che permettono di interrogare il DB.
- **Sistemisti** : verificano il corretto funzionamento dei sistemi hw e sw su cui gira il DBMS.
- **Spesso Sistemista = DBA.**

Il Sysadmin...

- Deve ovviamente conoscere i rudimenti del linguaggio SQL.
- Deve conoscere i DBMS più diffusi... Tutti? Ovviamente sì che ci vuole...
- Deve assicurare i backup dei vari DBMS.
- Deve assicurarne l'integrità.
- Deve suggerirne la ridondanza.
- Deve capire al volo quale DBMS sia migliore per una certa applicazione tenendo sempre conto delle licenze aziendali pre-esistenti.

DBMS

- Tra i più diffusi **DBMS open source** abbiamo:
 - **Mysql (MariaDB)**, **PostgreSQL**, SQLite, Firebird SQL.
- Tra i più diffusi **DBMS commerciali** abbiamo:
 - **Oracle**, IBM DB2, **MS SQL Server**, Sybase, **MS Access**.
- Sono tutti **DBMS relazionali basati su sql (RDBMS)**.
- Vi sono anche **DMBS non relazionali**.
- Vi sono anche **DBMS NO-SQL (MongoDB)**.

MySQL

- È un **Relational Database Management System (RDBMS)** composto da un client a riga di comando e un server.
- Scritto in linguaggio C e C++.
- Disponibile su molti differenti sistemi operativi tra cui AIX, AmigaOS, BSDi, Digital Unix, FreeBSD, HP-UX, GNU/Linux, macOS, NetBSD, Novell NetWare, OpenBSD, OS/2 Warp, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, SGI Irix, Tru64, Windows *.

MySQL

- Linguaggi di programmazione: ODBC, Java, Mono, .NET, PHP, Python, perl, C# ecc..
- **Nasce come piccolo database relazionale adatto a gestire piccole applicazioni e piccoli siti web dinamici.**
- La sua semplicità d'uso ne amplifica la diffusione tanto da essere nativo in CMS importanti come Joomla, WordPress e Drupal. Incluso in piattaforme LAMP e WAMP.
- **Purtroppo, nonostante il continuo sviluppo, resta sempre adatto ad applicazioni non intensive anche se spesso aziende emergenti lo adottano come dbms per applicazioni web.**
- **In particolare pecca in termini di prestazioni per query che riguardano i milioni di record e per la sicurezza.** Un buon tuning aiuta ma non è sufficiente.
- Sistema di gestione: phpmyadmin

PostgreSQL

- È probabilmente **il DMBS open source più avanzato esistente**. Forse il più avanzato in generale. Se la gioca con **Oracle**.
- È un **ORDBMS Object Relational DataBase Management System**:
 - Supporta un modello di database Object Oriented.
 - Permette l'estensione del modello dei dati con tipi di dato e metodi personalizzati.

PostgreSQL

- È scritto in C /C++ e poggia su di una architettura Client/Server.
- Supportato da i maggiori sistemi operativi: Linux, BSD,UNIX, Windows, OS X.
- Utilizzato per applicazioni web, realtime, GIS, sistemi di data warehousing ecc..
- Linguaggi di programmazione: ODBC, Java, Mono, .NET, PHP, Python, perl, C# ecc..

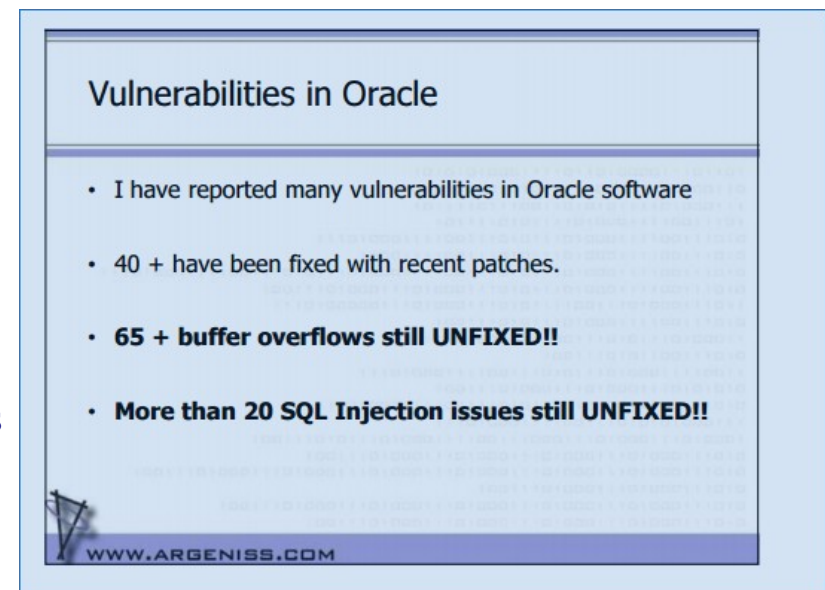
PostgreSQL

- Molto sicuro, i bug fix di sicurezza vengono rilasciati velocemente e costantemente.
- Veloce, molto più di MySQL, soprattutto dalla versione 8.0 ma per grandi moli di dati.
- Sistema di gestione phppgAdmin, pgAdmin3 HeidiSQL.
- **È consigliato per robustezza e sicurezza** rispetto a MySQL.
- Un benchmark vecchio ma interessante:

<http://posulliv.github.io/2012/06/29/mysql-postgres-bench/>

Oracle

- È un **DBMS relazionale esteso con supporto agli oggetti (ORDBMS)**.
- Basato su una **architettura C/S**.
- Dalla versione 10g ha introdotto il concetto di **grid computing**: il DBMS può essere visto come una griglia di risorse, ossia come un insieme di macchine eterogenee interconnesse per formare un ambiente che privilegia scalabilità e condivisione.
- Ora siamo alla versione 21c che migliora performance e aggiunge nuove misteriose funzionalità ma...
- Come nella tradizione di Oracle mantiene i vecchi bug, soprattutto di sicurezza.
- <https://www.blackhat.com/presentations/bh-usa-05/bh-us-05-fayo.pdf>



Oracle

- Disponibile per Windows e Linux.
- Ha un sistema di gestione sviluppato tramite la tecnologia servlet di Java.
- **Oracle** è un commerciale ma ne esiste una versione **express**, incompleta, poco supportata e senza bugfix (solo per lo sviluppo quindi) che potete scaricare:

`http://www.oracle.com/technetwork/
database/database-technologies/
express-edition/downloads/index.html`

MS SQL Server

- È il **database relazionale** di casa **Microsoft**
- Direttamente da msdn.microsoft.com:

SQL Server rappresenta la base della piattaforma dati di Microsoft, offrendo prestazioni mission-critical con tecnologie in memoria e analisi più rapide di qualsiasi tipo di dati, sia in locale che nel cloud.

- Invece wikipedia:

è un DBMS relazionale (Relational Database Management System RDBMS), prodotto da Microsoft. Nelle prime versioni era utilizzato per basi dati medio-piccole, ma a partire dalla versione 2000 è stato utilizzato anche per la gestione di basi dati di grandi dimensioni.

MS SQL Server

- L'ingresso di **Microsoft** nel mondo dei database di fascia "enterprise" risale intorno al 1989 quando cominciò la competizione con Oracle, IBM e Sybase che erano i dominatori del mercato.
- La prima versione fu **SQL Server per OS/2** ed era quasi identica a Sybase SQL Server 4.0 su Unix.
- Fino al 1994 Microsoft SQL Server riportava tre copyright della Sybase come indicazione della sua origine; in seguito Sybase cambiò il nome del suo prodotto in "Adaptive Server Enterprise" per evitare confusione con "**Microsoft SQL Server**".
- **SQL Server 7.0** è stato il primo database server basato su un'interfaccia grafica fabbricata dalla stessa ditta del Database.
- A partire dalla versione 2017, SQL Server è disponibile per Windows, Linux e contenitori Docker.
- La versione attuale è **MS SQL Server 2019**.

MS SQL Server

- Si interfaccia con i maggiori linguaggi di programmazione MS ma vi sono anche driver per Java, C, PHP ecc...
- Esiste una versione **COMPACT** per piccole applicazioni e sviluppo (ad esempio **WID Windows Internal Database**).
- **MS SQL Server Management Studio**: sistema di gestione del DBMS (anche backup).
- Ovviamente commerciale, nè esiste una versione **express** utilizzabile gratuitamente da studenti, sviluppatori ecc

<https://www.microsoft.com/en-us/download/details.aspx?id=52679>

Cosa scegliere?

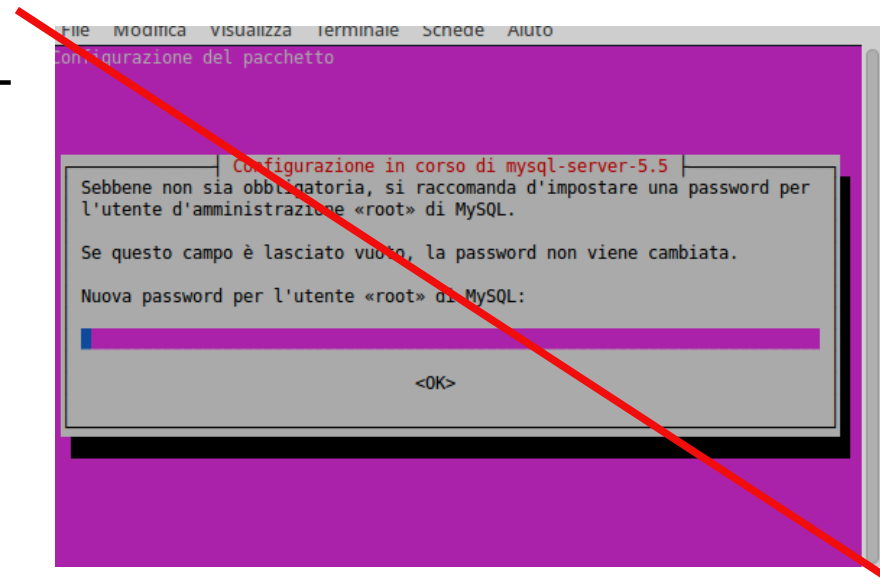
- **MySQL**: il vostro blog, sito web, piccolo forum, applicazioni di medie dimensioni.
- **PostgreSQL**: se avete bisogno di un database relazionale serio ma non avete il budget per permettervelo ;)...
- **SQL Server**: se siete in ambiente windows e avete applicazioni sviluppate conto terzi che richiedono esplicitamente MS SQL server.
- **Oracle**: per aziende di grosse dimensioni che si poggiano su gestionali (SAP), per applicazioni pesanti.
 - Ca Foscari lo utilizza per tutte le attività: web, amministrazione, esse3,...
- A casa vostra usate quello che volete ;)

Esempio: mysql su ubuntu

- Per installare mysql server su Ubuntu:

```
# apt install mysql-server mysql-client
```

- ~~Vi chiederà di inserire una password di root (dba) del database.~~



- Utilizziamo una comoda utility di Ubuntu/Debian per effettuare una configurazione iniziale di MySQL:

```
# mysql_secure_installation
```

Esempio: mysql su ubuntu

- La configurazione e il tuning di **MySql** avviene editando il file `/etc/mysql/conf.d/mysql.cnf`.
- Qui potete, ad esempio specificare quali reti possono provare ad accedere a **MySql**:

```
bind-address = 127.0.0.1 ; ammette solo  
localhost
```

- Una volta cambiati i parametri di `mysql.cnf` dovete riavviare il servizio

```
# service mysql restart  
# systemctl restart mysql
```

Esempio: mysql su ubuntu

- Per la gestione e l'utilizzo è disponibile una console da riga di comando:

```
# mysql -u root -p
```

```
# mysql -u root -p -h 192.168.1.1 ; se su un altro server
```

- Una volta avviata la console potete ad esempio vedere quali database siano installati:

```
# mysql> show databases;
```

- Per creare un db è sufficiente usare la clausola create database:

```
#mysql> create database books_db;
```

- E poi dare l'accesso al gestore del db:

```
# mysql> GRANT ALL privileges ON books_db.* TO  
'librarian'@'127.0.0.1' IDENTIFIED BY  
'unapasswordacasomadecente'
```

```
# mysql> FLUSH PRIVILEGES;
```

- Ora sul database appena creato root e librarian possono fare di tutto, creare tabelle, cancellarle, effettuare query e anche DROP DATABASE!

Esempio: mysql su ubuntu

- Un tool che aiuta nella gestione di **MySql** è **phpmyadmin**.
- È una **interfaccia realizzata interamente in php** che vi permette di effettuare operazioni anche complesse con il database.
- Richiede un web server per funzionare e va protetta da accessi indesiderati.
- Per installarla:

```
# apt install phpmyadmin
```

Progetto

- Realizzare un confronto tra:
 - PostgreSQL
 - Oracle
 - MS SQL Server
- Confronto: progettare una piccola base di dati con un certo numero di query e implementarla su almeno due dei tre dbms proposti.

Avete Abbiamo un Server LAMP yeah!

