

# La comunicazione sul Web



Andrea Giacomini

Istituto Tecnico Industriale Statale "Luigi Negrelli"

Indirizzo Informatica Industriale

*Tesina di maturità*

25 Giugno 2009

## Abstract

Sebbene il Web sia utilizzato da molti di noi, non tutti ne comprendono le complesse dinamiche di funzionamento. Questa tesina descrive le differenze che si sono venute a creare fra *Web 1.0* e *Web 2.0* e come web application, che prendono il nome di *Content Management System*, abbiano contribuito in questo cambiamento, rendendo la gestione dei contenuti web alla portata di tutti gli utenti. Oltre alla facile gestione dei contenuti possiamo avvalerci di una facile e flessibile archiviazione dei dati, attraverso i *database (basi di dati)*. Questi due concetti stanno alla base della tesina e sono spiegati nella loro interezza nei capitoli successivi.

## Acknowledgements

Desidero innanzitutto ringraziare il prof. Luigino Calvi per avermi seguito durante la stesura di questa tesina. Inoltre ringrazio il Dott.Ing. Stefano Angaran per l'aiuto datomi nell'apprendimento del linguaggio tipografico  $\text{\LaTeX}$  utilizzato per realizzare questo documento.

---

# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
<b>2</b>	<b>Storia della comunicazione</b>	<b>3</b>
2.1	Nascita della scrittura . . . . .	4
2.2	La stampa: dalla bibbia di Gutenberg ai quotidiani . . . . .	5
2.3	I cambiamenti socio-economici di fine 1800 . . . . .	6
2.4	Dai telegrafi al web . . . . .	8
<b>3</b>	<b>Web 2.0</b>	<b>13</b>
3.1	Web: cenni storici . . . . .	13
3.2	Web 1.0 . . . . .	13
3.3	Web 2.0 . . . . .	15
3.4	Gerarchia del Web 2.0 . . . . .	18
3.5	Mashup . . . . .	19
<b>4</b>	<b>I CMS</b>	<b>21</b>
4.1	I Web Content Management System . . . . .	22
4.2	Separazione dei dati . . . . .	24
4.3	Categorie di utenti . . . . .	26
4.4	CMS e Wiki a confronto . . . . .	27
4.5	CMS: un po' di storia . . . . .	29
<b>5</b>	<b>Blog &amp; WordPress</b>	<b>31</b>
5.1	WordPress: un po' di storia . . . . .	33
5.2	Funzionalità di WordPress . . . . .	34
5.3	Trackback e Pingback . . . . .	37

## INDICE

---

5.4	SQL Injection . . . . .	38
5.4.1	Classe WPDB di WordPress . . . . .	41
<b>6</b>	<b>Struttura del database di Wordpress</b>	<b>45</b>
6.1	wp_options . . . . .	47
6.2	wp_posts . . . . .	47
6.3	wp_comments . . . . .	49
6.4	wp_users . . . . .	50
6.5	wp_usermeta . . . . .	51
6.6	wp_postmeta . . . . .	51
6.7	wp_terms, wp_term_relationships e wp_term_taxonomy . . . . .	53
6.8	wp_links . . . . .	55
<b>7</b>	<b>Conclusioni</b>	<b>57</b>
	<b>Bibliografia</b>	<b>63</b>

# 1

## Introduzione

La tesina illustrata di seguito tratterà in modo tecnico tematiche legate al mondo dell'informatica. Saranno esaminate le differenze tra la vecchia concezione del Web e quella nuova, identificata con il termine "Web 2.0". Per dimostrare le potenzialità dei database, impiegati frequentemente nel World Wide Web, verrà studiato il funzionamento di un piccolo CMS (*Content Management System*): Wordpress. È doveroso ricordare che sulla rete, nella maggior parte dei casi, le basi di dati che sono necessarie per contenere tutte le informazioni che compongono un sito web possiedono una struttura molto più complessa e strutturata di quella Wordpress.

Nello spirito della tesina verranno analizzate tutte le più importanti tappe nel campo delle comunicazioni: dal principio fino all'introduzione del Web.

Le motivazioni che mi hanno spinto a realizzare questa tesina sono svariate e nello specifico sono:

- interesse nel campo della teoria dei database e relative applicazioni web;
- interesse nel campo della comunicazione tramite Internet;
- utilizzo di piattaforme (CMS) come Wordpress per la diffusione di idee e notizie;
- definire il passaggio dal Web 1.0 al Web 2.0;
- originalità del contenuto.

## 1. INTRODUZIONE

---



## 2

# Storia della comunicazione

*Scriviamo lettere a mano, leggiamo un giornale, telefoniamo a un conoscente e infine inviamo una email. In pochi gesti ripercorriamo 6 mila anni di storia della comunicazione.*

In questo paragrafo verranno trattati brevi cenni sulla storia della comunicazione, un'analisi dettagliata non è possibile data la vastità del tema trattato.

La storia della comunicazione è un'ampia disciplina che studia le tecniche e i modi utilizzati dagli uomini per comunicare con altri suoi simili nell'arco della storia. Questa disciplina studia in effetti la comunicazione dal suo primordiale stato: i graffiti sulle pareti delle caverne a quello finale: l'avvento del Web (discusso nei paragrafi successivi).

È necessario innanzitutto spiegare il significato del termine “comunicare”. La parola comunicare deriva dal latino “*communico*” (che si scinde in *cum* = con e *munire* = legare, costruire) ossia rendere partecipi e far conoscere.

Il destinatario della comunicazione non deve essere necessariamente un individuo senziente, l'emittente può limitarsi semplicemente a tramandare le sue idee e il suo pensiero nel tempo.

In una comunicazione si identificano sempre 5 elementi:

- emittente;
- destinatario;
- canale;
- codice;

## 2. STORIA DELLA COMUNICAZIONE

---

- referente.

Con emittente si intende colui che crea il messaggio utilizzando una specifica codifica, il destinatario deve essere in grado di recepire il messaggio inviato dall'emittente, decodificarlo e riuscire ad interpretarlo. Il canale è il mezzo trasmissivo sul quale il messaggio si propaga, questo può essere per propagazione delle onde sonore nell'aria, la scrittura su carta o su supporto digitale. Il codice è il modo con cui esprimiamo un certo messaggio: a parole (e relativo tono), tramite la scrittura o un'immagine. Con referente invece si indica l'oggetto della comunicazione.

s

In questi ultimi 5000 anni sono solo tre le importanti innovazioni nel settore della comunicazione:

1. invenzione della scrittura;
2. invenzione della stampa;
3. invenzione del telegrafo.

Queste tre grandi innovazioni hanno dato origine ad altrettante rivoluzioni nel campo comunicativo.

1. rivoluzione chirografica;
2. rivoluzione gutenberghiana;
3. rivoluzione industriale ed elettrica.

### 2.1 Nascita della scrittura

L'invenzione della scrittura risale a 5-6 mila anni fa. È opinione comune credere che la scrittura fosse nata presso le civiltà mesopotamiche e via via diffusa alle altre civiltà. Tuttavia con gli attuali sviluppi nell'archeologia si è scoperta la presenza di testi egizi risalenti al 3400 a.C. (le prime forme di scrittura mesopotamica risalgono al 3200 a.C.) e di scritture su terracotta da parte di civiltà europee che risalirebbero fino al 5400 a.C.

Prima della scrittura, l'unica possibilità che aveva l'uomo per trasmettere informazioni era per via orale. Questo metodo però era altamente inefficiente e nel tempo

## 2.2 La stampa: dalla bibbia di Gutenberg ai quotidiani

---



**Figura 2.1:** Ricostruzione di una tavoletta sumera raffigurante dei valori numerici

le informazioni si deterioravano fino a raggiungere un significato totalmente differente da quello iniziale. La scrittura quindi si affiancò al canale orale e in seguito videro la luce le prime tavolette di argilla incise (figura 2.1), che ancora oggi testimoniano gli avvenimenti del passato. Queste tavolette erano create da professionisti e la loro diffusione fu estremamente lenta.

Nell'età antica e medievale questo problema affliggeva ancora la società, solo una ristretta cerchia di colti era in grado di consultare i libri e quest'ultimi venivano prodotti dagli amanuensi (o copisti).

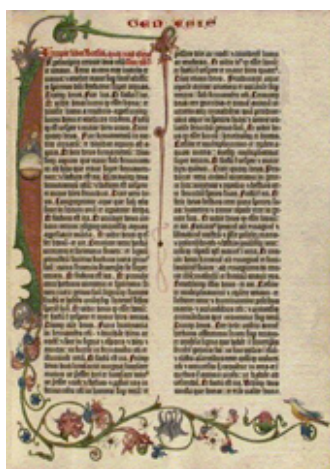
## 2.2 La stampa: dalla bibbia di Gutenberg ai quotidiani

Nel 1448, con l'introduzione della stampa a caratteri mobili da parte di Johann Gutenberg, ci fu una vera e propria rivoluzione nel campo della comunicazione di massa. La produzione di massa rese i libri molto più economici delle controparti prodotte dagli amanuensi e ne aumentò la diffusione in modo vertiginoso. Nonostante la maggior parte della società non avesse né le conoscenze adeguate per leggere un libro né tanto meno la possibilità economica di comperarlo. Questo periodo definì un grande slancio verso il futuro, la strada delle conoscenza era aperta a tutti gli individui e l'informazione circolava con maggiore velocità.

Nei due secoli successivi, verso la fine del '500 e inizi del '600 e soprattutto nel '700 da questa invenzione si svilupparono le riviste periodiche e i quotidiani. Questi furono i progenitori degli attuali strumenti di comunicazione di massa. Inizialmente la loro diffusione era minima ma toccarono l'apice durante il periodo illuminista e ben

## 2. STORIA DELLA COMUNICAZIONE

---



**Figura 2.2:** Una delle 1842 pagine della prima bibbia di Gutenberg

presto raggiunsero lo scopo per la quale erano stati creati. I primi periodici italiani furono: “*La Gazzetta veneta*” (1760), “*La Frusta letteraria*” (1763) e “*Il Caffè*” (1764) mentre il primo quotidiano fu il “*Diario Veneto*” (1765) un antenato dell’attuale “*Il Gazzettino*”.

Duecento anni fa però i quotidiani risentivano di un grosso problema: era impossibile a quei tempi fornire notizie in tempo reale, la velocità di diffusione di una determinata notizia era assai limitata e quindi gli avvenimenti potevano essere riportati anche con un notevole ritardo, basti pensare che una notizia proveniente dall’America doveva percorrere tutto l’oceano Atlantico, di conseguenza i giornali europei ne venivano a conoscenza solo molte settimane dopo. Al giorno d’oggi tutto questo non è più possibile, tuttavia, come facevano all’epoca, abbiamo la percezione di un evento da un unico punto di vista: quella del giornalista.

### 2.3 I cambiamenti socio-economici di fine 1800

Il momento cruciale che porterà ad una svolta sostanziale in moltissimi settori sarà il periodo della seconda rivoluzione industriale (1870 - 1914). Questa fase fu caratterizzata soprattutto dai un intenso sviluppo nelle vie di comunicazioni e dalle invenzioni che permisero di spostarsi più liberamente e velocemente: il motore sostituirà la vela nella navigazione e grandi tratte ferroviarie collegheranno interi continenti con un conseguente abbattimento dei tempi di percorrenza. Gli stati che per primi vissero questo

## 2.3 I cambiamenti socio-economici di fine 1800

---

grande cambiamento furono gli stessi che intrapresero la prima rivoluzione industriale, ossia: Gran Bretagna, Germania, Stati Uniti, Francia, Belgio, Svizzera ed in seguito l'Italia, il Giappone, la Russia ed i paesi scandinavi. Questi paesi controlleranno l'80% dell'intero mercato mondiale.

Le basi di questa rivoluzione furono in primo luogo le nuove invenzioni e le nuove tecnologie. Le invenzioni derivanti erano focalizzate a migliorare lo stile di vita dei cittadini. Tra le più celebri, usate tutt'oggi, vi sono: il frigorifero (1873), il ferro da stiro elettrico (1874), il ventilatore (1891), il rasoio Gillette e la cerniera-lampo (1895). Nel frattempo sorsero le primissime industrie chimiche allo scopo di produrre materie plastiche (la bachelite). Partendo dai progressi nel campo della chimica si sviluppò anche il settore farmaceutico.

Il fattore che accomunava molte nuove invenzioni era l'utilizzo dell'elettricità, infatti in questi anni vengono scoperte nuove fonti di energia, che andarono in parte a rimpiazzare il carbone fossile. Prime fra tutte furono: l'energia elettrica, derivata dall'impiego di turbine idrauliche e dalle dinamo, e il petrolio che comporterà gli studi sulla realizzazione del motore a scoppio da cui derivarono le prime possibilità di trasporto privato.

Tutto il periodo della seconda rivoluzione industriale fu caratterizzato da una continua produzione in massa. Era infatti necessario introdurre un nuovo modello di produzione per far fronte alla continua domanda dei prodotti. Tali prodotti dovevano essere standardizzati, cioè con caratteristiche uguali fra loro. Il nuovo modello impiegato fu denominato *catena di montaggio*, messo a punto dall'americano *Frederick W. Taylor* (da cui il movimento prese il nome di taylorismo) nel 1893. La catena di montaggio garantiva una produzione maggiore con tempi e prezzi sempre più ridotti. L'imminente espansione industriale che ne derivò attirò molti operai dalle campagne alle città, da cui conseguì una rapida crescita urbanistica.

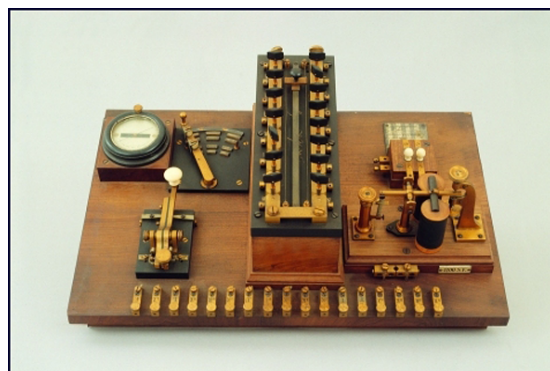
Per questi motivi la seconda metà del XIX secolo fu un periodo di grande fervore, definito anche con il termine "*Belle Époque*". Mai come allora il progresso e il benessere economico apparvero strettamente collegati allo sviluppo scientifico. Questo fu la causa principale di un ottimismo dilagante e di speranza nei confronti delle scienze sperimentali.

## 2. STORIA DELLA COMUNICAZIONE

---

### 2.4 Dai telegrafi al web

L'invenzione del telegrafo fece da spartiacque tra la vecchia concezione della comunicazione a distanza basata sulla scrittura a stampa e sulla rete postale e quella nuova, basata sui fondamenti della rivoluzione elettrica. I primi telegrafi vennero inventati nei primi anni '40 del 1800. Il più famoso tra tutti fu il telegrafo elettrico di Morse (figura 2.3). Questo era in grado di inviare, tramite un unico filo, impulsi elettrici opportunamente codificati per formare sequenze di caratteri alfanumerici. La prima trasmissione ufficiale venne effettuata il 24 maggio 1844 tra le città di Washington e Baltimora. La telegrafia su fili divenne talmente importante che si instaurò un collegamento telegrafico intercontinentale tramite cavi sottomarini. Ancora oggi tali cavi vengono ampiamente utilizzati.



**Figura 2.3:** Esempio di un telegrafo ottocentesco a filo

Tra il 1890 e il 1905 gli studi sui campi elettromagnetici applicati allo sviluppo dei telegrafi diedero vita ai telegrafi senza filo. I personaggi storici che si contesero la scoperta dell'invenzione furono: *Guglielmo Marconi*, *Nikola Tesla* e *Alexandr Stepanovich Popov*. Tutt'oggi la questione è ancora aperta poiché si sono presentate molte dispute legali sui brevetti di Marconi.

I primi telegrafi senza fili non erano in grado di inviare segnali audio ma si limitavano a segnali elettrici del tipo *on/off*, questi li rendeva ideali per trasmettere messaggi utilizzando il codice Morse. I primi esemplari irradiavano onde elettromagnetiche generate da un dispositivo chiamato *spinterometro*, costituito da due elettrodi in grado di produrre scariche elettriche nell'aria. Il campo elettromagnetico generato era intercettato

da un'antenna posta ad una certa distanza (300 - 400 metri) su cui era collegato un *co-esore*, una specie di rilevatore di onde elettromagnetiche realizzato da *Calzecchi Onesti* e costituito da un tubetto di vetro, al cui interno erano presenti due elettrodi e delle polveri di nichel e argento.

Questi esperimenti andarono sempre più a perfezionarsi, specialmente con l'invenzione dei tubi elettronici (1910 - 1915) che fecero da padroni indiscussi durante l'avvento della radio broadcasting. Con l'introduzione del telegrafo e del telegrafo senza fili, per la prima volta nella storia dell'umanità, le "parole" vengono veicolate in pochissimi secondi per raggiungere destinazioni impensabili. Questi due dispositivi rendevano possibile una comunicazione istantanea fra città distanti fra loro, cosa che solo pochi decenni prima era un puro sogno. I telegrafi consentivano l'invio di informazioni importanti fra città o comunicare con le navi dislocate sull'oceano, però non erano adatti per essere impiegati come strumenti di comunicazione di massa. La risposta a questo deficit fu l'invenzione della radio, che dagli anni '20 in poi rappresentò il più importante mezzo di comunicazione di massa dopo il cinema. I principi funzionali utilizzati dalla radio sono gli stessi della radiotelegrafia con l'aggiunta della possibilità di ricevere contenuti sonori per mezzo di onde radio. La prima trasmissione venne effettuata nel 24 dicembre del 1906, ma solo una decina di anni dopo vennero a crearsi le prime stazioni radio fisse. In Italia, come in altri stati con regime totalitario, la radio venne vista come uno strumento estremamente importante per diffondere un'incessante propaganda sul regime. Come altri mass media, la gestione delle trasmissioni radiofoniche passò sotto il controllo dello stato.

Non dobbiamo dimenticarci l'invenzione del telefono (figura 2.4) attribuita all'italiano *Antonio Meucci*, chiamato da lui telettrofono, che già nel 1871 ne dimostrò il funzionamento.

Dalla riproduzione di contenuti audio si passò a quelli audio e video con l'invenzione del televisore. Già nel 1925 un inventore scozzese, *John Logie Baird*, riuscì ad inviare a distanza un'immagine televisiva formata da 28 linee. Il suo progetto faceva utilizzo del *disco di Nipkow*, e proprio per questo, la televisione di Baird veniva definita televisione elettromeccanica a differenza di quella elettronica, basata sul tubo catodico realizzato nel 1927. Ovviamente questo era solo l'inizio e dovettero trascorrere diversi anni prima che questa nuova tecnologia diventasse operativa su larga scala.

In Italia le prime trasmissioni televisive comparvero nel 1934, in via sperimentale, solo

## 2. STORIA DELLA COMUNICAZIONE

---



**Figura 2.4:** Telefono svedese di fine 1800

nel 1954 la RAI cominciò a trasmettere regolarmente in bianco e nero e nel 1956 il segnale arrivò su territorio nazionale.



**Figura 2.5:** Il primo televisore italiano (1936)

È interessante il paragone tra tutti gli strumenti di comunicazione di massa elettrici (radio, televisione) e la stampa. La stampa richiede che il lettore sia innanzitutto alfabetizzato e poi che abbia una spirito piuttosto critico verso il giornale. Radio e televisione invece non presuppongono nessuna alfabetizzazione da parte del destinatario. Tutti possono comprendere le informazioni trasmesse.

Al giorno d'oggi, tutti noi disponiamo della struttura comunicativa più vasta e più importante di tutti i tempi, le *reti di calcolatori*. Al vertice ovviamente troviamo la rete delle reti: *Internet*.

Ognuna di queste innovazioni, dal telegrafo ad Internet ha permesso di rendere l'informazione più accessibile al destinatario, più veloce e meno costosa. Bisogna tuttavia



## **2.4 Dai telegrafi al web**

---

sottolineare che le nuove tecnologie non rimpiazzano quelle precedenti, anzi si sono affiancate fra loro. La stampa di Gutenberg non ha sostituito la scrittura a mano e l'avvento della comunicazione elettrica e digitale non si è sostituita ai giornali cartacei o alla carta in generale.

## 2. STORIA DELLA COMUNICAZIONE

---

## 3

# Web 2.0

*Il Web 2.0 siamo noi che cooperiamo*

Per capire le differenze fra Web 1.0 e Web 2.0 è necessario comprendere com'è nato il Web e cosa offriva nei suoi primi anni di vita.

### 3.1 Web: cenni storici

Il termine Web è una contrazione dell'ormai famoso *Word Wide Web* che tradotto letteralmente significa “grande ragnatela mondiale”. Il Web nasce da un'idea di Tim Berners-Lee e Robert Cailliau, quando i due lavoravano al CERN (*European Organization for Nuclear Research* tradotto in inglese). Diversamente da altri avvenimenti storici importati, la nascita del Web può essere fatta coincidere con un momento preciso: il 6 agosto 1991. Data in cui Tim B. Lee pubblicò il primo sito web (vedi fig. 3.1).

Tre anni dopo lo stesso Tim B. Lee, definito anche come il padre del Web, fondò un'associazione ad hoc per contribuire allo sviluppo e al mantenimento del World Wide Web chiamata *World Wide Web Consortium* (W3C) e risiedente al MIT (*Massachusetts Institute of Technology*).

### 3.2 Web 1.0

I primi siti che formavano il Web contenevano per la maggior parte elementi testuali e grafici, elementi che dovevano rispettare gli standard definiti dal W3C in modo da

### 3. WEB 2.0

---

#### World Wide Web

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#) , [Policy](#) , November's [W3 news](#) , [Frequently Asked Questions](#) .

[What's out there?](#)  
Pointers to the world's online information, [subjects](#) , [W3 servers](#), etc.

[Help](#)  
on the browser you are using

[Software Products](#)  
A list of W3 project components and their current state. (e.g. [Line Mode](#) ,[X11 Viola](#) , [NeXTStep](#) , [Servers](#) , [Tools](#) , [Mail robot](#) , [Library](#) )

[Technical](#)  
Details of protocols, formats, program internals etc

[Bibliography](#)  
Paper documentation on W3 and references.

[People](#)  
A list of some people involved in the project.

[History](#)  
A summary of the history of the project.

[How can I help?](#)  
If you would like to support the web..

[Getting code](#)  
Getting the code by [anonymous FTP](#) , etc.

**Figura 3.1:** Il sito pubblicato da Tim Berners-Lee

consentire la visualizzazione corretta di tutti i siti tramite un unico software: il browser. Oggi invece gli sviluppatori e i designer tendono ad andare controtendenza, creando siti web che fanno largo uso di elementi video/audio per il quale è necessario l'utilizzo di software di terze parti per ottenere una visualizzazione totale o corretta. Questi mini-software sono chiamati *plugin*. Ecco alcuni esempi: Adobe Flash, Microsoft Silverlight, QuickTime, RealMedia, . . .

I siti che vennero creati nell'arco degli anni '90 soffrivano di un'enorme limitazione, il linguaggio con il quale erano creati, ossia HTML (*HyperText Markup Language*) poneva grandi limiti alla dinamicità, poiché tale linguaggio definiva sì la struttura e il contenuto delle pagine ma non consentiva nessuna interazione con l'utente. Inizialmente questo problema fu ovviato dall'utilizzo di CGI (*Common Gateway Interface*) che davano la possibilità al web server di invocare un'applicazione esterna e di visualizzare il risultato dell'operazione tramite una pagina HTML. Tuttavia questa tecnica comportava ancora molte limitazioni: come la non gestione delle sessioni, che al giorno d'oggi ricopre un ruolo fondamentale durante la navigazione web. Per eliminare queste limitazioni e dare al sito una dinamicità totale vennero implementati nuovi linguaggi rivolti al lato web server come il PHP, ASP, JSP, Perl, Python, . . . Tali linguaggi garantiscono un'interazione con l'utente, ma allo stesso tempo anche l'accesso e l'utilizzo di database, dal quale vengono estratti i dati tramite apposite query, per poi essere visualizzati sotto forma di pagina HTML.

Parallelamente vi è stato un notevole sviluppo del lato client legato all'evoluzione del linguaggio HTML e dall'interpretazione di linguaggi di scripting, come *Javascript*. Molto utilizzato al giorno d'oggi è l'AJAX (acronimo di *Asynchronous JavaScript and XML*), ossia un insieme di tecnologie web formalizzate per la prima volta nel 2005 in un articolo scritto da *Jesse James Garrett*, presidente e fondatore di *Adaptive Path*. Tramite AJAX lo scambio di dati fra web server e browser avviene in background, ed è completamente trasparente all'utente. In questo modo si ottiene un aggiornamento dinamico della pagina senza che l'utente ne forzi un ricaricamento. Le nuove tecniche sia su lato server che su lato browser hanno favorito il passaggio da un Web 1.0 statico a un Web 2.0 dinamico ed interattivo per l'utente.

### 3.3 Web 2.0

Il termine Web 2.0 fa la sua comparsa solo nel 2004, da qui possiamo capire che questo concetto è relativamente giovane. Durante le numerose riunioni, molte sono state le spiegazioni e definizioni date sul termine Web 2.0, tuttavia, ancora oggi questo si nasconde dietro un velo di mistero. Il problema principale della definizione del "Web 2.0" è la sua assoluta genericità. La locuzione 2.0 può far richiamare alla mente il concetto di *controllo versione*, utilizzato nell'ambito software per identificare un'evoluzione nel software rispetto alle versioni precedenti. In realtà non è così, poiché il Web 2.0 non porta con sé alcuna evoluzione nel campo tecnologico o nel campo dei linguaggi di programmazione. La rete continuerà ad utilizzare il protocollo di comunicazione TCP/IP e HTML per la visualizzazione delle pagine. Quindi è sbagliato associare a Web 2.0 il significato di *Internet 2*. Il termine Internet 2 sta ad indicare tutt'altra cosa ed in particolare è stato coniato da un consorzio no profit riunitosi nel 1996, a cui parteciparono 208 università con l'intento di sviluppare nuove tecnologie per la rete e il trasferimento di dati ad alta velocità.

Il termine "Web 2.0" nasce dalla mente di Dale Dougherty nel 2004 durante la sessione di lavoro con Tim O'Reilly. Tuttavia è a Tim che si riconosce il merito per aver tentato di dare una definizione chiara ed esauriente. La prima definizione da lui data è citata di seguito:

"Il Web 2.0 è la rete come piattaforma, attraverso tutti i dispositivi collegati; le applicazioni Web 2.0 sono quelle che permettono di ottenere la maggior

### 3. WEB 2.0

---

parte dei vantaggi intrinseci della piattaforma, fornendo il software come un servizio in continuo aggiornamento che migliora più le persone lo utilizzano, sfruttando e mescolando i dati da sorgenti multiple, tra cui gli utenti, i quali forniscono i propri contenuti e servizi in un modo che permette il riutilizzo da parte di altri utenti, creando una serie di effetti attraverso un “architettura della partecipazione” e andando oltre la metafora delle pagine del Web 1.0 per produrre così user esperienze più significative”

Da questa prima definizione possiamo estrapolare i concetti di:

- la rete come piattaforma;
- software come servizio;
- utilizzo dei dati da sorgenti multiple (tecnica del mashup);
- produzione di contenuti da parte degli utenti;
- partecipazione.

Tim non si limita a questa definizione e nel novembre 2006 la ripropone come segue:

“Il Web 2.0 è un insieme di tendenze economiche, sociali e tecnologiche che formano insieme la base per la prossima generazione di Internet - un più maturo e distinto mezzo caratterizzato dalla partecipazione degli utenti, dall’apertura e dagli effetti della rete.”

Da qui capiamo che i concetti base del Web 2.0 dovrebbero essere:

- tendenze economiche, sociali e tecnologiche;
- partecipazione;
- apertura della rete.

Web 2.0 indica un concetto astratto, tanto che nemmeno chi il termine l’ha praticamente coniato è in grado di dare una definizione precisa ed esauriente. Nonostante questo è possibile individuare degli elementi principali, l’insieme di queste elementi formeranno in sintesi un prodotto Web 2.0. Tra i principali abbiamo:

- la rete è più orientata verso l’utente e l’interazione sociale;



### 3. WEB 2.0

---



Figura 3.3: Alcuni dei loghi più famosi del Web 2.0

queste persone troviamo Tim Berners-Lee, che ha preso una posizione piuttosto chiara, secondo la quale il Web 2.0 non esisterebbe.

### 3.4 Gerarchia del Web 2.0

La suddivisione in livelli è stata fornita da Tim O'Reilly, e all'interno di questi livelli possiamo catalogare qualsiasi servizio presente sul web.

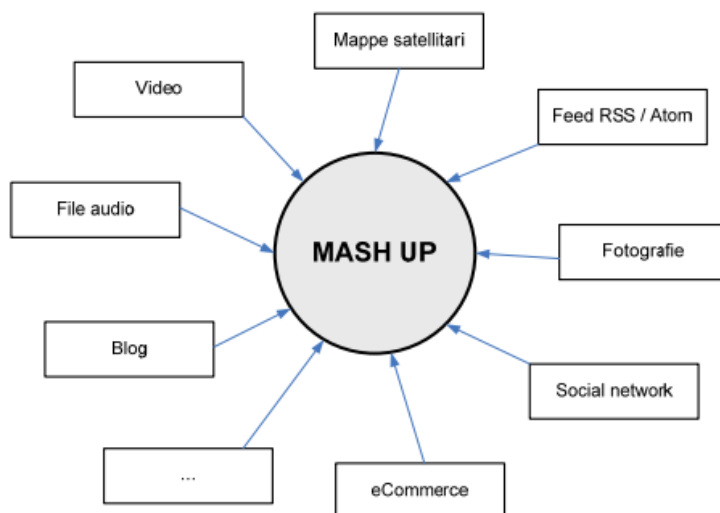
- *Livello 3*: in questo livello risiedono tutte le applicazioni che esistono esclusivamente sul web e si basano sul collegamento tra individui e altre applicazioni. Sono guidate dal lavoro svolto dalla comunità e di conseguenza hanno la necessità di essere utilizzate online. Alcuni esempi sono Wikipedia, eBay, Facebook, ...;
- *Livello 2*: risiedono tutte le applicazioni che potrebbero esistere offline, ma sono fortemente avvantaggiate quando hanno una presenza sul web. Ad esempio: Flickr e YouTube;
- *Livello 1*: fanno parte di questo livello tutte le applicazioni che funzionano benissimo offline, con la possibilità di ottenere un beneficio e funzionalità aggiuntive se utilizzate online. Un esempio è iTunes che mette a disposizione uno dei più grandi music store del mondo. Altri esempi sono Scribd o Google Docs, che ci permettono di condividere i nostri documenti online;



- *Livello 0*: l'applicazione funziona sia offline, avendo però a disposizione i dati in locale o nelle cache, sia online. La più famosa web application di questo tipo è Google Maps.

### 3.5 Mashup

La tecnica del mashup è utilizzata moltissimo dai siti che aderiscono in pieno alla filosofia del Web 2.0 e la utilizzano per includere dinamicamente informazioni provenienti da altre fonti. Il termine stesso, *mashup*, sta ad indicare “poltiglia”, ed utilizzato nell’ambito web indica un nuovo servizio creato mescolando tra loro le informazioni prelevate da più sorgenti. I mashup stanno sempre più rivoluzionando il mondo web in quanto permettono a chiunque di combinare dati proveniente da eBay, Google, YouTube, Yahoo!, Flickr, Amazon, ... in modo disparato ed innovativo.



**Figura 3.4:** Alcuni contenuti che possono combinarsi per dare vita ad un mashup

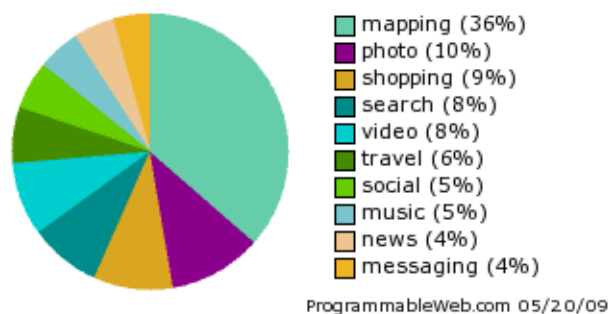
La facilità di progettazione dei mashup è un fattore fondamentale, in quanto non è richiesta nessuna competenza tecnica per la loro realizzazione. Per i meno esperti esistono soluzioni grafiche che permettono di realizzare mashup in pochi click. I più famosi sono Google Mashup Editor, Microsoft Popfly o Yahoo Pipes. Solitamente i mashup vengono realizzati utilizzando le API pubbliche<sup>1</sup> che gli amministratori del

<sup>1</sup>Le API (Application Programming Interface) offrono dei set di istruzioni base che vanno gestite dai comuni linguaggi di scripting su lato server.

### 3. WEB 2.0

---

servizio mettono a disposizione, tramite un semplice JavaScript oppure infine utilizzando le feed RSS. Un classico esempio è quello derivato dall'unione del servizio Google Maps (servizio di visualizzazione delle mappe) e Flickr (servizio di hosting e condivisione delle foto). Unendo questi due servizi otterremo un nuovo servizio che ci consentirà di geolocalizzare una foto, ossia legare tra loro il contenuto della foto e il luogo in cui è stata scattata. Nella figura 3.5 sono riportate le percentuali dei mashup più ricorrenti.



**Figura 3.5:** Percentuali dei mashup più realizzati secondo Programmableweb.com

I mashup si dividono in tre categorie principali:

- *Consumer Mashup*: semplice mashup con l'unione di due o più servizi combinati in un'unica grafica;
- *Data Mashup*: unione di diverse fonti di informazioni eterogenee con lo scopo di creare un unico contenitore di dati. Un esempio può essere la funzione "Bookmarks" presente nei browser o le informazioni contenute nelle feed RSS;
- *Business Mashup*: questo tipo di mashup è una combinazione di Data e Consumer Mashup al fine di realizzare un'applicazione enterprise. Solitamente questa tipologia viene utilizzata per monitorare le vendite, il personale o i costi interni di un'azienda.

## 4

# I CMS

*Internet è come un immenso magazzino (di informazioni), ma non può costituire di per sé la memoria. (Umberto Eco)*

Come descritto nel capitolo 3 con il passaggio dal Web 1.0 al Web 2.0 l'utilizzo di piattaforme CMS e Wiki è aumentato notevolmente a favore di una maggiore dinamicità delle informazioni. Definiamo ora il significato di CMS.

Un CMS (acronimo di *Content Management System*) è un sistema di gestione dell'organizzazione e della stesura dei contenuti web. Possiamo classificare un CMS come una sorta di strumento software che risiede su un *web server* con lo scopo di agevolare le operazioni di inserimento di nuovi contenuti. È quindi fondamentale specificare il concetto di contenuto. Un contenuto è la quantità minima di informazione gestita da un CMS. L'informazione può essere un testo, un'immagine, un grafico, un video o una musica, praticamente qualsiasi informazione in formato digitale ed atta ad essere condivisa. Il termine *contenuto* è completamente differente dal termine *dato*. Un contenuto può essere espresso tramite un insieme di dati, in modo da costituire un'informazione completa. Ad esempio il titolo di una news non può essere considerato come un contenuto bensì lo possiamo interpretare come un dato che, in seguito, con l'aggiunta del corpo, il nome dell'autore, la data di pubblicazione ed eventuali elementi grafici daranno vita ad un contenuto.

Solitamente tutti i CMS si caratterizzano dalla facilità d'uso, poiché non sono richieste particolari competenze tecniche di programmazione per poterlo utilizzare. L'amministratore del sito web non ha la percezione del "sottosistema tecnico", cioè

## 4. I CMS

---

le procedure richiamate, o la struttura del database sottostante, in gergo questa peculiarità viene chiamata *trasparenza d'uso per l'utente* ed è uno dei principali punti di forza dei CMS. Di conseguenza il responsabile del sistema non necessariamente deve avere approfondite conoscenze su linguaggi di programmazione come PHP (acronimo ricorsivo di *PHP Hypertext Preprocessor*), Javascript o HTML (*HyperText Markup Language*).

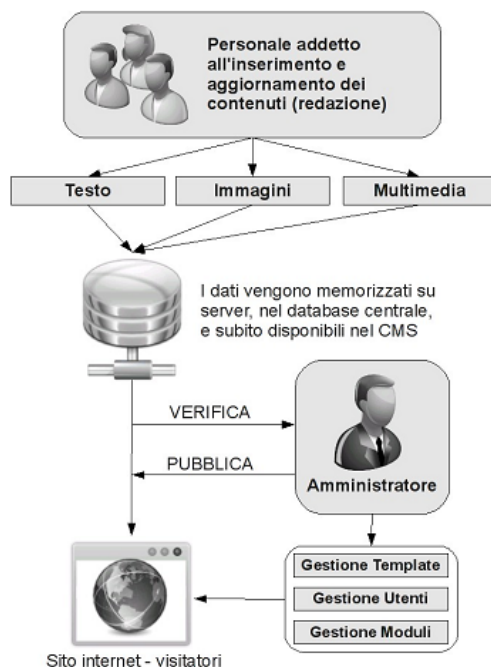
### 4.1 I Web Content Management System

Fino ad ora abbiamo trattato di CMS; tuttavia oggi con questo termine facciamo riferimento soprattutto a sistemi WCMS (*Web Content Management System*), ossia piattaforme per la gestione dei contenuti interamente online. Con questi sistemi possiamo amministrare qualsiasi tipo di contenuto presente sul nostro sito da remoto avendo a disposizione un computer con una connessione ad internet ed un browser. I contenuti tramite l'ausilio di opportuni linguaggi di programmazione dinamici, come il PHP, e l'utilizzo di database vengono resi disponibili istantaneamente garantendo una dinamicità al sito che si contrappone alla staticità dei vecchi siti realizzati interamente in HTML. Nella figura 4.1 è rappresentato schematicamente il funzionamento di un WCMS. Nel caso illustrato vi è una divisione netta tra la redazione, individui autorizzati a produrre e gestire nuovi contenuti e l'amministratore del sistema. In molti casi però, soprattutto nei blog, il ruolo della redazione e dell'amministratore coincidono e fanno entrambi riferimento allo stesso individuo.

Tutti i WCMS sono divisi in due sezioni ben marcate fra loro: back end e front end.

- la sezione di *back end*, ossia la sezione d'amministrazione, consente all'amministratore o ad altri utenti con privilegi speciali (moderatori, scrittori ...) di poter inserire, modificare, eliminare contenuti presenti sul sito;
- la sezione di *front end* è la parte pubblica del sito in cui tutti gli utenti possono accedervi per consultare i contenuti presenti.

Oggi giorno i WCMS sono alla base del funzionamento di moltissimi siti web. Possiamo quindi suddividerli in diverse tipologie, ognuno si caratterizza dall'ambito in cui verrà utilizzato, di conseguenza abbiamo una suddivisione netta in due famiglie. La prima costituisce i *WCMS ad hoc*, utilizzati per realizzare specifiche piattaforme web



**Figura 4.1:** Schematizzazione funzionale di una piattaforma WCMS

come blog, enciclopedie, forum e siti di social network. La seconda invece è formata da WCMS più flessibili che consentono la gestione di più tipi di contenuti. La tabella 4.1 posta di seguito illustra le piattaforme più famose disponibili per realizzare una determinata tipologia di sito web.

Quasi tutte le voci riportate in tabella fanno riferimento a WCMS *open source*, ossia piattaforme libere, in cui la consultazione e la modifica del codice sorgente è possibile. La maggioranza degli amministratori di siti web utilizza questi sistemi poiché non richiedono né un esborso economico per l'acquisto né spese ulteriori per la manutenzione, per via delle numerosissime community che forniscono documentazione ed aiuto. Un altro motivo per il quale difficilmente vengono utilizzati WCMS proprietari è dato dal linguaggio PHP. Questo linguaggio è interpretato dall'interprete PHP presente sul web server e non compilato, come ad esempio il *C* o il *C++*, questo fa sì che una volta acquistata la licenza della piattaforma il cliente possa comunque visionare il codice sorgente, con l'eccezione che la riproduzione di tale codice sarà legalmente perseguibile. Per ovviare a questi problemi alcune società impiegano opportune tecniche di offusca-

## 4. I CMS

---

Tipologia siti	Piattaforma disponibile
Blog	WordPress
Enciclopedia online	MediaWiki (base per il progetto Wikipedia) WikkaWiki
Forum	Phpbb Simple Machine Forum vBulletin (a pagamento) Invision Power Board (a pagamento)
Portale	Joomla! Plone Drupal Mambo XOOPS

**Tabella 4.1:** I CMS più utilizzati

mento o di crittografia del codice sorgente, rendendo impossibile la consultazione di quest'ultimo al cliente.

### 4.2 Separazione dei dati

In tutti i CMS vi è una differenza sostanziale su come i dati vengono memorizzati ed in seguito presentati. L'insieme di dati che forma i contenuti e persino le impostazioni della piattaforma stessa vengono comunemente salvate all'interno di un *database* e non in pagine HTML o PHP, se fosse così i CMS non avrebbero ragione di esistere. Con l'utilizzo di un database possiamo vantare di:

- effettuare ricerche efficienti sui dati memorizzati;
- accesso ai dati con diverse modalità;
- effettuare un mashup dei dati come contemplato dalla filosofia del Web 2.0;
- facile gestione dei privilegi;
- possibilità di effettuare elaborazioni secondarie (elaborazioni statistiche).

Nella fase di presentazione, il WCMS eseguirà query specifiche al DBMS (*Database Management System*) per richiedere tutti i dati che formeranno il contenuto da presentare. I dati richiesti transiteranno attraverso “pagine modello” chiamate *template* in cui tutte le informazioni verranno rielaborate per ottenere la modalità di presentazione voluta. Modificando i template otterremo diverse tipologie di presentazione (vedi figura 4.2).

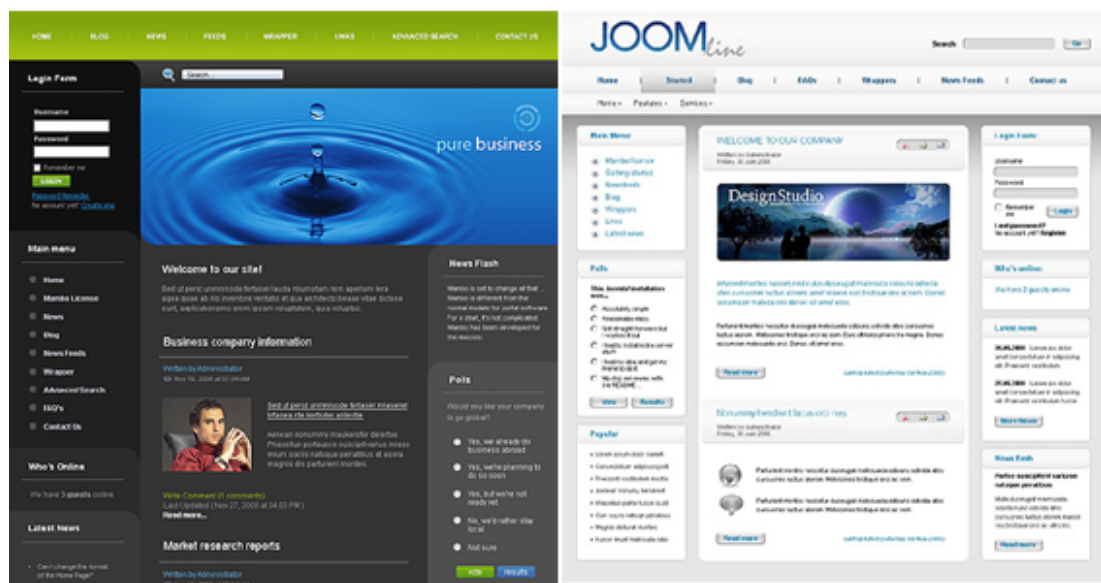


Figura 4.2: Due diversi template per Joomla! a confronto

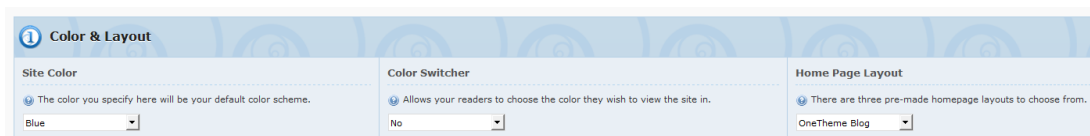
Questa modalità di presentazione dei dati è completamente opposta a quella utilizzata dai siti statici di vecchia concezione composti da pagine HTML e fogli di stile CSS (*Cascading Style Sheet*) in cui i contenuti erano fortemente legati alla modalità di presentazione. Con i WCMS i contenuti e l’aspetto grafico sono completamente dissociati. L’amministratore o la redazione che scrive i contenuti non deve preoccuparsi di come tali contenuti verranno poi presentati. Sarà cura dell’amministratore scegliere un template, sviluppato anche da terze parti, che fornirà la base di presentazione delle informazioni presenti sul sito.

Il template nella maggior parte dei Web CMS è un insieme di file con estensione .php, .css, eventuali immagini e script che forniranno all’amministratore la possibilità di presentare i dati in modi differenti. Un template che consente all’*admin* di persona-

## 4. I CMS

---

lizzare alcune sezioni del sito, dando la possibilità di attivare o disattivare dei moduli introdurrà il concetto di *modularità*.



**Figura 4.3:** Piccola parte di un template modulare per piattaforma WordPress

### 4.3 Categorie di utenti

Tutti i CMS devono fornire diversi livelli di accesso e di conseguenza adottano un sistema di suddivisione degli utenti. Abbiamo già parlato di *front end* e *back end* nel paragrafo 4.1 e sarebbe un guaio se un *utente guest* (ospite) potesse avere accesso alla sezione d'amministrazione del sito. Proprio per motivi di sicurezza i CMS suddividono gli utenti in diversi livelli, il riconoscimento dell'utente avviene tramite l'inserimento di un nome utente e password o tramite un sistema di network distribuito e decentralizzato (novità recente e poco utilizzato, il più famoso è OpenID). Tutte le informazioni riguardanti l'utente, quindi nome utente, nome reale dell'utente, e-mail, password, . . . sono memorizzate nel database in apposite tabelle. Per ovvi motivi di sicurezza la password non è memorizzata in chiaro ma vengono utilizzate complicate tecniche crittografiche, come l'algoritmo *MD5*, *SHA*, . . . per garantire che la password non cada in mani sbagliate.

Gli utenti registrati su un CMS vengono suddivisi fra i vari gruppi presenti, ovviamente ogni CMS definisce i suoi gruppi ma ne esistono di standard, presenti nella maggior parte delle piattaforme. Questi sono i principali:

- *Guest* (visitatore): utente non registrato che può consultare solamente i contenuti presenti. In certi casi l'amministratore non consente ai visitatori di consultare il sito senza prima registrarsi. Nessun'altra interazione è consentita.
- *Registered User* (utenti registrati): utenti che possono essere identificati univocamente durante la fase di accesso al sito. Hanno autorizzazioni sufficienti per visualizzare tutti i contenuti, possono accedere ad alcune aree riservate, oppure



gli è consentito interagire scrivendo commenti alle notizie, creando nuove discussioni e rispondendo ai messaggi in un forum. In alcuni WCMS, come WordPress, il gruppo utenti registrati è quasi inutile poiché anche ai visitatori è concessa la possibilità di rilasciare commenti.

- *Author* (autori): utenti privilegiati che hanno la capacità di inserire e modificare contenuti presenti sul sito. Nel caso dei forum vengono chiamati *moderatori* ed hanno la capacità di inibire l'interazione degli altri utenti con privilegi minori.
- *Admin* (amministratori): utenti che hanno il pieno controllo della piattaforma CMS. Ha un controllo totale su tutte le funzioni, su tutti gli utenti registrati, su tutti i contenuti, all'admin non è posto nessun limite. Solitamente l'amministratore è colui che possiede o noleggia il web server.



Figura 4.4: Struttura dei gruppi in un CMS generico

## 4.4 CMS e Wiki a confronto

Come visto in 4.1 i sistemi Wiki vengono utilizzati per realizzare enciclopedie digitali o documentazioni. Questa piattaforma rientra perfettamente nella filosofia del Web 2.0 e dopo i blog è uno degli strumenti più utilizzati. La definizione data da Wikipedia (il wiki più famoso al mondo) sul termine *Wiki* è la seguente:

“Un wiki è un sito web (o comunque una collezione di documenti ipertestuali) che permette a ciascuno dei suoi utilizzatori di aggiungere contenuti,

## 4. I CMS

come in forum, ma anche di modificare i contenuti esistenti inseriti da altri utilizzatori.”

Il termine Wiki ha origine hawaiana e sta ad indicare qualcosa di rapido, infatti questa piattaforma ha come punto di forza la velocità e l'immediatezza. Wiki si basa su un insieme di pagine statiche non organizzate in modo gerarchico e lineare, ma tutte sullo stesso livello di importanza. Sebbene CMS e Wiki apparentemente sembrano piattaforme atte a svolgere il medesimo compito, i due termini non sono nemmeno paragonabili. In primo luogo tutti gli articoli presenti in una Wiki devono essere resi modificabili dagli altri utenti, questo non è fattibile in un sistema CMS, in cui solo l'amministratore e gli autori possono modificare e aggiungere contenuti. In secondo luogo non possiamo più parlare di una netta suddivisione degli utenti in gruppi, come analizzato nel paragrafo 4.3. L'utente registrato, in una Wiki, può svolgere il compito di un autore.

La differenza più marcata però è che mentre in una piattaforma Wiki la concentrazione è rivolta maggiormente nella fase di realizzazione e di modifica di un articolo, il CMS deve anche gestire la sua catalogazione logica e presentazione grafica all'interno di un livello grafico coerente.

Una Wiki al giorno d'oggi è utilizzata in moltissime occasioni, per esempio quando si vuole stilare una documentazione tecnica che deve essere scritta da più individui.

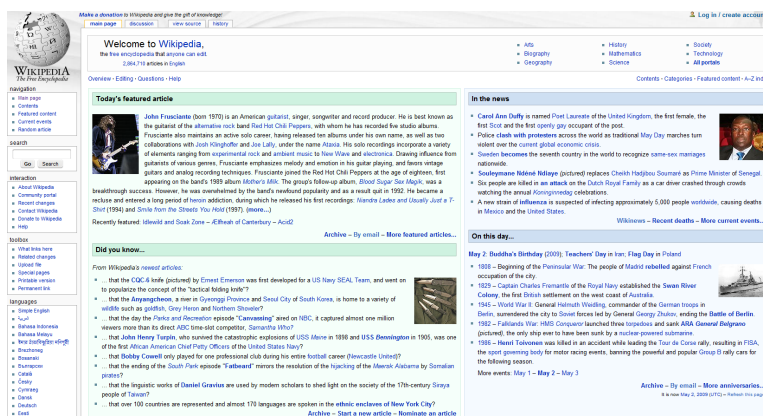


Figura 4.5: Homepage del progetto Wikipedia English

### 4.5 CMS: un po' di storia

I CMS sono nati negli Stati Uniti, la loro datazione è difficile da determinare, i primi esemplari risalgono al 1995, anno in cui la *Vignette Corp.* viene fondata. La Vignette forniva un primitivo CMS che rendeva semplice e personalizzata la pubblicazione dei nuovi contenuti. L'anno successivo Vignette fornì a CNET, tutt'oggi una compagnia leader nel settore tecnologico informatico, il proprio software che fu impiegato per la gestione del sito news.cnet.com. Un sito dedicato all'informazione tecnologica con più di 30 articoli al giorno, tra cui: recensioni, analisi, report, materiale audio/video, il tutto aggiornato 24 ore su 24. La prima piattaforma utilizzata prese il nome di *Story Server*, nata dalla fusione tra la tecnologia di Vignette e quella di *Prism*. Il software derivatone facilitò il lavoro degli autori dando loro la possibilità di inserire articoli in modo semplice e diretto senza l'intervento di tecnici. Piattaforme come quella descritta sopra hanno dato vita all'evoluzione nel mondo dei CMS. Grandi sviluppi sono stati apportati negli anni tra il 1999 e il 2002, in cui sono nati importanti WCMS open source tra cui *PHP-Nuke* (1999), *Mambo* (che darà origine al fork Joomla!), *WebGUI* (2001) e *XOOPS* (2002). Una lista dei CMS open source, e non, è disponibile all'indirizzo [http://en.wikipedia.org/wiki/List\\_of\\_content\\_management\\_systems](http://en.wikipedia.org/wiki/List_of_content_management_systems).

#### 4. I CMS

---

## 5

# Blog & WordPress

*Express yourself. Start a blog. (WordPress Slogan)*

*Esprimiti. Inizia a scrivere un blog.*

WordPress è il nome di una famosa piattaforma di *personal publishing*. La forma più comune di pubblicazione personale è il *blog*. WordPress infatti nasce con lo scopo di facilitare la creazione e il mantenimento di un blog. Prima di addentrarci nell'analisi di questa piattaforma è doveroso soffermarsi sul significato del termine blog. Innanzitutto il termine nasce dalla contrazione tra *web* e *log* (diario, giornale). Solitamente i blog sono siti personali (gestiti da un individuo) e a tema. Proprio da quest'ultimo aspetto nasce una vasta differenziazione tra blog e blog. Possiamo quindi avere:

- blog personali: è la categoria più diffusa. Ogni giorno l'amministratore dovrebbe inserire nuovi articoli riguardanti l'esperienze vissute;
- blog realizzati da una comunità: realizzati da più utenti che collaborano in un campo d'interesse comune;
- blog tematici: è la tipologia di blog che esprime gli hobby e le passioni dell'amministratore;
- photoblog: blog dedicato a soli contenuti fotografici;
- blog politico: blog che trattano argomenti politici in ogni forma, vengono utilizzati per diffondere idee e soluzione. La collaborazione dei visitatori è necessaria;

## 5. BLOG & WORDPRESS

---

- audio blog: blog dedicati al podcasting, ossia brevi registrazione che possono essere scaricate ed ascoltate in un qualsiasi riproduttore musicale;
- video blog (vblog): blog che fa largo uso di video e foto. Si avvale di servizi di video sharing come YouTube, Google Video, Metacafe, ...
- ...

L'insieme di tutti i blog presenti nella rete forma la *blogsfera*. Il termine è dovuto al fatto che i blog sono interconnessi fra loro, e i bloggers in moltissimi casi leggono e prendono ispirazione da articoli di altri bloggers.

Gli articoli (chiamati *posts*) vengono visualizzati in ordine cronologico e la piattaforma utilizzata ha il compito di mantenere organizzati tutti i vecchi posts dando la possibilità all'amministratore di revisionarli anche dopo un lungo periodo temporale. L'incessante nascita di nuovi blog, che da un paio d'anni sta caratterizzato tutto il web, ha favorito l'adozione delle RSS (acronimo di *Really Simple Syndication* o *RDF Site Summary*) su lato server e l'utilizzo di aggregatori RSS per una facile consultazione delle RSS su lato client.

Dal 2001 sono stati creati moltissimi servizi per la creazione e gestione di blog. Alcuni esempi sono: Blogger, Splinder, Libero, MySpace, Windows Live Spaces. Nel frattempo piattaforme specifiche per la gestione di blog sono comparse, una delle più famose è appunto WordPress che fa la sua prima apparizione nel 2003.

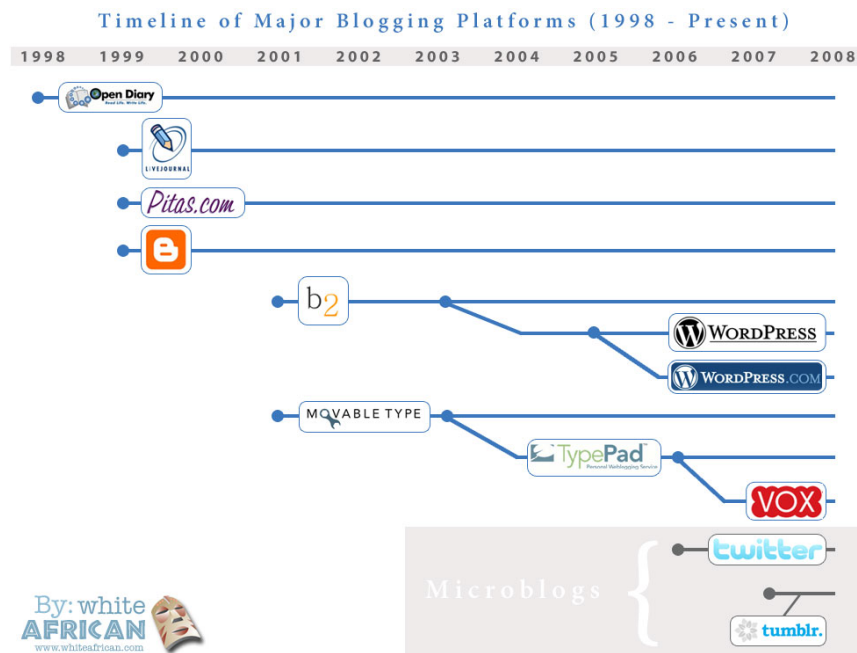


**Figura 5.1:** Il logo di WordPress

Come WCMS, WordPress rispecchia in pieno tutti i concetti analizzati nel capitolo precedente (cap. 4.1).

## 5.1 WordPress: un po' di storia

Il progetto WordPress nasce con l'intento di realizzare un sistema di pubblicazione personale elegante, veloce, aderente agli standard, personalizzabile e ben strutturato, basato sul linguaggio PHP e MySQL, ma soprattutto che aderisse alla filosofia open source e quindi soddisfacesse in pieno la licenza GPL (*GNU General Public License*).



**Figura 5.2:** Timeline delle maggiori piattaforme di blogging

WordPress è ufficialmente il successore di *b2/cafelog* (conosciuto anche con il nome di b2 o cafelog). Come possiamo vedere dalla figura 5.2, che rappresenta lo sviluppo nel tempo delle maggiori piattaforme di blogging, b2/cafelog fa la sua prima comparsa nel 2001. b2 fu realizzato da Michel Valdrighi, che attualmente lavora per il team di sviluppo di WordPress. Nel 2003 il fork WordPress prese vita, grazie allo sforzo di Matt Mullenweg e Mike Little. Da allora il progetto iniziale b2/cafelog non fu più mantenuto e diventò un progetto morto. Sebbene WordPress divenne da subito il successore di b2, quest'ultimo continua ad essere supportato e mantenuto tutt'oggi grazie al fork b2evolution. Dal 2004 WordPress cominciò a vantare di importanti riconoscimenti. Nel dicembre del 2005 viene inaugurato il servizio WordPress.com che farà diventare

## 5. BLOG & WORDPRESS

---

WordPress una delle piattaforme blog più conosciute e più popolari. Questo servizio offre tutt'ora uno spazio web con piattaforma WordPress preinstallata sul quale è possibile realizzare un piccolo blog personale. Ovviamente, essendo lo spazio web gratuito, non è possibile effettuare l'upload di file.

Dal 2003 si susseguirono regolari rilasci; nel giugno del 2009 è stata rilasciata l'ultima versione nota, ossia la 2.8 che prende il nome in codice di *Baker*. È interessante notare che dalla versione 1.0 tutte le successive, eccetto una, prendono il nome di importanti figure legate al mondo del Jazz.

A causa del massiccio utilizzo di questa piattaforma è nata una conferenza internazionale chiamata *WordCamp*, in cui sviluppatori e utenti si possono confrontare, e proporre le proprie idee ai creatori di WordPress. I WordCamp non hanno sede fissa e vengono organizzati un po' ovunque per dare la possibilità a tutti di parteciparvi.

### 5.2 Funzionalità di WordPress

Come spiegato nel paragrafo precedente, WordPress si focalizza sulla semplicità d'uso e sull'immediatezza nell'apprendimento delle funzionalità offerte. Per garantire questi aspetti mette a disposizione un'ottima interfaccia grafica, sia su lato front end, facendo sì che le informazioni vengano riportate nella loro naturalezza, sia su lato back end, garantendo all'amministratore una facile gestione dei contenuti.

Innanzitutto, come descritto in 4.3, WordPress suddivide i propri utenti in categorie; nello specifico in: Amministratori, Editori, Autori, Collaboratori e Sottoscrittori. Questo fa sì che la piattaforma supporti una gestione multi-autore degli articoli. È quindi possibile che lo stesso articolo venga manipolato da più autori, anche nello stesso tempo. Solo recentemente WordPress ha introdotto un sistema chiamato *Post Revisions* in grado di tener traccia delle varie modifiche effettuate dagli autori. Tale funzione si dimostra indispensabile nei blog in cui compare una redazione, in caso di blog gestito da un'unica persona questa funzionalità perde completamente il suo valore. Oltre al Post Revision, WordPress offre molte altre funzionalità, rivolte soprattutto nell'ambito degli articoli/posts, in quanto in un blog l'elemento chiave è proprio l'articolo. Per facilitare la creazione di nuovi contenuti è stato scelto un editor testuale WYSIWYG (*What You See Is What You Get*), ossia un editor semplice, intuitivo e simile ad un word processor. Questo termine si contrappone totalmente a quello di WYSIWYM



(*What You See Is What You Mean*). La scrittura di un articolo tramite linguaggio HTML (mediante tag) è un chiaro esempio di quest'ultimo concetto.

Con gli strumenti forniti da WordPress possiamo anche creare bozze di articoli, o impostare la data di pubblicazione di un articolo per un dato giorno o ora. La funzionalità forse più innovativa presente in questa piattaforma è l'invio di articoli tramite email. Questa ci dà la possibilità di pubblicare un articolo anche se ci troviamo fisicamente lontani da un computer; basterà essere armati di un dispositivo elettronico abilitato per la connessione ad internet (uno smartphone, un palmare o un netbook) in modo da riuscire ad inviare un'email contenente il corpo dell'articolo. Sarà compito di WordPress controllare l'apposito account di posta elettronica e provvedere alla pubblicazione dell'articolo.

Tutti i WCMS, compreso WordPress, offrono una suddivisione degli articoli in apposite categorie create dall'amministratore e inerenti al tema trattato (fig. 5.3). Oltre alla suddivisione in categorie, ogni articolo fa uso di *tags*, definiti anche come *metadati* (fig. 5.4). I tags sono parole chiave che in sintesi definiscono l'argomento principale (*topic*) dell'articolo.

Categorie e tag sono molti importanti e garantiscono:

- al visitatore di trovare gli articoli che suscitano un certo interesse, facilitando quindi la ricerca dei vecchi articoli;
- un'ottima indicizzazione da parte dei motori di ricerca. Questa fase è molto importante, soprattutto durante la fase di nascita di un blog poiché permette di acquistare visibilità agli occhi della rete. Tutte le azioni finalizzate ad ottenere questo scopo vengono definite con il nome di SEO (*Search Engine Optimizator*).

La piattaforma concede che gli utenti o i visitatori possano dare un feedback all'autore in merito ad un post, utilizzando i commenti. Con WordPress non esiste limitazione sull'invio dei commenti, non è necessario essere registrati per inviarli, basterà solo identificarsi con un nome e un'email (a meno che l'amministratore non abbia disattivato i commenti). Per far fronte a possibili commenti di spam o non adeguati, WordPress utilizza algoritmi in grado di capire se il commento è lecito o no. Solitamente questa funzionalità viene introdotta utilizzando opportuni plugin. Se l'amministratore lo desidera, l'IP del mittente può essere bloccato permanentemente. Questa funzione creerà una blacklist di utenti i cui permessi sono stati limitati a causa di cattiva condotta.

## 5. BLOG & WORDPRESS

---

<input type="checkbox"/> Nome	Descrizione	<input type="checkbox"/> -- Software
<input type="checkbox"/> Badalis.it	Bacheca del sito, verranno riportati gli annunci importanti.	<input type="checkbox"/> Internet
<input type="checkbox"/> Fun! <small>Modifica   Modifica veloce   Cancella</small>	Raccolta di articoli divertenti	<b>News</b> <span style="float: right;">Ultime news.</span>
<input type="checkbox"/> Guide / Rece	Raccolta di articoli che comprendono guide, recensioni di un prodotto.	<input type="checkbox"/> Real life
<input type="checkbox"/> Hi-Tech	Articoli di carattere Hi-Tech.	<input type="checkbox"/> Senza categoria
<input type="checkbox"/> Informatica		<input type="checkbox"/> VideoBlog
<input type="checkbox"/> -- Hardware		<input type="checkbox"/> Vita Scolastica =D
<input type="checkbox"/> -- OS		

**Figura 5.3:** Esempio di alcune categorie in WordPress

### Tags

---

Android Apple Badalis.it Cars  
Chimica inorganica Codec Contact us  
Curiosità **Download** Ds Eventi  
Exploit Fun! Google Guide Hardware  
Hi-Tech Howto HTC Immagini  
Inglese Interfaccie Internet iPhone  
iPod Joomla Letteratura Lifehack  
Linux MacWorld Expo Mobile Musica  
**News** Nokia Opera OS Poker  
Programmazione Real life Recensioni  
Riviste e Libri School Sicurezza  
**Software** Tecnologia Tecnologie  
energetiche Tip&Trick Utility Video  
VideoBlog Vista Windows  
**Windows Mobile** Wordpress

**Figura 5.4:** Tags Cloud di WordPress

Dalla versione 2.7 di WordPress è stato introdotto un innovativo sistema di aggiornamento automatico, che permette anche ai meno esperti di mantenere aggiornata la propria piattaforma, i plugin e i template installati.

### 5.3 Trackback e Pingback

Come accennato poco fa, WordPress concede agli utenti di commentare gli articoli pubblicati sul blog. Molto più interessante invece è la possibilità di creare dei collegamenti o di segnalare questi articoli ad altre persone. Questo è possibile grazie a due strumenti chiamati rispettivamente *trackback* e *pingback*. I trackback hanno il compito di fornire un sistema di notifiche tra siti web. Il loro funzionamento è suddivisibile in quattro passi:

1. una persona A scrive un articolo sul proprio blog;
2. una persona B legge l'articolo di A e vuole scrivere un commento a riguardo, ma vuole anche che i suoi lettori possano leggere ciò che ha da dire e possano commentare sul suo blog;
3. B quindi scrive una notizia sul suo blog ed invia un trackback al blog della persona A;
4. A riceve il trackback, se questo viene accettato verrà mostrato sotto forma di commento all'articolo originale, ed esso conterrà un semplice link all'articolo scritto da B nel suo blog.

Sfortunatamente questo sistema non adotta nessun tipo di controllo sull'autenticità dei trackback e quindi molto spesso questo strumento è utilizzato dagli *spammers*<sup>1</sup> per generare trackback fasulli.

I pingback vengono spesso confusi con i trackback, in realtà essi hanno una funzionalità diversa, e si caratterizzano per:

---

<sup>1</sup>Individuo che genera grandi quantità di spam, ossia messaggi indesiderati che per la maggior parte fanno riferimento ad annunci pubblicitari. In questo caso lo spammer utilizza i commenti (pingback e trackback compresi) messi a disposizione da WordPress per diffondere messaggi del tutto fuori contesto dall'argomento generale dell'articolo

## 5. BLOG & WORDPRESS

---

- i pingback fanno uso di una tecnologia di comunicazione chiamata XML-RPC, mentre i trackback utilizzano HTTP POST;
- i pingback possono essere inviati automaticamente, mentre i trackback supportano un invio solamente manuale;
- i pingback non inviano contenuti.

Per avere una buona idea di cosa siano i pingback possiamo paragonarli a commenti remoti.

1. una persona A scrive un articolo nel suo blog;
2. una persona B scrive un articolo di risposta a quello di A e successivamente gli invia un pingback;
3. il blog di A riceve il pingback di B e automaticamente controlla che il pingback sia stato inviato dall'articolo di B.

Generalmente il pingback viene visualizzato come un semplice link sull'articolo della persona A che punta all'articolo della persona B. I pingback, rispetto ai trackback, hanno il vantaggio che la verifica del mittente ne renda più difficile la falsificazione.

### 5.4 SQL Injection

Per *SQL Injection* si intende un attacco (tecnica di hacking) ad un'applicazione, solitamente una web application, con lo scopo di eseguire query su di un database SQL, utilizzando variabili passate dall'utente senza averle precedentemente verificate. Questo problema non affligge solo il PHP, ma più in generale tutti i linguaggi rivolti al web e qualsiasi sia il database SQL utilizzato. La risoluzione di questa tipologia di vulnerabilità è relativamente semplice, tuttavia molte applicazioni ancora ne soffrono. Con questa tecnica si possono portare a termine attacchi come il login non autorizzato di un amministratore e di conseguenza l'accesso a qualsiasi zona riservata del sito, l'eliminazione o la modifica di record presenti nel database o infine la visualizzazione dei record legati alle informazioni personali degli utenti registrati.

Analizziamo ora come un attacco del genere possa essere portato a termine. Il pre-requisito necessario per effettuare l'attacco al database è che sia presente una vulnerabilità nei controlli dei dati ricevuti in input. Prendiamo in considerazione il seguente codice PHP che fa riferimento al form di inserimento dell'username e password:

---

**Codice 1** Esempio di form di login

---

```
<form action='check.php' method='POST'>
  Login: <input type='text' name='username' /><br>
  Password: <input type='password' name='pwd' /><br>
  <input type='submit' name='login' value='Login' />
</form>
```

---

Ora facciamo riferimento al codice presente nella pagina check.php che avrà il compito di elaborare i dati inseriti, stabilire una connessione ad un database e verificare l'autenticità dell'utente. Per semplicità verrà riportata solo la porzione di codice più significativa.

---

**Codice 2** Esempio di codice per la gestione di un login

---

```
<?php
...
$sql = "SELECT * FROM users WHERE username = '" . $_POST['user']
      . "' AND pwd = '" . $_POST['pwd'] . "'";
$query = mysql_query($sql, $db);

if (mysql_num_rows($query) != 0){
  ...
  $auth = 1;
}else{
  ...
  $auth = 0;
}
...
?>
```

---

## 5. BLOG & WORDPRESS

---

Nel caso di un input dati corretto questa porzione di codice eseguirà una query del tipo: `SELECT * FROM users WHERE username='Mario Rossi' AND pwd='mariorossi'`; autenticando l'utente Mario Rossi, se presente nel database, in caso contrario l'autorizzazione sarà negata e l'accesso alla parte protetta del sito non sarà consentita. Tutto questo avviene se effettivamente l'utente Mario Rossi utilizzi la form nel modo per la quale è stata progettata. Facciamo ora un'ipotesi che ci porterà in seguito a capire come funziona la tecnica del SQL Injection. Se l'utente al posto di inserire la stringa "mariorossi" nel campo password, inserisse "mario'rossi", l'apice presente nella stringa ingannerebbe l'interprete SQL, portandolo a credere che la sequenza di caratteri finisca dopo "mario", in più avanzerebbe nella query un `rossi'`, del tutto incomprensibile per il server. L'errore che si verificherà non comprometterà in alcun modo il nostro sistema di autenticazione. Il problema è che a questo punto, chiunque abbia una conoscenza base della sintassi SQL, può effettuare il passo successivo, mettendo a grave rischio la sicurezza sia del database sia del sito. In questo caso la tecnica del SQL Injection ci porterà ad effettuare un login non autorizzato non avendo a disposizione la password dell'utente inserito. Per ottenere questo un malintenzionato, utilizzando degli appositi apici, inserirà nella query un'ulteriore condizione WHERE che renda l'espressione sempre vera e faccia in modo che l'interprete SQL trovi sempre almeno un risultato per la query eseguita. Un semplice modo per ottenere questo è inserire nella form, Mario Rossi come nome utente e `' OR 1='1` come password, questi dati daranno origine ad una query del tipo: `SELECT * FROM users WHERE username='Mario Rossi' AND pwd=' ' OR 1='1'` che consentirà un'autenticazione valida per l'utente "Mario Rossi", in quanto la condizione `1=1` sarà sempre verificata.

Prevenire questi attacchi è relativamente semplice ed è sempre consigliabile utilizzare una delle tre soluzioni poste di seguito.

La prima è utilizzare la funzione `magic_quotes_gpc` presente nel file di configurazione `php.ini`. Questa aggiungerà dei caratteri di escape davanti a tutti i contenuti presenti nei cookie, in POST e in GET. Tuttavia questa soluzione non è la migliore poiché potrebbe causare incompatibilità con molte web application di terze parti ed inoltre un hacker con buone doti potrebbe trovare altre vie per iniettare comandi SQL.

Una seconda soluzione è quella di utilizzare `addslashes()` o `mysql_escape_string()` su tutte le variabili inserite in una query. Queste funzioni assicureranno che a tutti i valori passati saranno aggiunti caratteri di escape, come `\'`, rendendoli così inoffensivi.

L'ultima soluzione, la più complessa, ci garantirà il massimo livello di sicurezza e consiste nel controllare ogni variabile inserita in una query con un'*espressione regolare* (regular expression o regex) o un'altra funzione specifica. Se per esempio il dato di input è un numero intero, la soluzione migliore sarà effettuare un cast ad int di quel dato, e poi inserirlo nella query. Il casting ad int, ritornerà 0 se la variabile seguente non è formata da un campo numerico, l'utilizzo degli apici quindi diventa superfluo. Ecco il risultato:

---

**Codice 3** Cast ad int di una variabile per proteggere la query

---

```
$id = (int)$_POST['id'];  
$query = mysql_query("SELECT * FROM users WHERE id=$id", $db);
```

---

Nel caso invece di campi testuali o misti la miglior soluzione è l'adozione di un'espressione regolare che filtrerà la stringa di testo immessa, ritornando errore nel caso contenga caratteri non ammessi. Ecco un esempio:

---

**Codice 4** Utilizzo di una regex per proteggere la query

---

```
if (ereg("[A-Z0-9]+$", $stringa)){  
    $query = mysql_query("SELECT * FROM users WHERE  
        username=$stringa", $db);  
}else die('Stringa inserita non valida');
```

---

Le regex sono strumenti importantissimi messi a disposizione da PHP per verificare che ogni input corrisponda effettivamente al tipo di dato richiesto. Questi strumenti, utilizzati in maniera opportuna, garantiscono una sicurezza elevata.

### 5.4.1 Classe WPDB di WordPress

WordPress per proteggersi da attacchi di SQL Injection utilizza in primo luogo dei cast ad int, nel caso in cui vengano passati parametri di tipo numerico. In secondo luogo fa uso di una specifica classe denominata *WPDB* (WordPress DataBase class). Tale classe è presente nativamente in WordPress e fornisce un insieme di metodi per interagire con qualsiasi tabella presente nel database. Questi metodi non dovrebbero essere

## 5. BLOG & WORDPRESS

---

richiamanti direttamente, poiché WordPress fornisce una variabile globale `$wpdb`, contenente un'istanza di questa classe e già pronta per dialogare con il database. Vediamo un esempio di funzionamento.

---

**Codice 5** Una SELECT effettuata tramite il metodo query della classe wpdb

---

```
$wpdb->query("SELECT * FROM $wpdb->users WHERE ID = '1'");
```

---

Con questa query otterremo tutte le informazioni sull'utente con ID 1, quindi l'utente *admin*, l'amministratore della piattaforma. Con la stessa metodologia possiamo ottenere operazioni di INSERT, UPDATE, DELETE, ... Tuttavia il metodo "query" è preferibilmente indicato per effettuare operazioni di SELECT. Per inserire o modificare dati presenti su una tabella è preferibile utilizzare metodi specifici offerti da WPDB, come `$wpdb->insert` e `$wpdb->update`.

L'implementazione di questa classe ha dei risvolti positivi soprattutto nel campo della sicurezza, e più in particolare nella protezione delle query da possibili attacchi di SQL Injection (*Protect Queries Against SQL Injection Attacks*). La classe offre infatti un metodo essenziale a tal proposito: *prepare*. Tale metodo consente di costruire una "query sicura", in cui sono presenti caratteri di escape. Di conseguenza l'inserimento di apici in un qualsiasi campo di input non darà luogo ad un tentativo di SQL Injection. L'utilizzo di "prepare" è di estrema facilità ed è contemplato nelle linee guida per lo sviluppo di WordPress. Andiamo ora ad analizzarne il funzionamento.

---

**Codice 6** Proteggere una query con il metodo prepare

---

```
$authorName = "andrea-giacomin";
$searchString = "wordpress";
$query = "SELECT * FROM wp_posts WHERE post_author = (SELECT ID FROM
        wp_users WHERE user_nicename = %s) AND post_content LIKE '%(%s)%'";

$wpdb->query($wpdb->prepare($query, $authorName, $searchString));
```

---

Questo esempio "simula"<sup>1</sup> una form di ricerca in cui l'utente inserisce il nome dell'autore, che ha creato i post, sul quale si vuole effettuare la ricerca della parola chiave.

<sup>1</sup>In un vero form le variabili `$authorName` e `$searchString` conterranno i valori prelevati da POST o GET.



La particolarità del metodo *prepare* è di sostituire al posto dei segnaposto, all'interno della query (%s, %d)<sup>1</sup>, il valore delle variabili passate come parametri. Ovviamente *prepare* avrà cura di inserire i dovuti caratteri di escape. La sicurezza di questa tecnica sta proprio nell'inserire i segnaposto e sostituirli in seguito con valori sicuri per la query. Se questo aspetto non viene assicurato il metodo *prepare* perde di significato e il risultato di questo sarà una “query vulnerabile”.

---

<sup>1</sup>Una cosa simile avviene nel C con la funzione `printf`.

## 5. BLOG & WORDPRESS

---

## 6

# Struttura del database di Wordpress

*La più importante motivazione per il lavoro di ricerca che portò al modello relazione fu l'obbiettivo di fornire un evidente e chiaro limite tra gli aspetti logici e fisici della gestione di database. (Edgar Frank Codd)*

Quest'ultimo capitolo lo dedicheremo alla struttura del database di WordPress. Il database in questione non è per niente complesso e si articola su 10 tabelle con un paio di campi ciascuna; queste avranno in compito di gestire utenti, impostazioni generali, commenti, posts, links, ... Il database tuttavia è espandibile tramite plugin di terze parti che sfruttano la sua flessibilità per creare nuove tabelle, integrarsi ed offrire nuovi servizi. La gestione del database è effettuata tramite il RDBMS *MySql*.

WordPress offre numerose API per la manipolazione dei dati presenti sul database, quindi non è necessario conoscerne la struttura ed in seguito effettuare delle modifiche manualmente.

La figura 6.1 è una rappresentazione logica dell'intero database di WordPress creato durante un'installazione standard, quindi privo di plugin aggiuntivi. Di seguito verranno esaminate approfonditamente solo le tabelle più importanti, mentre alcune verranno solo accennate.

## 6. STRUTTURA DEL DATABASE DI WORDPRESS

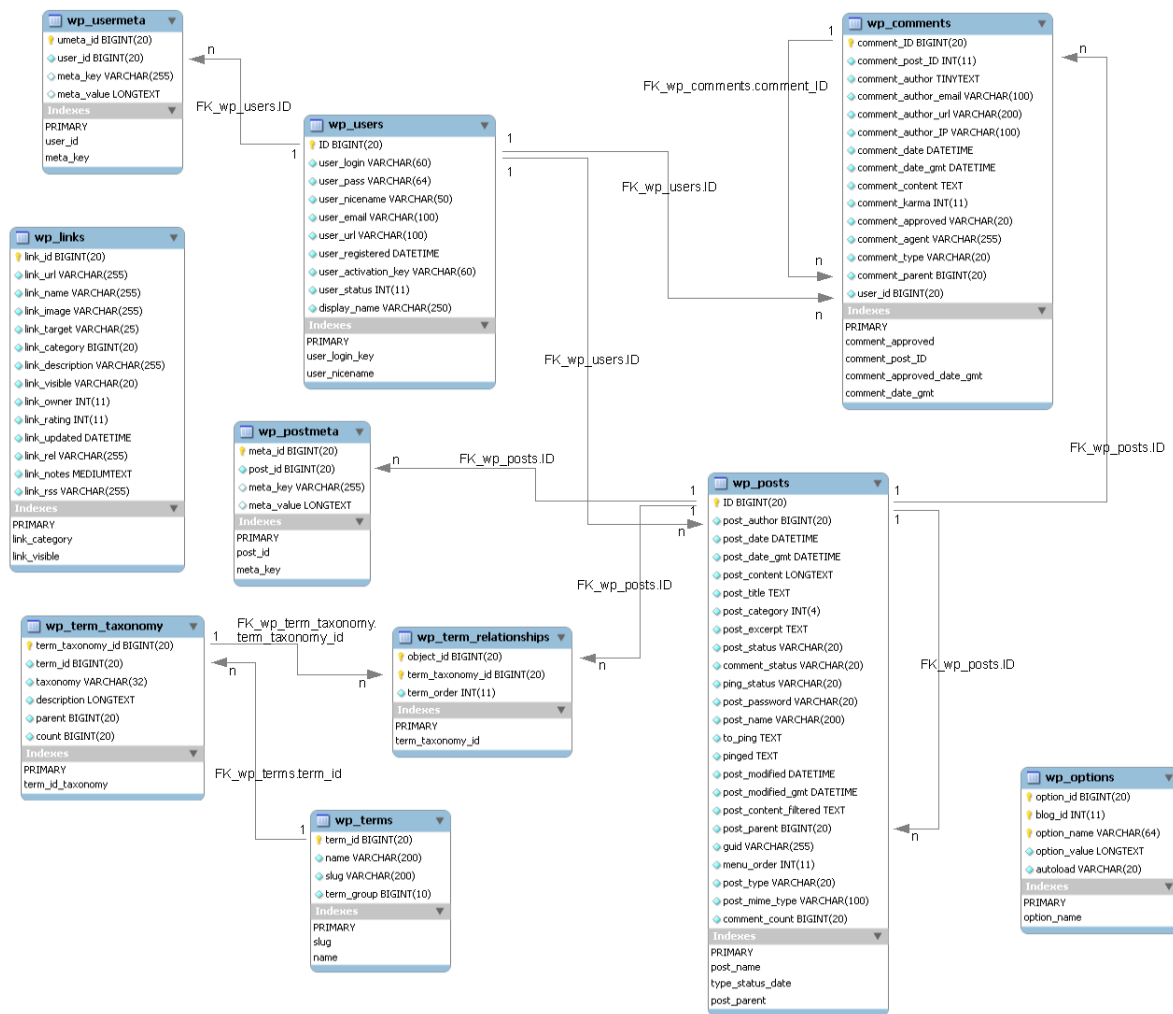


Figura 6.1: Struttura del database di WordPress

## 6.1 wp\_options

La tabella wp\_options contiene tutte le impostazioni globali del blog. I campi presenti in questa tabella sono:

Field	Type	Null	PrimaryKey	Default	Extra
option_id	bigint(20)		PRI Pt1	NULL	auto_increment
blog_id	int(11)		PRI Pt2	0	
option_name	varchar(64)		PRI Pt3 & IND		
option_value	longtext				
autoload	varchar(20)			Yes	

**Tabella 6.1:** Tabella wp\_options

La colonna blog\_id non è utilizzata in un’installazione standard di WordPress. La sua presenza è dovuta al fatto che la versione di WordPress MU (Multi-Users) ne fa uso.<sup>1</sup>

Questa tabella è una delle più consistenti in quanto conta circa un migliaio di record, tutti dedicati alle impostazioni generali della piattaforma. Inoltre, come si vede nella rappresentazione grafica del database, wp\_options non è collegata con nessun’altra tabella in quanto funge da “magazzino” per i vari settaggi. Una piccola parte del contenuto di questa tabella è presente nella figura 6.2.

## 6.2 wp\_posts

WordPress è un WCMS e quindi la sua funzione primaria è la gestione dei contenuti. La tabella wp\_posts ha in compito di tenere traccia di tutti i contenuti: articoli e qualsiasi genere di allegato (*attachments*).

Gli allegati per definizione sono dei file e quindi non sono adatti ad essere contenuti in un database, tuttavia nella tabella wp\_posts saranno presenti le informazioni riguardanti l’URL del file, l’id dell’utente che lo ha inviato, l’id del post in cui quel file verrà usato, la data d’invio ed altre informazioni secondarie. WordPress per differenziare i contenuti fra loro riserva un campo apposito, chiamato *post\_type* che può assumere tre valori distinti a seconda del tipo di contenuto:

<sup>1</sup>WordPress e WordPress MU condividono la stessa struttura del database

## 6. STRUTTURA DEL DATABASE DI WORDPRESS

---





























































	option_id	blog_id	option_name	option_value	autoload
 	1	0	siteurl	http://www.badalis.it	yes
 	2	0	blogname	Badalis	yes
 	3	0	blogdescription	Sapere Aude!	yes
 	4	0	users_can_register		yes
 	5	0	admin_email	andrea.giacomin@badalis.it	yes
 	6	0	start_of_week	1	yes
 	7	0	use_balanceTags		yes
 	8	0	use_smilies	1	yes
 	9	0	require_name_email	1	yes
 	10	0	comments_notify	1	yes
 	11	0	posts_per_rss	10	yes
 	12	0	rss_excerpt_length	50	yes
 	13	0	rss_use_excerpt	0	yes
 	14	0	mailserver_url	mail.example.com	yes
 	15	0	mailserver_login	login@example.com	yes
 	16	0	mailserver_pass	password	yes
 	17	0	mailserver_port	110	yes
 	18	0	default_category	9	yes
 	19	0	default_comment_status	open	yes
 	20	0	default_ping_status	open	yes
 	21	0	default_pingback_flag	1	yes
 	22	0	default_post_edit_rows	40	yes
 	23	0	posts_per_page	10	yes
 	24	0	what_to_show	posts	yes
 	25	0	date_format	j F Y	yes
 	26	0	time_format	H:i	yes
 	27	0	links_updated_date_format	j F Y H:i	yes
 	28	0	links_recently_updated_prepend	<em>	yes
 	29	0	links_recently_updated_append	</em>	yes
 	30	0	links_recently_updated_time	120	yes

Figura 6.2: wp\_options

- *post*: identifica un articolo;
- *page*: identifica una pagina<sup>1</sup>;
- *attachment*: identifica un allegato generico, una foto, un documento di testo, un file musicale, ...

I campi che formano questa tabella sono parecchi, tra i più importanti ricordiamo:

- *ID*: identifica univocamente un contenuto. L'id viene generato automaticamente dal DBMS;
- *post\_author*: questo campo fa da chiave esterna e contiene l'id dell'utente che ha scritto/inserito il contenuto;
- *post\_content*: contiene l'intero corpo dell'articolo (può essere in plaintext o in HTML). Se il contenuto è un allegato questo campo rimane vuoto ("");

---

<sup>1</sup>Le pagine sono differenti dai posts in quanto è possibile dividerle in sottopagine ed inoltre possono utilizzare un template diverso ed includere altro codice PHP.

- *post\_title*: contiene il titolo del contenuto;
- *post\_status*: identifica lo stato del contenuto: `publish` se il post è pubblicato, `draft` se il post è una bozza, `inherit` se si tratta di un contenuto, ...
- *post\_password*: contiene la password se il post è protetto. Il valore di default è "";
- *post\_parent*: contiene l'id del post in cui è presente il contenuto in questione (nei post questo campo assume il valore "");

### 6.3 wp\_comments

In questa tabella sono immagazzinati tutti i commenti scritti dagli utenti e dai visitatori. Oltre ai commenti compaiono i pingback e trackback (accennati nel paragrafo 5.3), i commenti in attesa di moderazione e i commenti di spam. Anche in questo caso la tabella è piuttosto complessa, analizziamo quindi solo i campi chiave.

- *comment\_ID*: identifica univocamente il commento ed è generato automaticamente da MySQL;
- *comment\_post\_ID*: chiave esterna che contiene l'id del post sul quale è stato effettuato il commento;
- *comment\_author*, *comment\_email* e *comment\_url*: contengono rispettivamente il nome, l'email e il sito web (facoltativo) di colui che ha scritto il commento;
- *comment\_author\_IP*: contiene l'IP dell'autore del commento. Tramite questo campo è possibile effettuare sospensioni degli utenti che non rispettano le regole di comportamento o che producono spam;
- *comment\_content*: contiene il contenuto del commento;
- *comment\_approved*: riporta il valore "1" se il commento è stato approvato da un editore o amministratore, "spam" se il commento è stato classificato come spam e "0" se il commento è in attesa di approvazione;
- *comment\_agent*: riporta la signature del browser con cui è stato inviato il commento (Firefox, Opera, MSIE, Safari, ...);

## 6. STRUTTURA DEL DATABASE DI WORDPRESS

---

- *comment\_type*: contiene le stringhe “comment”, “trackback” a seconda di che tipo di commento sia;
- *user\_id*: se l'autore del commento è un utente registro allora questo campo conterrà l'id dell'utente (chiave esterna alla tabella wp\_users).

### 6.4 wp\_users

La tabella wp\_users contiene la lista di tutti gli utenti iscritti al blog. Le specifiche della tabella sono:

Field	Type	Null	PrimaryKey	Default	Extra
ID	bigint(20) unsigned		PRI	NULL	auto_increment
user_login	varchar(60)		IND		
user_pass	varchar(64)				
user_nicename	varchar(50)		IND		
user_email	varchar(100)				
user_url	varchar(100)				
user_registered	datetime			0000-00-00 00:00:00	
user_activation_key	varchar(60)				
user_status	int(11)			0	
display_name	varchar(250)				

**Tabella 6.2:** Tabella wp\_users

L'id identifica univocamente un utente e perciò fa da chiave primaria, verrà poi usato come chiave esterna dalle tabelle wp\_comments, wp\_posts e wp\_usermeta. I campi *user\_login* e *user\_pass* sono utilizzati per contenere le credenziali di login. La password è criptata e non in chiaro (es: \$P\$BHewsgQU6KcOL.rw0BTKS0EsCb2J9B0). I campi successivi indicano l'email dell'utente, il sito web dell'utente nel caso ne possedesse uno, la data di registrazione, la chiave di attivazione (utilizzata durante la fase di registrazione) e il nickname da visualizzare. Il campo *user\_status* non è utilizzato nella versione standard di WordPress ed il suo valore di default è 0.



ID	user_login	user_pass	user_nicename	user_email	user_url	user_registered	user_activation_key	user_status	display_name
1	admin	\$P\$BoZF6oSxImrg5WHK4M HNUjgc7/rj	admin	andrea.giacomin@badalis.it	http://www.badalis.it	2008-12-23 17:42:33		0	admin
2	Andrea Giacomini	\$P\$BuUFY5qbtunMuZK3SEZSQPLVxaiPUi	andrea-giacomin	andrea.giacomin@badalis.it	http://www.badalis.it	2008-12-23 17:50:01		0	Andrea Giacomini
4	Francesco Colao	\$P\$BHeWagQURKCOI.nv0BTkS0EsCbZ9BD	francesco-colao	francesco.colao@badalis.it	http://www.badalis.it	2008-12-24 16:25:13		0	Francesco Colao
5	Alberto Donati	\$P\$BmiW5g4HfVEM8fZ3sqfG2mOdcV	alberto-donati	alberto.donati@badalis.it	http://www.badalis.it	2008-12-24 19:29:13		0	Alberto Donati

Figura 6.3: wp\_user

## 6.5 wp\_usermeta

La tabella `wp_usermeta` contiene informazioni supplementari per ogni utente registrato. In particolare conterrà informazioni riguardanti il nickname dell'utente, il colore della parte di back-end del sito, ma soprattutto i privilegi dell'utente. In questa tabella è definito il livello dell'utente (*wp\_user\_level*), il massimo livello è 10 e fa riferimento alla figura dell'amministratore. La struttura della tabella è la seguente:

Field	Type	Null	PrimaryKey	Default	Extra
<code>umeta_id</code>	<code>bigint(20)</code>		PRI	NULL	auto_increment
<code>user_id</code>	<code>bigint(20)</code>			0	FK->wp_users.ID
<code>meta_key</code>	<code>varchar(255)</code>	Yes	IND	NULL	
<code>meta_value</code>	<code>longtext</code>	Yes	IND	NULL	

Tabella 6.3: Tabella wp\_usermeta

La chiave primaria della tabella è un id (*umeta\_id*). Il campo *user\_id* è una chiave esterna e fa riferimento alla chiave primaria della tabella `wp_users`. I campi *meta\_key* e *meta\_value* contengono parole chiave (*nickname*, *admin\_color*, *wp\_user\_level*, ...) che verranno interpretate correttamente da WordPress.

## 6.6 wp\_postmeta

Questa tabella è molto simile a `wp_usermeta`, con l'unica differenza che questa conterrà informazioni supplementari su ogni post presente nel blog. La struttura è definita nella tabella 6.4.

Le *meta\_key* di questa tabella si prestano ad essere estese con l'installazione di nuovi plugin che offriranno nuove funzionalità.

## 6. STRUTTURA DEL DATABASE DI WORDPRESS

---

	umeta_id	user_id	meta_key	meta_value
	1	1	nickname	admin
	2	1	rich_editing	true
	3	1	comment_shortcuts	false
	4	1	admin_color	fresh
	5	1	wp_capabilities	a:1:{s:13:"administrator";b:1;}
	6	1	wp_user_level	10
	7	2	first_name	Andrea
	8	2	last_name	Giacomin
	9	2	nickname	Andrea Giacomin
	10	2	rich_editing	true
	11	2	comment_shortcuts	false
	12	2	admin_color	fresh
	13	2	wp_capabilities	a:1:{s:13:"administrator";b:1;}
	14	2	wp_user_level	10
	15	2	wp_autosave_draft_ids	a:38:{i:-1235415809;i:323;i:-1235490224;i:325;i:-1...
	29	4	admin_color	fresh
	28	4	comment_shortcuts	false
	27	4	rich_editing	true
	26	4	nickname	Francesco Colao
	25	4	last_name	Colao
	24	4	first_name	Francesco
	30	4	wp_capabilities	a:1:{s:6:"editor";b:1;}
	31	4	wp_user_level	7

**Figura 6.4:** wp\_usermeta

Field	Type	Null	PrimaryKey	Default	Extra
meta_id	bigint(20)		PRI	NULL	auto_increment
post_id	bigint(20)			0	FK->wp_posts.ID
meta_key	varchar(255)	Yes	IND	NULL	
meta_value	longtext	Yes		NULL	

**Tabella 6.4:** Tabella wp\_postmeta

## 6.7 wp\_terms, wp\_term\_relationships e wp\_term\_taxonomy

Queste tre tabelle hanno il compito di associare ad ogni post i relativi tags e categorie. Ogni post può appartenere a più categorie e può essere identificato da più tags. Si viene quindi a creare una relazione N a N.

Nella tabella wp\_terms sono presenti i nomi delle categorie e dei tag con relativi id e slug (abbreviazione del nome). In questa prima fase non c'è nessuna differenza tra tags e categorie. Gli id di questa tabella (ossia la chiave primaria) fa da chiave esterna per la tabella wp\_term\_taxonomy.

Field	Type	Null	PrimaryKey	Default	Extra
term_id	bigint(20)		PRI		auto_increment
name	varchar(200)				
slug	varchar(200)		UNI		
term_group	bigint(10)			0	

Tabella 6.5: Tabella wp\_terms

































	term_id	name	slug	term_group
 	1	Senza categoria	senza-categoria	0
 	2	Blogroll	blogroll	0
 	9	News	news	0
 	6	Contact us	contact-us	0
 	7	Badalis.it	badalisit	0
 	10	Hi-Tech	hi-tech	0
 	11	Guide / Rece	guide-rece	0
 	12	Real life	real-life	0
 	54	Informatica	informatica	0
 	14	OS	os	0
 	15	Software	software	0
 	16	Hardware	hardware	0
 	17	Programmazione	programmazione	0
 	18	VideoBlog	videoblog	0
 	19	Howto	howto	0
 	20	Guide	guide	0

Figura 6.5: wp\_terms

In wp\_term\_taxonomy avviene la suddivisione tra tag, categorie. La tabella 6.6 ne riporta la struttura. *term\_id* rappresenta l'id del tag o della categoria nella tabella wp\_terms. Il campo *taxonomy* contiene "post\_tag" o "category" a seconda se l'elemento è un tag o una categoria. In *description* è possibile inserire una breve descrizione sulla categoria o sul tag. Se nel blog abbiamo utilizzato una suddivisione in sottocategorie il campo *parent* conterrà l'id della categoria madre. Il campo *count* tiene traccia di quanti articolo hanno usato il tag o fanno parte delle categoria in questione.

## 6. STRUTTURA DEL DATABASE DI WORDPRESS

---

Field	Type	Null	PrimaryKey	Default	Extra
term_taxonomy_id	bigint(20)		PRI		auto_increment
term_id	bigint(20)		UNI Pt1	0	FK->wp_terms. term_id
taxonomy	varchar(32)		UNI Pt2		
description	longtext				
parent	bigint(20)			0	
count	bigint(20)			0	

**Tabella 6.6:** Tabella wp\_term\_taxonomy

	term_taxonomy_id	term_id	taxonomy	description	parent	count
	1	1	category		0	0
	2	2	link_category		0	7
	9	9	category	Ultime news.	0	51
	6	6	post_tag		0	1
	7	7	post_tag		0	11
	10	10	category	Articoli di carattere Hi-Tech.	0	4
	11	11	category	Raccolta di articoli che comprendono guide, recens...	0	17
	12	12	category		0	4
	65	54	category		0	0
	14	14	category		54	2
	15	15	category		54	19
	16	16	category		54	0
	17	17	category		54	1
	18	18	category		0	10
	19	7	category	Bachecca del sito, verranno riportati gli annunci im...	0	9
	20	19	post_tag		0	14

**Figura 6.6:** wp\_term\_taxonomy

I post sono associati alle categorie e ai tags dalla tabella wp\_terms e questa associazione è mantenuta in wp\_term\_relationships.

Field	Type	Null	PrimaryKey	Default	Extra
object_id	bigint(20)		PRI Pt1	0	
term_taxonomy_id	bigint(20)		PRI Pt2 & IND	0	FK->wp_term_taxonomy.term_taxonomy_id
term_order	int(11)			0	

**Tabella 6.7:** Tabella wp\_term\_relationships

	object_id	term_taxonomy_id	term_order
	1	2	0
	2	2	0
	3	2	0
	4	2	0
	5	2	0
	6	2	0
	7	2	0
	24	19	0
	2	1	0
	19	1	0
	19	6	0
	24	7	0
	36	19	0
	36	7	0
	41	18	0
	41	31	0

**Figura 6.7:** wp\_term\_relationships

## 6.8 wp\_links

La tabella wp\_links memorizza i link che verranno poi visualizzati sulla sezione destra del blog. Solitamente i link fanno riferimento a siti “partner”, tra i quali c’è un buon scambio di informazioni.

## 6. STRUTTURA DEL DATABASE DI WORDPRESS

---

## 7

# Conclusioni

Nel corso della tesina sono stati analizzati molti aspetti, che spaziano da brevi cenni storici sulla comunicazione per arrivare poi alla nascita del Web e alle novità che esso ha introdotto. Originalmente lo scopo della tesina era quello di spiegare il significato di blog e come questo venga utilizzato oggi giorno per diffondere “informazioni”. Durante la stesura il campo di argomenti trattati si è poi allargato, includendo delle note sul contetto di CMS e il relativo funzionamento (in via del tutto teorica). Da quest’ultimo aspetto sono poi giunto ad una breve analisi della piattaforma WordPress, soffermandomi brevemente sull’aspetto della sicurezza e sulle funzionalità che quest’ultima offre. L’ultimo passo è stato effettuare l’analisi della struttura del database di WordPress, passando in rassegna le più importanti tabelle che lo costituiscono ed evidenziandone i punti salienti.

## 7. CONCLUSIONI

---



# Glossario

<b>Admin</b>	È l'abbreviazione del termine <i>Administrator</i> , ed identifica l'amministratore di una rete o di una piattaforma., 27
<b>Blog</b>	Il blog è un sito internet in cui l'autore (o gli autori) scrive a riguardo dei propri interessi, pensieri, opinioni. Generalmente un blog è molto più flessibile rispetto ad un portale., 31
<b>CERN</b>	o <i>Conseil européen pour la recherche nucléaire</i> , è il più grande laboratorio al mondo di fisica delle particelle. Si trova al confine tra Francia e Svizzera., 13
<b>CMS</b>	La sigla indica un <i>Content Management System</i> , ossia un sistema in grado di gestire qualsiasi tipo di contenuti digitali., 21
<b>Database</b>	I database sono insiemi di dati, suddivisi in ordine logico, in modo da permetterne una gestione da parte di software appropriati: i DBMS., 45
<b>DBMS</b>	Acronimo di <i>Database Management System</i> , un sistema software in grado di manipolare/inserire/interrogare i dati presenti in un database., 24
<b>Disco di Nipkow</b>	È un disco metallico che ha lo scopo di analizzare e riprodurre le immagini., 9

<b>GPL</b>	Con questo termine si fa riferimento alla <i>General Public License</i> o <i>GNU GPL</i> , una licenza software (documento legale) che concede al licenziatario il permesso di modificare, copiare e ridistribuire il software gratuitamente o a pagamento., 32
<b>Internet</b>	Internet rappresenta una rete di computer mondiale ad accesso pubblico., 10
<b>Mashup</b>	Tecnica utilizzata allo scopo di unire dinamicamente informazioni provenienti da più sorgenti., 19
<b>MySQL</b>	È uno degli RDBMS più famosi ed utilizzati. Viene distribuito sotto licenza GNU GLP e viene sviluppato da <i>Sun Microsystems.</i> , 45
<b>Plugin</b>	Un plugin (o addon) è un programma che si appoggia su un altro per estenderne le funzionalità., 13
<b>Post</b>	In generale in tutti i siti, con post si intende un messaggio (formato anche da immagini e altri elementi) inserito da un individuo autorizzato., 32
<b>RDBMS</b>	Sta ad indicare un DMBS basato sul modello relazione. La teoria del modello relazione fu stata stilata da <i>Edgar F. Codd.</i> , 45
<b>Template</b>	Il template (o modello) è una pagina che ne definisce solo il layout grafico, ed in seguito verrà completato con le informazioni da visualizzare., 24
<b>TeX</b>	Il TeX è un linguaggio tipografico ideato da <i>Donald Knuth</i> , utilizzato per scrivere testi scientifici, tesi e libri. La pronuncia “tex” è errata mentre quella corretta è “tech” ., iii
<b>W3C</b>	Il World Wide Web Consortium è un’associazione fondata da Tim B. Lee allo scopo di mantenere, sviluppare e regolare il Web, 13
<b>WCMS</b>	Un WCMS è un CMS per la gestione di contenuti online., 22

<b>Web</b>	Con Web si intende l'insieme di tutti i contenuti e servizi offerti agli utenti che usufruiscono della rete Internet., 13
<b>Wordpress</b>	Wordpress è un WCMS atto alla creazione e al mantenimento di blog., 31

## Glossario

---

# Bibliografia

- [1] **SQL-Injection**. Documento Online stilato da CreativeWeb.
- [2] **Storia della comunicazione**. Articolo online. <http://www.aiutoscuola.it/storia-della-comunicazione.html>.
- [3] **Wikipedia**. Enciclopedia online. Voci: Blog, CGI, CMS, Comunicazione, Meshup, Storia della televisione, Telegrafo, Web 2.0, Wordpress, World Wide Web.
- [4] **WordPress Docs**. Documentazione ufficiale WordPress.
- [5] CERIOLI ALDO. *CMS Open Source: siti web alla portata di tutti. Il Fenomeno Joomla!* Master's thesis, Università di Roma, 2008.
- [6] SIMONIN DANIELE. **Il Web 2.0**. Articolo online. <http://projects.melodycode.com/Web20/>.
- [7] ZERILLO DARIO. **Mashup con PHP**. Articolo online. <http://php.html.it/articoli/leggi/2463/mashup-con-php/>.
- [8] CASALI DAVIDE. **IWordCamp: Wordpress as CMS**. Articolo online. <http://im.digitalhymn.com/2008/05/11/iwordcamp-wordpress-as-cms/>.
- [9] COPES FLAVIO. *Un CMS per gestire un APT*. Master's thesis, Politecnico di Milano, 2008.
- [10] DE LUNA GIOVANNI, MARIGGI MARCO, AND TARPINO ANTONELLA. *Codice Storia - volume terzo*. paravia, 2000.
- [11] PRATI GIULIANO. *Web 2.0: Internet è cambiato*. UNI Service, 2007.

## BIBLIOGRAFIA

---

- [12] SAMBUGAR MARTA AND SALÀ GABRIELLA. *GAOT2 - Generi Autori Opere Temi*. La Nuova Italia, 2004.
- [13] PELIZZONE MAURIZIO. **Evitare SQL Injection con Wordpress**. Articolo online. <http://maurizio.mavida.com/2009/01/25/evitare-sql-injection-con-wordpress/>.
- [14] BAGNATICA STEFANO. *CMS Open Source: Joomla*. Master's thesis, Università Cattolica del Sacro Cuore, 2008.