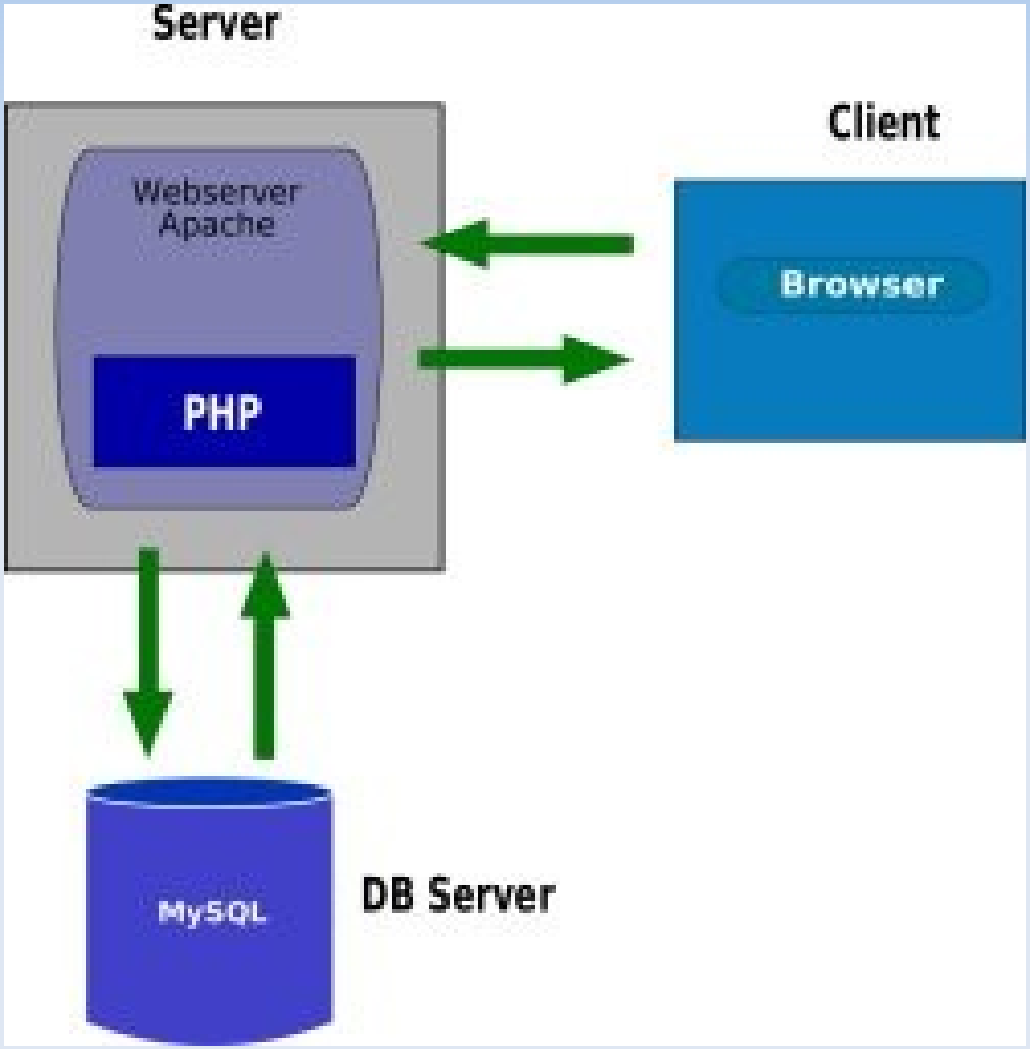
Sviluppare nel Web

La programmazione Web oggigiorno viene sviluppata su livelli distinti ma complementari:

- Server side:
  - Server Web
  - Pagine HTML
  - Programmazione PHP,Perl,Python,Ruby,Bash
  - Basi di Dati relazionali
- Client Side:
  - Browser Web (web client)
  - Javascript, Ajax (codice eseguito dal client)
  - Flash, Java plugins locali, Adobe AIR

Il web dinamico



Il client effettua una chiamata GET su un contenuto (URL).

Il WebServer esegue la funzione (PHP) associata.

Questa interroga la BaseDati per recuperare i contenuti e renderizzarli in HTML

Perchè usare i RDBMS

- Metodi di Interrogazione: Architettura Client-Server
- Sicurezza, gestione degli accessi, letture scritture concorrenti, monitor delle prestazioni
- Strumenti Odierni:
  - Server: appunto MySQL
  - Clients:
    - Server-side: Php-MyAdmin
    - Puri: mysql da riga di comando, SQLyog, Workbench, cursori in PHP/PERL/PYTHON

Terminologia

DataBase: traducibile in italiano come "base di dati" non è un altro che un insieme di dati logicamente correlati fra loro.

I Data Base Management System (DBMS) sono software in grado di gestire:

- capacità di gestire grandi quantità di dati
- condivisione dei dati fra più utenti e applicazioni
- utilizzo di sistemi di protezione e autorizzazione per l'accesso ai dati stessi

Caratteristiche peculiari MySQL

- OpenSource con ampia comunità di sviluppo/supporto
- Velocità di risposta, MySQL con la tecnologia MyISAM diviene l' RDBMS più veloce al mondo
- Ampiamente utilizzato nel WEB
- Distribuito e Documentato in maniera da facilitarne la conoscibilità e l'accesso
- Con la versione 5 (stabile dall'ottobre 2005) MySQL è pronto per le soluzioni Enterprise.

Reperire MySQL

Se Fornito dal provider:

1. Sfruttare PhpMyAdmin Server-side
2. Ottenere la porta di connessione per interfacciare via internet il nostro Client MySQL

Altrimenti Downloads per conto nostro:

http://dev.mysql.com/downloads/

Fondamenti: Tabelle e SQL

i dati vengono presentati in forma tabulare, cioè come un insieme di tabelle ciascuna composta da righe e colonne.

ogni tabella contiene i dati relativi ad una entità.

le colonne della tabella rappresentano i campi, ovvero le proprietà o attributi dell'entità.

le righe della tabella esprimono le ricorrenze dell'entità.

Insieme al modello relazionale è stato introdotto il linguaggio SQL (Structured Query Language).

Documentazione e References

http://database.html.it/guide/leggi/87/guida-mysql/

http://dev.mysql.com/doc/refman/5.0/en/

http://a2.pluto.it/a2896.htm

http://www.w3schools.com/sql/default.asp

http://it.wikipedia.org/wiki/MySQL

http://www.morpheusweb.it/html/manuali/sql.asp

Inserimenti di dati

```
insert into Iscritti  (nome, cognome, email,
   Corsi_id_corsi) VALUES ('giovanni', 'presa',
   'ciao@yaho.it', 1)
```

Le Query SQL indispensabili

select * from Iscritti where nome = 'giovanni'
order by cognome;

select * from Lezioni where obiettivi LIKE
"comprensione%"

Gli operatori LOGICI:

> help logical operators

select * from Iscritti where nome = 'giovanni' and
cognome like '%sa%';

select * from Iscritti where nome = 'giovanni' or
nome='giuseppe'

Contare, raggruppare, filtrare

L'operatore GROUP BY raggruppa i risultati di una SELECT in base al campo specificato dopo il BY. L'operatore va usato in congiunzione con una funzione statistica (COUNT, SUM ...).

```
select nome, congome, count(*) from Iscritti where nome LIKE 'giovanni%' group by nome;
```

L'operatore HAVING filtra ulteriormente i risultati visualizzati con GROUP BY.

```
select nome,cognome,  count(*) as total  from Iscritti where nome like 'giovanni%' group by cognome having total > 1;
```

Cancellare e aggiornare i dati

```
delete  from Iscritti where nome is 'NULL';

delete  from Iscritti where nome='fabio';

UPDATE Iscritti  SET nome='franco',
   cognome='pagliuso' WHERE nome='giuseppe';

UPDATE Iscritti SET nome = replace(nome,
   "franco", "giuseppe");
```

Un pò di SQL – Selezionare i Dati

SELECT * FROM nome_tabella

SELECT COUNT(*) FROM nome_tabella

SELECT COUNT(*), MAX(colonna1) FROM nome_tabella WHERE colonna2 = valore

SELECT categoria, max(stipendio) FROM dipendenti GROUP BY categoria |*(HAVING)*

SELECT column_name(s) FROM table_name

ORDER BY column_name(s) ASC|DESC

SELECT * FROM Persons WHERE LastName='Svendson' AND (FirstName='Tove' OR FirstName='Ola')

Select DISTINCT

Il comando SELECT DISTINCT di SQL serve ad estrarre una sola volta ogni diversa occorenza riscontrata in un dato campo del DB. Spesso viene usato per creare le voci di menu in modo tale che, se una stessa categoria viene trovata più di una volta all'interno di una colonna, viene stampata a video una volta soltanto.

DISTINCT: select distinct (nome) from Iscritti ORDER BY cognome ASC;

Subquery :Query dentro altre

```
select * from Iscritti where id_iscritti in (select
   count(*) from Iscritti)
```

Varie Permessi

```
GRANT SELECT ON acquisti.* TO
  luca@localhost IDENTIFIED BY 'password'
  WITH GRANT OPTION

GRANT ALL ON acquisti.ordini TO
  paolo@localhost

REVOKE SELECT on acquisti.* FROM
  luca@localhost

REVOKE ALL PRIVILEGES, GRANT OPTION
  FROM paolo@localhost

UPDATE mysql.user SET Password =
  PASSWORD('pwd') WHERE User = 'root';

FLUSH PRIVILEGES;
```

Le JOINs: usare diverse tabelle

L'INNER JOIN restituisce le righe delle tabelle se c'è un legame, altrimenti non le mostra.

SELECT campi FROM prima_tabella INNER JOIN seconda_tabella ON prima_tabella.chiave_primaria = seconda_tabella.chiave_esterna

select * from Iscritti as I, Corsi as C where I.Corsi_id_corsi=C.id_corsi;

select *, replace('Programma_id_programma', 1, 'lamp') from Iscritti as I, Corsi as C where I.Corsi_id_corsi=C.id_corsi;

Altre Join

- la LEFT OUTER JOIN restituisce tutte le righe della prima tabella (nell'esempio Impiegati), anche se non ci sono corrispondenze nella seconda tabella (nell'esempio Ordini).

SELECT campi FROM prima_tabella LEFT OUTER JOIN seconda_tabella ON prima_tabella.chiave_primaria = seconda_tabella.chiave_esterna

- Una RIGTH OUTER JOIN restituisce tutte le righe della seconda tabella, anche se non ci sono legami con la prima

SELECT campi FROM prima_tabella RIGTH OUTER JOIN seconda_tabella ON prima_tabella.chiave_primaria = seconda_tabella.chiave_esterna