

Chapter 1: First Post

Какво разглеждаме в настоящата Глава?

- Фактори, обуславящи развитието на платформата WordPress
- Избор на подходяща платформа за инсталиране на WordPress
- Изтегляне, инсталиране и базово конфигуриране на WordPress
- Диагностициране и разрешаване на често срещани проблеми при инсталиране

Ако сметем, че с извеждане на съобщението: “Hello World” на дадено устройство демонстрираме минимално познаване на даден език за програмиране, то същото в публицистичните кръгове предполага да публикуваме първия си пост. В Главата описваме накратко генезиса на WordPress и представяме опции за хостване на WordPress инсталация. Изброените често срещани грешки и недоразумения и поместени съвети за разрешаването им ни дават практически насоки как умело да демонстрираме вещина Мрежата.

След, като инсталирали, конфигурираме WordPress и организираме базовата администрация на Системата, можем да се възползваме от интерактивно презентирания код и подробно анализираните в последващите Глави компоненти. Ако пък вече имаме работещ WordPress Сайт, можем да прескочим текущата Глава и да пристъпим непосредствено към Глава 2: „Представяне на Кода“.

Съдържание

Какво разглеждаме в настоящата Глава?	1
Що е WordPress?	3
WordPress е всепризнат	4
Текущ статус	4
Взаимодействие с Общността ползватели	6
WordPress и GPL (General Public License – Общ Публичен лиценз)	6
Какво съдържание публикуваме и как общуваме	8
WordPress в ролята на Content Management System	8
Как да завържем разговор?	9
Откъде да започнем?	9
Hosting варианти	10
Инсталация „Направи си сам“	11
Инсталиране на WordPress Файловете	12
Конфигуриране на Базата-данни	17
ЗАКЛЮЧИТЕЛНИ ПРОЦЕДУРИ	21
Първоначално администриране	21
Първи Post	23
ОБОБЩЕНИЕ	24

Що е WordPress?

WordPress е сред най-широко застъпените днешни Open Source Content Management Systems с глобално представени реални аудитория, развойни специалисти и групи за поддръжка. При все и съпоставима с TypePad, Moveable Type, Google's Blogger и Apache Roller project в качеството си на потребителски генерирана развойна среда, WordPress се характеризира с широко разнообразие от хостинг опции, функционални модули (plugins) и естетически дизайн и елементи (theme-и).

С разпространението на собствени публикации, на нискотарифен Web хостинг и свободен базов софтуер от типа на MySQL СУБД, Blogging софтуерът последва тренда при повечето други дигитални технологии на пренасочване от скъпи продукти от висок клас, към широко достъпни, ниско тарифни потребителски или „любителски“ системи. WordPress не е замислен с цел просто да създадем Blog, така че да разполагаме с цифрово тефтерче с календар към претенциозния ни URL, а вече е пълноценна Content Management System както за индивидуални, така и за корпоративни потребители. Текущата Глава накратко ни запознава със зараждането на WordPress и ни представя подобренията в новата Версия, както и широката база ползватели.

WordPress се заражда по подобие на множество други популярни Open Source софтуерни пакети: Няколко талантиливи софтуерни разработчици предугаждат потенциал в създаването на мощно, просто средство въз основа на вече съществуващ Проект с Отворен/Общ публичен лиценз ([GPL](#) и/или [GNU General Public License](#)). Изхождайки от Системата [b2/cafeblog](#) на Michel Valdrighi, системните аналитици Matt Mullenweg и Mike Little я доразвиват и надграждат. Оповестената през 2003 WordPress се базира на MySQL Open Source СУБД за персистентно съдържание и PHP развойна платформа. Valdrighi продължава с приноса си за процъфтяващия с множеството си заинтригувани привърженици и широка база развойни специалисти проект с Общност от все повече потребители и проектанти.

Подобно на други Системи, писани на PHP, и WordPress е автономна с това, че всички задания по инсталиране, конфигуриране и администриране, са заложили в PHP-модули. Популярността на WordPress донякъде произтича от простотата, залежала във фразата „петминутна инсталация“, която срещаме в почти всяко описание или книга за WordPress. WordPress е не просто възможност да публикуваме в Мрежата, а предоставя заложили мащабируемост и адаптивност към всякакви ситуации и потребители.

WordPress понастоящем се поддържа от неколцина чисто развойни специалисти и множество ключови поддръжници. Mike Little се грижи за специализирания комерсиален Сайт [zed1.com](#) на WordPress и подпомага ситуационните промени в Код. Компанията Automattic на Matt Mullenweg продължава да се грижи за Хостинга на [wordpress.com](#), както и подпомага добавянето на релевантно съдържание и нови опции, от типа на Akismet, WordPress за множество Сайтове и Gravatar. [Akismet](#) е мощна, хоствана от Automattic, услуга за защита и детектор на spam с пренебрежимо малка статистическа грешка. Доскоро известна като [WordPress MU](#), многосайтовата функционалност на WordPress е в основите на [wordpress.com](#) хоситнговата система и понастоящем е вградена в главното Source Tree на основния WordPress. [Gravatar](#) обслужва динамично e-mail адреси със съответно изображение, предоставяйки множество варианти на показване на конкретна икона. Представете си го като услуга за динамично представяне на профила ни в технически и социален план.

В ролята си на Content Management System, WordPress не се изразява единствено в поредица от постове и коментари. [BuddyPress](#) е съвкупност от theme-и и plugin-и, които надграждат WordPress до платформа за функционална социална комуникация по Мрежата на регистрирани потребители с възможност да общуват и взаимодействат помежду си, като цялото съдържание и тук отново се управлява от WordPress. По същия начин, [bbPress](#) е PHP- и MySQL-базирана система за организиране на Форуми (Bulletin Boards), различна от WordPress, но пълноценно интегрирана с нея.

В Глава 16, „Общност на разработчиците на WordPress“, са разгледани в повече детайли някои от съпровождащите WordPress системи, но това правим с цел да разясним защо WordPress е еволюирала отвъд пределите на базово, Single-User ориентирано, средство. При това, авторите не черпят дивиденди и не правят опити да популяризират Automattic, но да разглеждаме в детайли WordPress, без да споменем Mullenweg и Little, е почти непостижимо и граничи с недобрите социални порядки.

WordPress е всепризнат

В настоящата книга се базираме на WordPress 3.5. Всяка последваща Версия на WordPress включва подобрения в администрирането и контролните функции (Контролен екран/Dashboard); функции за backup, експорт и импорт; възможности за инсталиране и upgrade. Дори и да започнем с по-стара версия на WordPress, с лекота я актуализираме до текуща такава и така винаги разполагаме с последна версия. Вариантите за инсталиране и upgrade са представени по-нататък в тази Глава. Доколко популярен обаче е WordPress?

Текущ статус

Интересът към WordPress и възможностите му за използване ескалира. В ръцете си държим доказателство на такова становище. Допреди само 3 години, на пазара присъстваха само няколко книги по въпроса. Сегашното издание търпи вече втора публикация. Понятието „популярност“ неминуемо има субективни измерения, но можем да му придадем и статистическа обосновка. Automattic съобщават, че считано към 2011г., ежедневно се регистрират над 100,000 нови Сайта на WordPress (<https://wordpress.com/stats/>). Сред тях са такива с Content Management на WordPress, Blogging-инструменти и такива за Personal Rants (Лични Дъски за съобщения); измежду тях следва да приспадне потребителите с повече от една свои WordPress инсталации, но дори и при подобна умерена оценка, WordPress е изключително популярен. Automattic оповестява приблизително 74 милиона WordPress Web-Сайта в глобален мащаб, като приблизително половината са host-вани от WordPress.com (<https://wordpress.com/stats/>). При предишното издание на тази книга, Сайтовете бяха едва 5 милиона. През 2008г., официалното Хранилище за plugin-и на WordPress съдържа над 6,300 plugin-а, двойно повече от броя им през 2007г. Към момента на написване на настоящата книга, броят на plugin-ите вече надхвърля 19,000 (<https://wordpress.org/news/2012/05/plugins-refreshed/>). Той касае официалното Хранилище за theme-и на WordPress и не включва всички vendor-и на комерсиални theme-и, както и независимите разработчици на собствени theme-и.

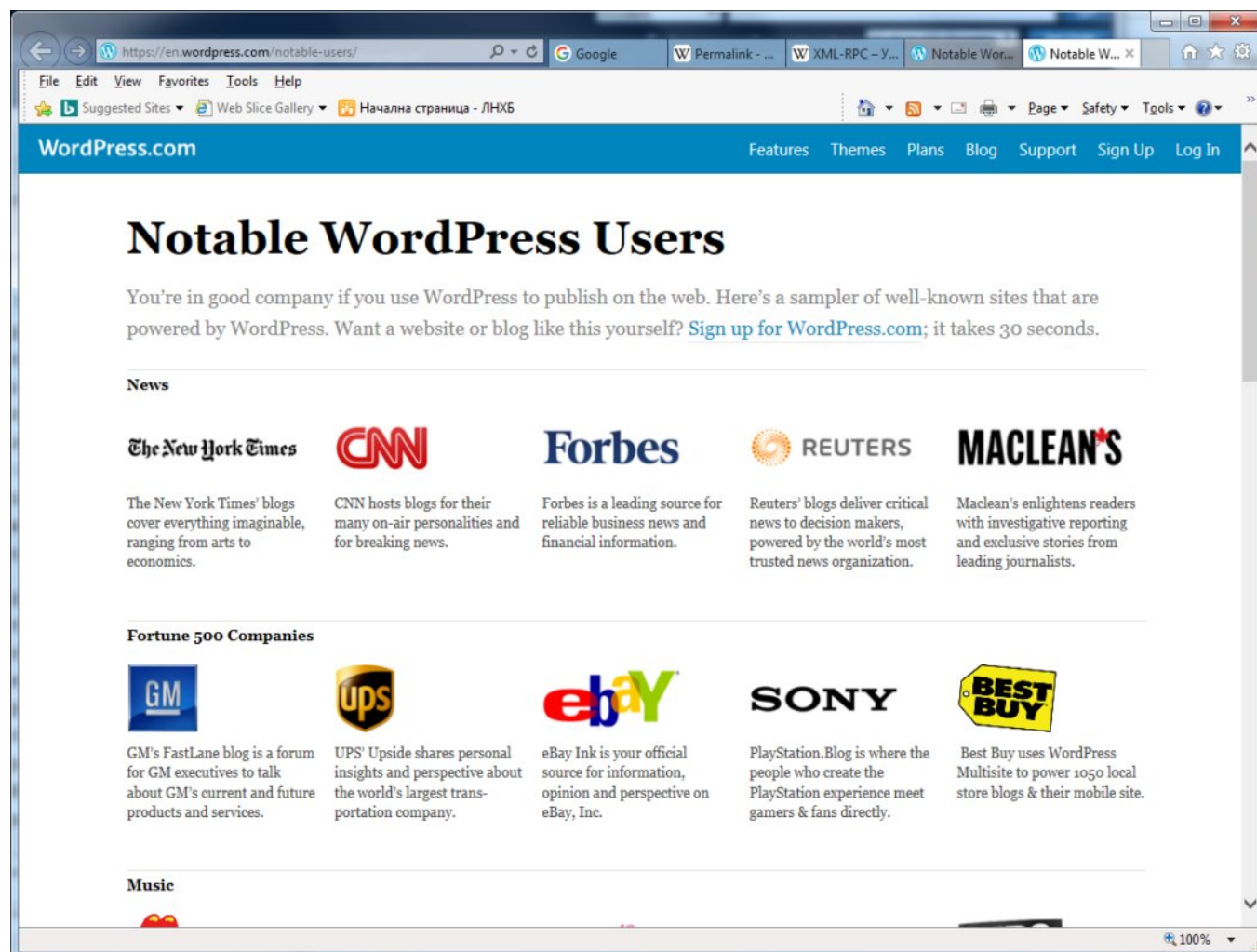
Комбинациите от plugin-и и theme-и изискват научен подход за детайлно представяне, но паралелно, всички те еднакво лесно се идентифицират, интегрират и използват в следствие от стройна архитектура и добре организирана Общност ползватели. Накратко, екосистемата, обкръжаваща WordPress, е жизнена и процъфтяваща.

През Август 2011г., Matt Mullenweg представи текущото ниво на използване на WordPress, наред с резултатите от първо изследване на WordPress в San Francisco WordCamp въобще. Проучването на WordPress наподобява общо преброяване на WordPress ползвателите, проследяващо ежедневното използване на WordPress. Включва репрезентативна извадка ползватели на WordPress, в т.ч. независими Web Developer-и, корпоративни клиенти и любители. Долната статистика илюстрира доминиращия дял на WordPress в глобалната Мрежа:

- Приблизително 15 процента от най-често посещаваните 1 милион Уеб-сайта използват WordPress.
- Като цяло, 22 от всеки 100 нови Уеб-сайта, използват WordPress.
- Над 200 милиона plugin-и са били изтеглени от Plugin-хранилището.
- В проучването във връзка със 170,000 Web-сайта, взеха участие 18,000 потребители.

[Презентацията](#) на Mullenweg за развитието Word може да бъде гледана по WordPress.tv.

Понастоящем, WordPress се използва отчасти или изцяло в Web-сайтовете на големи медийни компании, сред които: [блоговете](#) на CNN, [All Things D](#) на Wall Street Journal, Reuters, Forbes, както и в недотам уместния, но чест неформален пример с [icanhazcheeseburger.com](#). (Ако търсите аргументирана обосновка на понятието “snowclone”, знайте, че е свързано и с насладата, носена от пикантни подробности от кухнята на замесената медия.) WordPress използват Компании сред 500-те водещи в класацията на Fortune, като: GM, UPS и Sony. WordPress е обоснован избор за множество ползватели: от интернационални конгломерати, през ключови фигури в звукозаписа, до големи медийни компании. Някои намират аргументирана обосновка на избора на WordPress в списъка с големи пазарни участници – ползватели на WordPress; такъв online-списък с реални Видни ползватели е изложен на съответната Страница (<https://en.wordpress.com/notable-users/>):



Простотата и удобството на употреба, както и не на последно място, мощните plugin-и и theme-и, правят WordPress пригоден както за Web-сайта със семейна информация на майка ни, така и за Дъската с обявления на учителя в съседното начално училище или за целите на любители в дадена област. Начините на приложение на WordPress го правят текущо толкова успешен, а широката му популярност произтича от множеството широко достъпни, по-специализирани сайтове.

WordPress притежава заложената адаптивност с цел да бъде прост или комплексен, съобразно нуждите ни. Предоставяните на “lower tech” (по-ретроградни) потребители възможности да публикуват в Мрежата и словото им (Word - играта на думи е умишлена) да разгласява пред семейства и приятелски кръгове колко лесен за използване е WordPress, подхранват експлозивните му популяризиране и възприемане.

Върху какво стъпваме? wordpress.org е извор на настоящи и тепърва разработвани версии. На wordpress.org/extend имаме на разположение plugin-и и theme-и, както и класифицирани идеи и предстоящи нововъведения.

На wordpress.com присъстват както безплатни, така и с платени hosting Услуги. На адрес wordpress.org/hosting намираме изброени hosting provider-и, поддържащи WordPress, чиито hosting-Услуги често включват и допълнителен първоначален support при инсталиране и настройки.

Взаимодействие с Общността ползватели

WordPress се развива и процъфтява не само благодарени на редови потребители, но и посредством приноса на Общността ползватели. Всичко се прави, както при гимназиалния спортен отбор, в името на играта – с участие в неофициално регламентирани мероприятия, които канализират усилията ни и където се ползваме от подкрепа.

WordCamp мероприятията се организират от Общността и провеждат на местно ниво в множество градове по света. Официалните WordCamps (WordЛагери) са изброени на wordcamp.org, но още по-удачно е да се огледаме за WordCamp събития в най-близкия голям град. WordЛагери се провеждат почти всеки уийкенд и в тях участват blogger-и, фотографи, писатели, редактори, развойни специалисти и дизайнери с всякаква квалификация и опит. WordЛагерите са стопански обоснован начин да се представим пред местната Общност и удачен шанс да срещнем WordPress знаменитости. На wordcamp.org е публикувана информация за следващия WordCamp.

Не така добре организирани, но по-често провеждани от WordЛагерите, са WordPress Meetups (WordPress Събиранията) на местни потребители и развойни специалисти от почти 200 населени места (сравни с броя им от 40, споменат в предишното издание на книгата). За това е нужен meetup.com account, но регистрираните потребители могат да направят справка за местоположение и график на провеждане на wordpress.meetup.com, откъдето да разберат кога и къде участници беседват на тема content management.

В Хранилището, host-вано на codex.wordpress.org, се съдържа богата Документация на множество езици. Достопочтеният Codex на WordPress, наименован с термина за древни ръкописни пергаменти, съдържа журнализирани от Общността тънкости и трикове във всеки аспект на WordPress, от инсталиране до debugging (отстраняване на грешки). Ако желаем да обогатим WordPress документацията, трябва да се регистрираме и да изложим това, което ни интересува или мъчи, във WordPress Codex. Надяваме се въпросната книга да бъде за вас както компаньон, така и водач към нашия Codex.

Включили сме и множество списъци (и съответни архиви) с адреси за кореспонденция на множество сътрудници и Общности на WordPress. Актуален такъв е поместен на адрес [codex.wordpress.org/Mailing Lists](http://codex.wordpress.org/Mailing_Lists); особено актуални са може би Списъкът wp-docs на Codex-сътрудниците, както и този wp-hackers на проектантите на ядрото, които очертават насоките за развитие на WordPress.

WordPress и GPL (General Public License – Общ Публичен лиценз)

WordPress е лицензирана под [GNU General Public License \(GPL\) version 2](http://www.gnu.org/licenses/gpl-2.0.html), упоменат в `license.txt` файла, съдържащ се на най-горно ниво в дистрибутирания пакет с код. Болшинството ползватели не четат Лицензионното споразумение и възприемат WordPress като стандартен проект с отворен код; при това, обаче, ред корпоративни правни отдели не спират да таят притеснения във връзка със зловредни компоненти на GPL лиценза по отношение на код и съдържание, добавяни към, използвани паралелно с или надграждащи първоначалния пакет с код. Това объркване до голяма степен произтича от самоволното използване на понятията „безплатен“ (“free”) и „авторско право“ (“copyright”) в контекста на некоректно поставени ситуации.

Авторите на настоящата книга не са юристи — нито пък претендират да са такива в сферата на Internet или телевизията — и затова, ако се интересувате от детайли на Закона за Авторското право и какво именно е плагиатство и „заимстване на код“, запознайте се с [някой](#) от трудовете на [Lawrence Lessig](#) или [Cory Doctorow](#) по въпроса. Намерението ни тук е да сведем до минимум безпокойствата на IT-Отделите, свързани с използването на Корпоративна Content Management System от типа на недотам зорко пазената от екипи юристи WordPress. Спрете с подобни притеснения; след, като WordPress устройва [CNN](#) и [Wall Street Journal](#): две компании, процъфтяващи, предоставяйки Авторските права върху съдържанието си, навярно би удовлетворявала правните рестрикции и на повечето днешни корпоративни потребители.

В замисъла на GPL е залегнала презумпцията да можем, когато пожелаем, да изтеглим source code-а на което и да е издание на софтуер с GPL-лиценз. Когато Компания направи промени в софтуерен пакет с GPL-лиценз и публикува новата Версия, тя трябва да направи достъпен и съответния source code. Това е „зловредната“ страна на GPL в действие; целта е да гарантираме достъпа до първоначалната и всяка последваща Версия на софтуера. Ако възнамеряваме да променяме ядрото на WordPress и да разпространим кода, трябва да сме сигурни, че промените ни се вменят в GPL и че кодът е на разположение под формата на source code. И доколкото WordPress е написан на PHP: скриптов език, дистрибуцията на софтуера на практика е равностойна на дистрибутиране на source code-а.

Следват разпространени погрешни схващания и съответни разяснения относно използването на WordPress в пазарни условия.

- **“Free software” не може да бъде използван за стопански цели.** Имате право да предоставяте срещу заплащане на трети лица вашата WordPress платформа, да отдавате рекламна площ в сайта си, както и да използвате WordPress Content Management платформата като база за online-търговия. Така работи [wordpress.com](#); същото позволява на [Google](#) да отдават срещу заплащане Linux-базирани услуги. Можем да намерим специални WordPress-предложения на theme-и с професионално качество, както и да плащаме на hosting provider стотици или дори хиляди долари годишно, за да host-ва софтуерните ни пакети (при положение, че и в двата случая трансферираме WordPress на комерсиална основа).
- **Ако променим Кода, така че да обслужва специфични наши {тип съдържание, политика на сигурност, неясни изисквания за навигация}, ще трябва да публикуваме съответните промени.** Ще трябва само да направим достъпен Кода на Софтуера, разпространяван от нас. Ако решим да осъществим тези промени в рамките на нашата Компания, не е нужно впоследствие да ги извеждаме извън нея. От друга страна, чиито и да е промени в ядрото на WordPress повлияват цялата Общност. Да накараш по-консервативните служители да проумеят значението на колективния принос и да разхлабят регламента по отношение на Авторски права и принос на служителите често е възтрудно, но нашият солиден резон би бил основание и други работодатели са достигнат до същите заключения в полза на глобална WordPress общност.
- **Общият Публичен лиценз GPL ще “навреди на безопасността” на нашите данни във WordPress.** Данните — включително графични елементи от theme-и, post-ове и Страници, управлявани от WordPress — са независими от WordPress ядрото. Управлява ги софтуер, необвързан с тях. Theme-ите, обаче, са производни на WordPress Кода и, следователно, попадат под GPL-изискванията да направим достъпен съответния Source code. Можем да предоставяме theme-а срещу заплащане в случай, че направим същата достъпна на пазара. Важно с случая е Source Code-ът да стане достъпен за всеки ползвател на Software-а. Ако ще предоставяме срещу заплащане дадена theme-а, паралелно трябва да направим Source Code-а достъпен

съгласно GPL; както вече отбелязахме, потребителите – ползватели на theme-ата получават фактически и Source Code-a.

Далеч по-важно от историческия опит на WordPress и лицензионната проверка е това за какво ще ни послужи WordPress и какво ще спечелим от всеобхватната Система. В следващата Секция разглеждаме WordPress не просто като инструмент за редактиране на blog-ове, а в ролята на пълноценна Content Management System.

Какво съдържание публикуваме и как общуваме

Много линейни метри по рафтовете в книжарниците са отрупани с книги, учещи ни на това как да пишем по-добре, на висок стил, на blogging-техники и прочие аспекти на изкуството да създаваме Съдържание (content). Една от целите на тази книга е да дефинираме Визуалните (visual), Стилистични (stylistic) и Context Management (Управляващи контекста) механизми, които WordPress ще изгради у нас с оглед на това Съдържанието ни да формира ангажирани потребителски общности. Въпросният контекст провокира разговор с посетителите ни. Не визираме единствено думите във всеки post, нито дали и доколко текстовете ни са ангажиращи. Как ни възприемат хората? Какво ни отличава от всички останали? Каква е характерната визия на нашия Сайт и отговаря ли на поставената цел: персонална, корпоративна, социална или комерсиална?

WordPress в ролята на Content Management System

Първоизточник на Blogging системите са рутинни Content Management дейности: създаваме post, помещаваме го в надеждна система за съхранение и извеждане на данни от типа на Файлова система (File-System) или База-данни (Data-Base) и визуализираме резултати, форматиращи съобразно текущи или базови критерии. Обогащават се многообразието и типовете съдържание, представяни в страниците с blog-ове и изискванията при сортиране, търсене, избиране и представяне, при което извежданата информация все по-често касае метаданни и класифицирано съдържание, което на своя страна размива границата между т.нар. 'vanilla' (неподлежащ на индивидуални донастройки) blogging- софтуер за един-единствен потребител и Content Management системите на корпоративно ниво.

Content Management системите (CMS) обслужват създаването, съхранението, извеждането, описанието или анотацията и извеждането или представянето на множество типове съдържание. CMS покриват цели етапи на работния процес, най-вече такива в редакторската сфера и в публицистиката, но често касаят дейностите при подбор и маркиране на съдържание с цел последващо редактиране или ретроспекция. Контролният екран WordPress Dashboard предоставя такива елементи на Управление на съдържанието (workflow management) и Редакторски контрол (editorial control). WordPress не е единствената широко използвана Open Source Content Management System в наши дни; проектите Drupal и Joomla са не по-малко популярни. Drupal и Joomla са в ролята на доставчици на поддържани Хранилища на съдържание; обслужват множество типове съдържание, множество автори в множество роли и предоставят съдържанието въз основа заявките на потребители. WordPress е blogging система по своята същност и акцентува върху процеса по представяне съдържание на посетителя. Позволява интеграция и с други Content Management системи дори и когато функциите им отчасти се припокриват. Процесът на интегриране е разгледан обстойно в Глава 14.

WordPress се е утвърдила като същинска/удачна (*bona fide*) Content Management System с дизайна си с оглед мащабируемост и с гъвкаво представяне на съдържанието, невлияещо по никакъв начин надеждното съхранение на първични данни в неизменен вид. Гъвкавият шаблон за представяне Model-View-Controller на WordPress е независим от персистентната логика на MySQL данновия модел, от тематично организирания Потребителски интерфейс, от функциите за визуализиране и plugin-архитектурата, обезпечаваща функционално представяне на данновия

поток. Основен плюс на WordPress е съхранението на данните във вида, в който са въведени от потребителят или са постъпили от дадено приложение посредством съответни Програмни интерфейси (APIs) на WordPress. Съдържанието не се форматира, преобразува с template-и или аранжира до момента на зареждане на страницата, когато това правят изключително мощните функции, генериращи същинския HTML. Данновият модел на WordPress прави това, обръщайки се към богата колекция от таблици, с цел класифициране въз основа категории (taxonomies), етикети на съдържанието (folksonomies), данни за автора, коментари и прочие валидни референтни критерии. Това става възможно благодарени на структурата на Базата-данни (database schema) на WordPress, разгледана детайлно в Глава 6.

Разгледаният дизайн придава огромна мощност и гъвкавост на WordPress в ролята на Content Management System, което от своя страна изисква да познаваме влиянието ѝ върху съответните персистентни данни и контролни потоци (подобна дисекцията на WordPress във функционален план ни подтикна да напишем настоящата книга).

Как да завържем разговор?

“Най-важен е разговорът помежду ни; темата е просто нещо, което обсъждаме.”

— Cory Doctorow

Надеждността на CMS намира израз в целепригодността на нейното съдържание. Дори и най-многообразните типове съдържание и най-стриктно управляваните процеси не са от полза, ако никой не се възползва от тях. Нещата не се заключават в това да инсталираме blogging софтуер, да напишем няколко post-а и да чакаме хората да се наредят на опашка пред виртуалната ни врата; трябва да заложим, по думите на Tim O'Reilly, “архитектура на приобщаване”. Социалните мрежи, рекламата, поощренията и стъпки по посока на това Сайтът ни да бъде показван от Търсачките ще привличат посетители; дизайнът, брандингът и графичните елементи, в добавка към качество на съдържанието, ще ги стимулират да участват активно.

От гледна точка на посетителя, проблемът изглежда така: при наличието на десетки милиони Web-сайтове (на много от които, освен “първа публикация” /“first post”, няма кой знае какво), кой ще ви види, чуе, или ще ви отговори? Последователите ни в Twitter трябва да имат стимул да посетят Сайта ни, а WordPress Сайтът ни увеличава броя на Туитовете. От своя страна, промени в Twitter можем да показваме в sidebar-а на WordPress, придавайки колорит на общата, старателно премислена хронология, с помощта на актуално към момента съдържание. Ако сме активни във Facebook, бихме могли да поместваме броя посещения на Страница с видни личности ([Public Figure Pages](#)), като така Facebook-участниците, от своя страна, ще насочат трафик обратно към Web-сайта ни. Ако съдържанието ни касае специфични, обстойни или мистериозни сфери, съответни търсения в Google ще насочват посетителите директно към нас, при което те ще се включат в разговора. В Глава 11, „Агрегиране на съдържанието“ (“Content Aggregation”), разглеждаме как включваме във WordPress съдържание от Социални медии (Social Media) и други Content системи, а Глава 12, „Проектиране на Потребителско изживяване“ (“Crafting a User Experience”), изследва как още повече да популяризираме нашето WordPress-съдържание.

Откъде да започнем?

Преди да се захващаме сериозно с презентиране, стил или съдържание, трябва да обмислим къде да поместим Web-сайта си (независимо от вече обсъдените WordPress и Content Management System-и, ще насочим внимание към собствения ни Web-сайт и реалната WordPress-инсталация в качество на негова фактическа реализация/противовес най-вече с цел удобство и за краткост). Сред факторите в обосновка на нашия избор, са:

- **Цена** — Безплатните hosting-услуги ограничават полето за развойна дейност и често изключват възможности за предоставяне на комерсиални рекламни услуги. Комерсиалните

варианти често включват по-добър support, повече пространство за съхранение и по-широка честотна лента, както и Бази-данни с множество инстанции за допълнителни приложения.

- **Контрол** — С какви Система за управление на MySQL База-данни, инсталационен пакет на WordPress и друго нужно разполагаме? Ако възнамеряваме да ровичкаме на ниво SQL посредством MySQL Command Prompt, ще трябва да си осигурим Hosting provider, предоставящ достъп до съответните интерфейси.
- **Сложност** — Можем самостоятелно да инсталираме Apache Web Server с PHP Interpreter, MySQL и WordPress пакета, но повечето Hosting provider-и предоставят като услуга съответния процес, с което ни спестяват подобни затруднения. Във връзка с нужната техническа помощ по отношение на платформата на конкретната Операционна система, следва да си осигурим provider (или да разчитаме на собственият си IT-Отдел) с цел да имаме на разположение Support в разумен срок.

В настоящата Глава правим преглед на възможни Hosting варианти, даваме кратки насоки за самостоятелна инсталация и инспектираме възможни колизни ситуации, при които инсталациите на WordPress и MySQL взаимно се изключват (в случай на противоречие).

Hosting варианти

Различаваме три главни категории WordPress hosting, характеризиращи се с баланс Административна сложност/Ниво на контрол. Най-удобно и разпространено е използването на wordpress.com - безплатна hosting-услуга на Automattic, базирана на многосайтовия вариант на WordPress (първоначално известен като: WordPress MU). Инсталираме theme-и и plugin-и през Контролния екран (Dashboard-a), но можем единствено да активираме/деактивираме налични, априорно заложили опции. Нямаме достъп до съответните MySQL Бази-данни и Базов код (Core code), и не можем да интегрираме WordPress с други Системи. Можем да пренасочим някой от собствените си URL-и към wordpress.com, но при нужда от пълен контрол над всичко: от Кода, до използваните URL-и, вероятно бихме преминали към платен вариант. Безплатният вариант е разумен като начало с уговорката, че рано или късно ще трябва да пристъпим към радикална трансформация на цялата наша инсталация.

На www.wordpress.org ще намерим първоначален списък с hosting provider-и, включително и платените варианти на wordpress.com. Повечето включват последни или текущо актуални Версии на WordPress ядрото, налични под формата на инсталационен пакет, изискващ паралелно инсталиране на MySQL и Web server. Трета hosting опция е да инсталираме всичко на собствени, управлявани от нас Server-и. Ако Server-ите ни са част от чуждо техническо средство за hosting, но имаме root administrative access, в случая отново е нужна наша собствена инсталация. WordPress предполага Web-сървър с PHP-поддръжка, средство за пренаписване на URL-адреси и инсталирна MySQL. Apache е най-разпространения вариант за front-ending WordPress, защото предоставя `mod_php` в ролята на PHP-Интерпретатор и `mod_rewrite` в ролята на средство за пренаписване на URL-адреси. Все по-често се използва `lighttpd` (Lighty) вместо Apache въпреки, че при пренаписването на URL-адреси, от време на време трябва да го „водим за ръка“. В ролята на Web server можем да използваме и модула `URL_rewrite` на Microsoft's IIS 7.0. Акцентът върху пренаписването на URL-адреси произтича от WordPress поддръжката на “прилични” входни [permalinks](#) за blog-ове, позволяващи да създадем URL-дърво съобразно дата, категория, tag и пр. метаданни. Такива мнемонични (разбираеми за хората) URL-адреси се конвертират в Заявки към MySQL Базата-данни за извеждане на съответно WordPress-съдържание съобразно Заглавия или други критерии, заложили в Основния цикъл (Main Loop) на WordPress, представен в детайли в Глава 5. Web-сървърът ни решава дали URL-адресът следва да бъде преобразуван (parsed) от WordPress или касае даден HTML-файл въз основа на информацията, залежала във файла `.htaccess`, като правилата за пренаписване на URL-адреса гарантират правилното му интерпретиране. В технически план, пренаписването на URL-адреси не е задължителна предпоставка за инсталиране на WordPress, но е желателно с оглед огромната гъвкавост,

предоставяна от Конвенциите за именуване и презентиране (Presentation and Naming Conventions) и заложена при формиране на URL-адресите на съдържанието. При подготвяне съдържанието на нашия WordPress, следва да се придържаме към заложените в [Permalink](#) идеи и Добрите практики (разгледани в повече детайли в Глава 2).

Дотук споменавахме MySQL с недомлъвки, но познаването на изискванията на MySQL е сред ключовите hosting-предпоставки. Добре е да познаваме базовата терминология и да разграничаваме MySQL софтуера, Текущи състояния на Базата-данни (database instances) и WordPress-конфигурации (instances), използващи MySQL. След инсталиране и конфигуриране на MySQL, разполагаме с функционална, пълноценна релационна СУБД. Не е обезателно тя да бъде конфигурирана на машината, където е нашият Web-сървър и затова някои hosting provider-и паралелно с Потребителските интерфейси на своя Web-сървър, създават хоризонтално мащабируеми (horizontally scalable) MySQL „ферми“. Конкретна *установка (instance)* на MySQL на даден сървър поддържа множество *бази-данни (databases)* с уникални имена. При инсталирането на WordPress, ще трябва да укажем *името (name)* на MySQL Базата-данни, резервирано за нашето съдържание, която информация принципно се генерира автоматично и конфигурира автономно (в случай, че доставчикът ни поддържа WordPress и MySQL като интегриран пакет). WordPress създава съответни даннови *таблици (tables)* към въпросните именувани Бази-данни във връзка с всеки създаден от нас Web-сайт.

Номенклатурата на база заданието и комплексният му характер могат да породят объркване. Ако ние (или hosting provider-ът ни) работим с множество установки на MySQL на различни server-и, ще искаме да знаем къде се host-ва Базата ни данни. Доколкото всяка установка на MySQL поддържа множество Бази-данни, а те съдържат групи таблици, бихме могли да използваме няколко MySQL-базирани приложения на една-единствена hosting-платформа посредством една MySQL-установка или дори една-единствена MySQL База-данни.

Ако имаме множество WordPress-сайтове на един и същи server, но желаем да поддържаме една-единствена поделена помежду им установка на MySQL База-данни, следва да укажем на WordPress да разграничава *имената на Таблиците в MySQL Базата-данни (MySQL database table names)* в рамките на MySQL Базата-данни. Подобен рутинен вариант сме изложили в следващата Глава с акцент върху различията между множество колекции от Таблици в Базата-данни и множество Бази-данни на различни Приложения.

Веднъж извършили базови настройки, време е да пуснем Кода да работи. Дори и в случай, че hosting provider-ът ни предоставя готови, работещи MySQL и WordPress, добре би било да знаем как server-side компонентите взаимодействат помежду си в случай, че ни се наложи да се оправяме с проблем, тъкмо когато сме ангажирани изцяло с разработката на plugin-и.

Инсталация „Направи си сам“

Прословутата, легендарна, станала нарицателна, петминутна инсталация на WordPress е възможна тогава, когато нещата са правилно конфигурирани и координирани. Тук проследяваме стъпки, които обикновено ни убягват, когато provider-ът ни предоставя пакетите инсталирани и наблюдаваме на чести разминавания между WordPress и MySQL инициализациите (instances).

Процесът на инсталиране е прост и удобен (при вече изрядно работещи Web- и MySQL-сървъри): изтегляме WordPress пакета, инсталираме го в Дървото с директории (Directory Tree) на нашия Web-сървър, навигираме до нашия собствен Домейн-адрес (top-level URL) и изпълняваме процес, който можем да изкажем с едно сложно съставно изречение.

Препоръчително и резонно е да инсталираме пълноценна WordPress конфигурация на мобилен или настолен компютър най-вече ако ще се занимаваме с работа над ядрото, с разработка на plugin-и, т.е. ако тестването на публичен Web-сайт касае промени с вероятни смущаващи последствия. Mac OS X има априорно заложен Apache Web Server (с PHP и Преобразуване на URL-

адресите); трябва да изтеглим MySQL от www.mysql.com или да използваме готов пакет, от типа на MAMP (www.mamp.info с включен phpMyAdmin инструмент), с цел да разполагаме с автономна (self-contained) развойна платформа. По отношение на други Операционни системи, на XAMPP (www.apachefriends.org) се съдържа елегантно интегриран платформен stack, интегриран с Windows, Mac OS и Linux Операционни системи. Това да разполагаме с всичко под един покрив, е великолепна възможност за предпазване от възможни грешки, както ще видим с следващите две Глави. Локалната работа с WordPress разглеждаме по-детайлно в Глава 3.

Инсталиране на WordPress Файловете

Ако изтеглим WordPress кода от wordpress.org, ще разполагаме със zip (или tarball) архив, който разархивираме в папка с име `wordpress`. Първи етап на WordPress инсталацията е да поместим Кода в Дървото с папки на нашия Web-сървър; от ключово значение е разполагането му на правилното място. Ако не следваме изложените тук напътствия, Адресът на Web-сайта ни може да се превърне в нещо от типа на: <http://example.com/wordpress> и ще се наложи да правим всичко отначало, или пък ще разпращаме на близки и приятели причудливи URL-адреси. Ако пък желаем да разграничим WordPress съдържанието от останалото съдържание на своя Web-сайт или да разделим множество Секции, изборът на организация на Файловата система (Filesystem Layout) е не по-малко важен.

Тръгваме от най-горната директория, където инсталираме WordPress. Обикновено това е Главната директория на нашия Web-сървър, а ако използваме Hosting Provider, това най-често е поддиректорията `public_html` в Дървото с файлове. Ако използваме Инсталационен пакет с Меню за указване локация за инсталиране, трябва да изберем директорията на най-горно ниво (визираме факта, че такава гарантирано присъства!); ако копираме файлове от локална машина на Web-сървър с помощта на FTP-клиент, следва да изберем правилна дестинация. Интуитивната процедура да копираме zip-файла на Сървъра и впоследствие да го разархивираме, насочва всичко към поддиректорията `wordpress`, така че ако държим URL-адресът на WordPress-сайта ни да бъде <http://example.com/>, а не: <http://example.com/wordpress>, ще трябва да преместим файловете с едно ниво нагоре в Дървото с директории, преди да продължим. Възможна е конфигурационна опция да поместим WordPress-инсталацията в поддиректория на нашия Домейн-адрес (top-level URL), което прави разполагане на нашия WordPress на място с не най-добрата възможна география на Файловата система (filesystem geography) недотам критично. Как правим всичко това, ще видим в края на тази Глава.

Вече инсталираните файлове на WordPress би следвало да изглеждат в Браузъра на Файловата ни система по начин, подобен на показания на Figure 1-1, с файл `index.php` и шаблонен (template-) файл `wp-config-sample.php`. Това е WordPress системата в нейната цялост и във вида, в който работи успешно под PHP Интерпретатора на Web-сървъра.

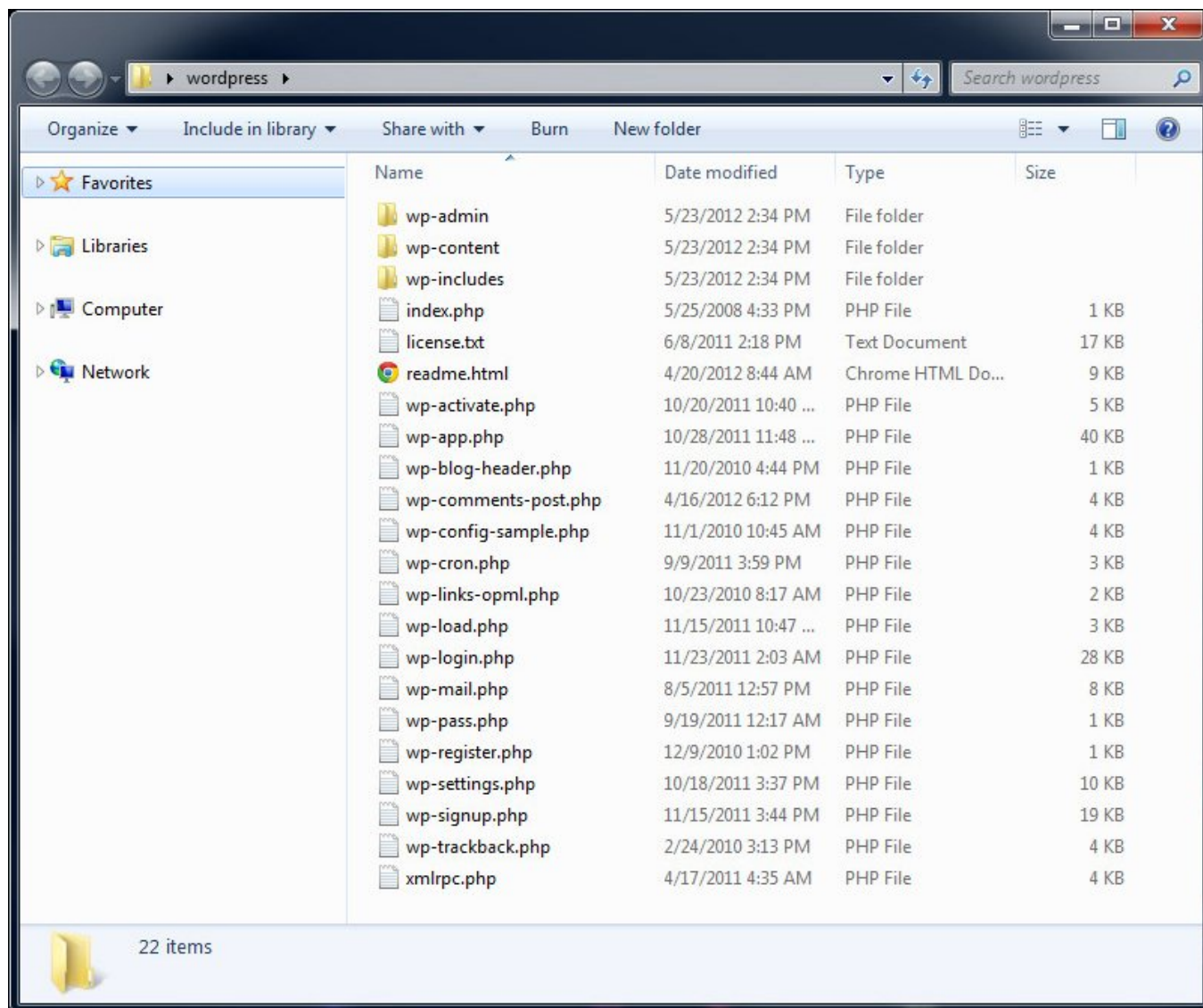


FIGURE 1-1: Чиста, неконфигурирана, WordPress-инсталация

На този етап, в случай, че правим manual installation, ще трябва да създадем собствен `wp-config.php` файл, редактирайки готовия (sample) такъв и съхранявайки го в най-горната директория на нашия WordPress. Ако навигираме до URL-адреса на нашия Web-сайт, WordPress Кодът ще установи отсъствието на Конфигурационен файл (configuration file) и ще изведе Диалогови прозорци, сходни с показаните на Figure 1-2 и 1-3, на които следва да се укажат допълнителни данни. Изискват се името на нашата MySQL База-данни, Име на нейния User (ползвател) и конкретен Префикс на Таблицата на WordPress базата-данни (различен от подразбиращия се `wp_`). Въпросните детайли на ниско ниво са залегнали в основата на следващата Глава относно Конфигуриране на Бази-данни (database configuration). Ако ползваме услугите на hosting provider и готов Инсталационен пакет (packaged installations), процесът вероятно ще протече невидимо за нас, защото WordPress-файловете ще бъдат разархивирани и информацията от MySQL Базата-данни ще постъпи автоматично в Конфигурационния файл (configuration file), без нужда от потребителска намеса.



FIGURE 1-2: WordPress ще създаде нов wp-config файл, при отсъствие на такъв.

A screenshot of the WordPress database configuration dialog box. At the top is the WordPress logo. Below it, a message says: "Below you should enter your database connection details. If you're not sure about these, contact your host." There are five input fields with labels and descriptions: "Database Name" (wp_halstern) with description "The name of the database you want to run WP in.", "User Name" (wp_user1) with description "Your MySQL username", "Password" (foobar) with description "...and your MySQL password.", "Database Host" (localhost) with description "You should be able to get this info from your web host, if localhost does not work.", and "Table Prefix" (wp_hs) with description "If you want to run multiple WordPress installations in a single database, change this." At the bottom left is a "Submit" button.

FIGURE 1-3: Диалогов конфигурационен прозорец на Базата-данни (Database configuration dialog box)

Какво следва да направим, ако на указания URL -адрес вече има HTML и/или друго съдържание и искаме да интегрираме WordPress към вече съществуващ Сайт? Кой от налични файлове следва да изтрием, зависи от това каква е била ролята на досегашната навигация до URL-адреса ни. За да използваме WordPress като Content Management System, което разискваме тук, най-добре е да запазим текущото съдържание, като го конвертираме в нови Постове или Страници, с чиито съдържание и колорит успешно да допълним новия, управляван от WordPress, Сайт. Или пък бихме могли да инсталираме WordPress в поддиректория, запазвайки досегашния `index.html` файл, като пренасочваме посетителите към новото съдържание с бутон или link на досегашната ни Начална страница. Нека не разчитаме на Провидението; ако инсталираме WordPress при вече съществуващ наш `index.html` файл, ще имаме паралелно `index.php` и `index.html` файлове,

като това кой ще видят посетителите ще зависи от Directory Index конфигурацията на Web-сървъра, host-ващ Сайта ни. При работа върху текущото съдържание, следва да се ръководим от трафика към Сайта ни, породен от него: ако Страниците ни генерират трафик за Търсещите машини, следва не да премахваме текущите, вече cache-ирани URL-адреси, а да инсталираме WordPress в поддиректория. Ако обаче предпочитаме да въздействаме на потребителя главно посредством организацията и възможностите на WordPress, ще трябва да преместим досегашното съдържание, да преобразуваме URL-адресите и/или те да пренасочват към света на WordPress.

Ако ползваме Инсталационен пакет на hosting provider или сами сме създали `wp-config.php`-файл, когато навигираме до Домейн-адреса си, би трябвало да ни бъдат изведени вече създадени от WordPress Таблици на Базата-данни, Потребител-администратор (administrative user) на нашия WordPress, и първоначална Парола, като показаните на Фигура 1-4. Резонно е да променим Потребителското си име на стринг, различен от 'admin'.



Welcome

Welcome to the famous five minute WordPress installation process! You may want to browse the [ReadMe documentation](#) at your leisure. Otherwise, just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Don't worry, you can always change these settings later.

Site Title

Username

Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods and the @ symbol.

Password, twice

A password will be automatically generated for you if you leave this blank.

Strong

Hint: The password should be at least seven characters long. To make it stronger, use upper and lower case letters, numbers and symbols like ! " ? \$ % ^ &).

Your E-mail

Double-check your email address before continuing.

Privacy

☒ Allow my site to appear in search engines like Google and Technorati.

Install WordPress

FIGURE 1-4: Въведете данни за Web-сайта и укажете Потребител-администратор

При успешна инсталация, би следвало да ни бъде изведен Диалогов прозорец, като този на Фигура 1-5, индикиращ, че знаменитата 5-минутна инсталация е приключила успешно.



Success!

WordPress has been installed. Were you expecting more steps? Sorry to disappoint.

Username hal_stern

Password Your chosen password.

Log In

FIGURE 1-5: Административна информация в края на чиста инсталация (clean install)

В следващия Раздел разглеждаме процеса по конфигуриране на MySQL-WordPress в повече детайли и препоръчваме да я прегледате дори и в случай, че самият термин SQL ви лази по нервите. Ако пък процесът ви е познат, преминете направо към Глава „Заклучителни процедури“.

Конфигуриране на Базата-данни

Ако hosting provider-ът ви е подготвил MySQL База-данни и е създал ваш Потребителски account, прегледайте съответните данни във вашия `wp-config.php` файл. Ще са ви нужни при изпитването на MySQL, изложено в настоящия Раздел и с оглед възможни проблеми с MySQL тепърва, е препоръчително да разполагате с тях. Файлът съдържа комбинация от Потребителско име и Парола и затова гледайте на него като на конвенционална login информация. Ако пък се наложи да навлезете в дебрите на do-it-yourself Инсталация, Разделът ще ви спести евентуални бъдещи обърквания и затруднения при поддръждане частите на пъзела.

Идеята е инсталацията и конфигурирането на MySQL за WordPress да са тривиални: след запуск на MySQL, създаваме WordPress Потребител в MySQL, след което от името на същия този Потребител, създаваме База-данни: колекция от Таблиците на WordPress. Можем да използваме MySQL command line или инструменти от типа на: phpMyAdmin или Chive при подобни задания имайки предвид, че MySQL поддържа собствен набор потребители със съответни правомощия, различен от този на нашата (или на hosting provider-а ни) Операционна система. Веднъж инсталиран, MySQL създава Таблица по подразбиране с Потребители и Правомощия, прибавяйки потребител `root` към тези на Unix-система, който е MySQL superuser, различен от Unix root user-a. Ако все пак опитаме да работим с MySQL-конфигурацията като MySQL root user, ще можем да я достъпваме само локално – от localhost (от машината, на която работи MySQL). Повече за MySQL правомощията, за Таблиците, които ги предоставят на потребителите и как се администрират MySQL-потребителите, можем да разберем от “MySQL Reference Manual” (<http://dev.mysql.com/doc/>) от Главите относно безопасността на първоначалните MySQL account-и.

Няма установени конвенции за именуване на WordPress Потребители или Бази-данни; hosting provider-ите обичайно добавят името на Пакета или информация за account-а с цел идентификация на потребителите, използващи паралелно MySQL База-данни. Припомняме, че

често множество Бази-данни биват притежавани от един потребител или различни MySQL-потребители използват един и същ [Server Instance](#) на конкретна MySQL База-данни. В примера на Фигура 1-3, `wp_` е префиксът и за Потребителско име, и за име на База-данни, най-малкото с цел да подсказва на Администратора на Базата-данни, че и двете касаят инсталация на WordPress.

Какви разминавания са възможни между WordPress и MySQL? По-долу излагаме трите възможни базови предпоставки за проблеми с инсталацията. Обръщаме внимание, че разглеждаме евентуална проява на някоя измежду изброените причини по време на инсталация; приемаме, че опитваме да се `log-нем` като Потребител с Администраторски правомощия във вече създадена първична структура на База-данни със съответен Потребител `admin`.

- 1. Web server can't find MySQL** (Web сървърът не открива MySQL). Въвели сме `host-име` (`hostname`) на MySQL server, различно от упоменатото в `wp-config.php` файла или Web-сървърът неуспешно търси локална MySQL конфигурация (MySQL instance), към която да отвори сокет връзка (`socket connection`). Прост пример: при локален запуск на WordPress под Mac OS, MySQL създава сокет: `/tmp/mysql.sock` за локални свързвания към нея, но WordPress PHP-кодът ще търси: `/var/mysql/mysql.sock` посредством MySQL-модула на PHP-конструктора (PHP engine). За целта, чисто формално, указваме следната връзка:

```
# ln -s /tmp/mysql.sock /var/mysql/mysql.sock
```

Фактическият път на Файловата система до Локалния MySQL-сокет (MySQL socket) е функция, произтичаща от конфигурацията на Базата-данни; при стартиране, тя създава Локален сокет (`local socket`). Това къде PHP-конструкторът (PHP engine) и вследствие: всички PHP-базирани приложения ще търсят въпросния Сокет, се определя от PHP-конфигурацията. Действителния първоизточник на несъответствието можем да установим с помощта на безапелационното `printf()` отстраняване на стилови грешки (`style debugging`).

Редактираме `wp-includes/wp-db.php`: набора функции (`set of functions`), осъществяващи конекция към WordPress Базата-данни. Ако при инсталация ни се извежда съобщение: "Error establishing a database connection" „Грешка при свързване с Базата-данни“, добавяме `echo(mysql_error());` команда там, където е възникнала грешката с цел да видим детайли към Системното съобщение, подобно на изведените на Figure 1-6:

```
if (!$this->dbh) {  
    echo(mysql_error());  
    $this->bail(sprintf(/*WP_I18N_DB_CONN_ERROR*/"  
    <h1>Error establishing a database connection</h1>
```


Error establishing a database connection

This either means that the username and password information in your `wp-config.php` file is incorrect or we can't contact the database server at `localhost`. This could mean your host's database server is down.

- Are you sure you have the correct username and password?
- Are you sure that you have typed the correct hostname?
- Are you sure that the database server is running?

If you're unsure what these terms mean you should probably contact your host. If you still need help you can always visit the [WordPress Support Forums](#).

Try Again

FIGURE 1-6: `mysql_error()` съобщава за проблем във връзка със Сокета

Функцията `mysql_error()` е Функция на PHP Библиотеката, която извежда описание на Грешката, възникнала при последната извикана MySQL Функция.

- 2. Web finds MySQL but can't log in** (Web открива MySQL, но не може да Log-не). Обикновено MySQL username и/или password са сбъркани, най-често при прехвърляне на произволни username-и, генерирани от hosting-provider. Свепете своите username-данни и вижте дали същите съответстват на тези в `wp-config.php` файла. Ако използваме MySQL 4.1 или MySQL 5.0, е възможно е да се сблъскаме с password authentication проблем при PHP-приложенията на някои Web-сървъри поради поддържаната единствено стара MySQL 4.0 схема за хеширане на Пароли. В подобен случай използваме Функция `OLD_PASSWORD()` на MySQL, която хешира Паролите на WordPress user-ите в Обратно съвместим (backward-compatible) формат; с магическо SQL-заклинание (посредством MySQL command-line prompt или SQL-прозорца на [MAMP](#)) указваме следното:

```
SET PASSWORD FOR user@host = OLD_PASSWORD('password');
```

В случая, `user@host` е нашето WordPress database username и database hostname, а `password` (текстов низ) е Паролата, която сме указали в Конфигурационния файл (configuration file).

- 3. WordPress connects to MySQL but can't select the database** (WordPress се свързва с MySQL, но не може да се свърже с Базата-данни). Това, че Web-сървърът не може да се log-не към Сървъра с Базата-данни с Потребителските ни данни във WordPress Базата-данни не означава, че съответният user няма правомощия за съответната База-данни. Подобен сценарий най-добре диагностицираме с `mysql_error()`, вмъкнато в `wp-db.php`, с което диагностицираме грешката при опит за свързване:

```
function select($db) {  
    if (!@mysql_select_db($db, $this->dbh)) {  
        $this->ready = false;  
        echo (mysql_error()) ;  
        $this->bail(sprintf('/*WP_I18N_DB_SELECT_DB*/'  
        ... <h1>Can't select database</h1>  
        ..
```

Ако при вмъкване на командата `mysql_error()`, както току-що видяхме, при опит да приключим инсталацията, ни се извежда съобщението, показано на Figure 1-7, MySQL Базата ни данни не е създадена от правилния Потребител (database user) или същият няма правомощия да работи с нея. Проверяваме внимателно зададеното в MySQL с помощта на следния Команден ред:

```
% /usr/local/mysql/bin/mysql -u wp_user1 -p
Enter password:
Welcome to the MySQL monitor. Commands end with; or \g.
Your MySQL connection id is 174
Server version: 5.1.37 MySQL Community Server (GPL)
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| test |
+-----+
2 rows in set (0.00 sec)
```

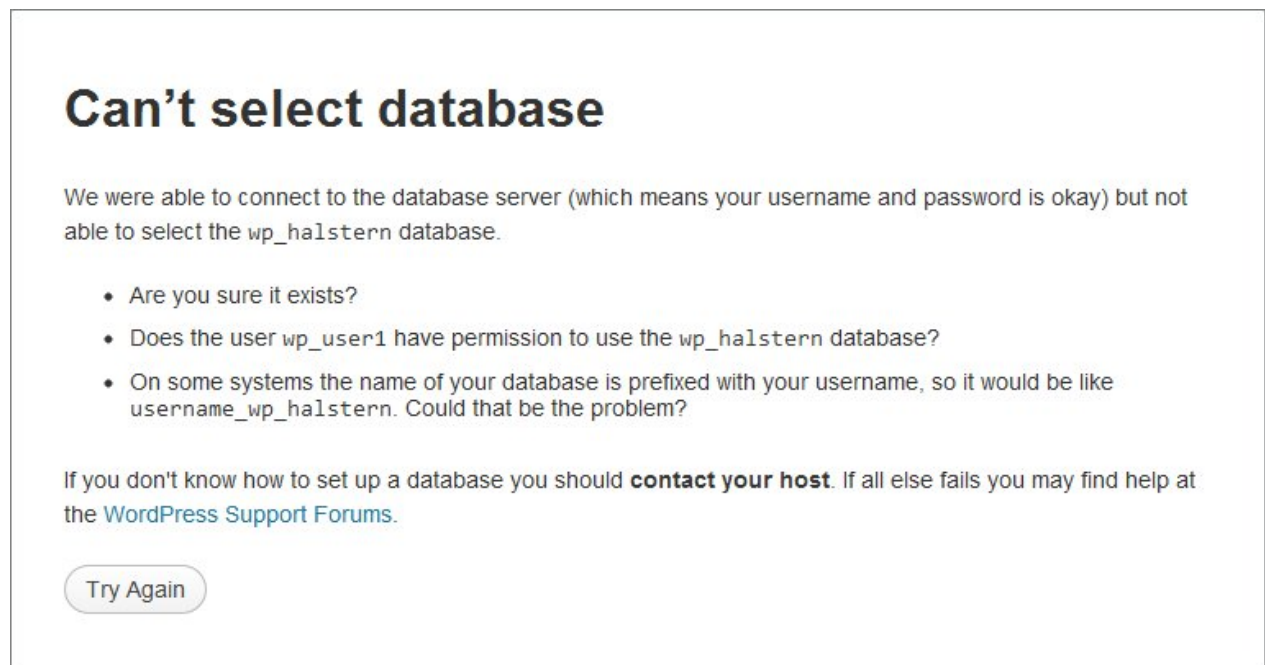


FIGURE 1-7: MySQL грешка при опит за свързване с База-данни

Log-нали сме се като user в желаната MySQL База-данни, но не я виждаме – в подобен случай тя вероятно е създадена от [root](#) потребител (user root/Администратор) на MySQL и правомощията за достъп или промени не са предоставени на MySQL user-а на WordPress-инсталацията. Ако имаме MySQL [root](#)-достъп или съответни MySQL Потребителски правомощия да създаваме нови Бази-данни в рамките на конкретния MySQL-модел (instance), след log-ване, База-данни създаваме с Команден ред (command line):

```
mysql> create database wp_halstern;
Query OK, 1 row affected (0.00 sec)
```

Припомняме: важно е да разграничаваме user-и на Операционната система от MySQL user-и или WordPress user-и. MySQL user-ите са указани в Базата-данни и имат предоставени правомощия да създават Бази-данни и обработват Таблицы или с други думи: да генерират полезни данни. WordPress user-и на Таблиците на WordPress Базата-данни се

създават при нейната инсталация; ролите и правомощията им имат отношение единствено по отношение на WordPress-сесиите.

При вече направена чиста WordPress-инсталация, би следвало да имаме набор Таблици, наименувани съобразно Табличния префикс, указан от нас в `wp-config.php`; и отново, най-удобен в случая отново е MySQL Командният ред (command line):

```
mysql> use wp_halstern; show tables;
Database changed
+-----+
| Tables_in_wp_halstern |
+-----+
| wp_hs_comments |
| wp_hs_links |
| wp_hs_options |
| wp_hs_postmeta |
| wp_hs_posts |
| wp_hs_term_relationships |
| wp_hs_term_taxonomy |
| wp_hs_terms |
| wp_hs_usermeta |
| wp_hs_users |
+-----+
10 rows in set (0.00 sec)
```

В случая указваме Префикс на Таблици на Базата-данни `wp_hs_`; ако при последваща WordPress-инсталация, използваме отново същите user и instance на Базата-данни, можем просто да укажем различен Prefix, при което и двете ще принадлежат към една и съща Таблица в Базата-данни. Работим със Схемата и разновидностите на десетте Основни таблици на WordPress Базата данни от Глава 6. На този етап, вече осъществили успешна конекция към MySQL, сме готови за съответни финални дейности по почистване и Първоначално администриране.

Заклучителни процедури

MySQL Базата ни данни е вече подготвена и работеща. В нея помещаваме нашето съдържание, а Web-сървърът ни с охота изпълнява Базовия WordPress-код. Остава да се запознаем с още само две неща.

Първоначално администриране

При успешно приключила инсталация, след log-ване с Потребителските данни от Figure 1-4, преминаваме към Контролния WordPress екран (Dashboard) от Figure 1-8.

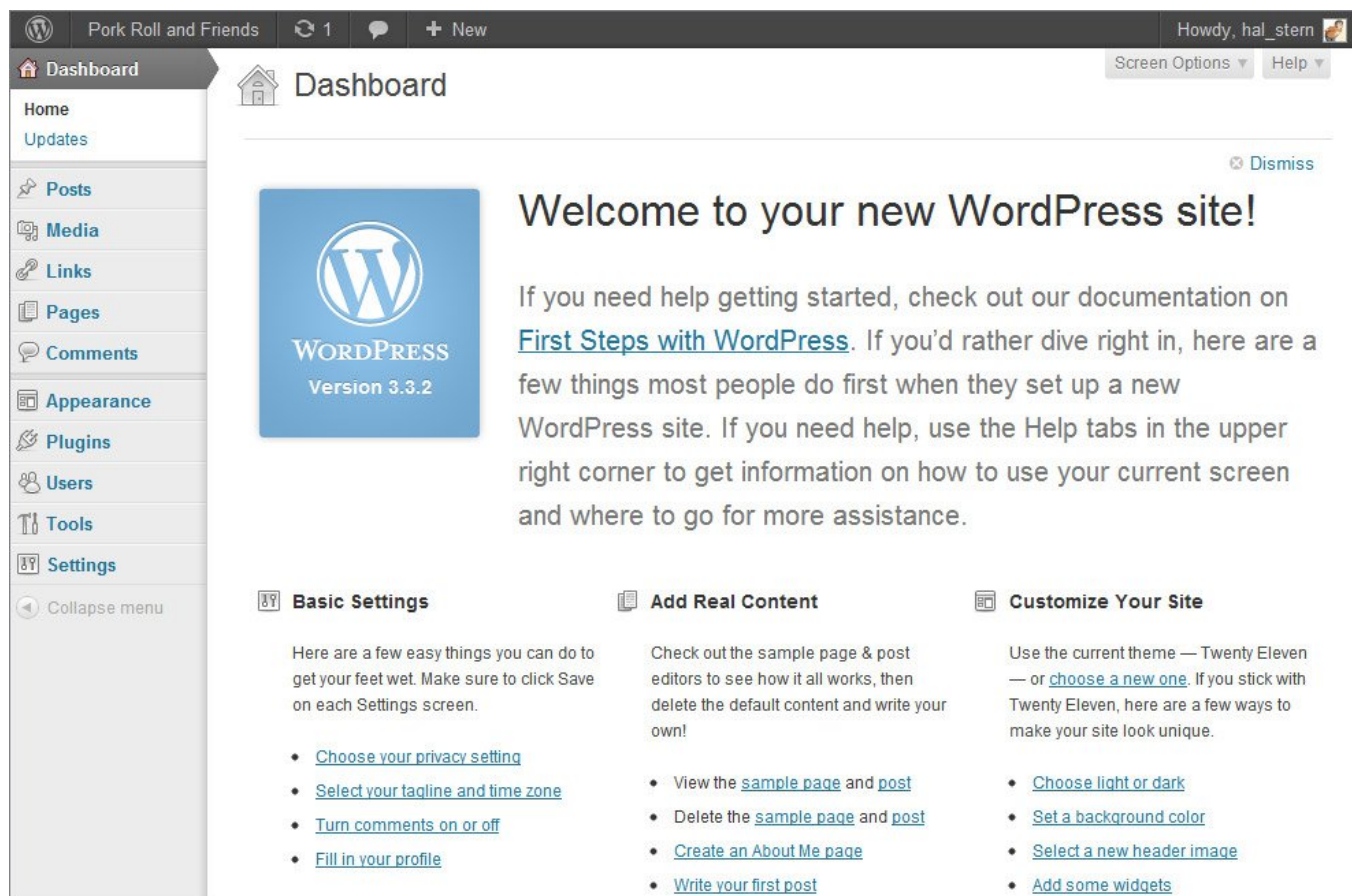


FIGURE 1-8: Примерен Контролен екран (Dashboard view) при първоначален login

Ако с натискане на Log In – бутона не се окажем на Контролния екран (Dashboard) или ако се озовем на нашия Базов (Top-Level) URL, следва или първо да изберем Log In link-а на нашия Web-сайт, или изрично да навигираме до Поддиректорията wp-admin (example.com/wp-admin), за да ни бъде изведен login Диалогов прозорец. Log-ването в нашия Web-сайт ни извежда Контролния WordPress-екран (WordPress Dashboard), едновременно удивително лесен при цялата си мощ и богат на възможности и индивидуални особености.

Последващите ни действия на Dashboard-а ще зависят от това доколко удовлетворени сме от първоначалната инсталация. Ако, както в горния Пример, инсталираната Версия на WordPress е стара, натискаме Бутон Update, с цел upgrade ad hoc към текущо най-новата Версия. Освен наличната подчертано изразена self-installation възможност, WordPress предлага self-update функции (същите се залагат във wp-admin/includes/update.php).

Може да се наложи да променяме базови конфигурационни настройки, от типа на Име на Базата-данни или Име на потребител на MySQL Базата-данни, при все че ако разполагаме с пълен контрол над Web-сървърите и Сървърите на Базата-данни, би била достатъчна промяна в root@localhost. В Конфигурационния файл също са указани „ключовете за защита“ (“security keys”), използвани с цел по-сигурна защита на Browser cookie-тата. Ключовете за защита разглеждаме детайлно в Глава 11. Промените във Файл wp-config.php влизат в действие незабавно. Промяна в Префикса на Таблица в Базата-данни, например, кара WordPress да генерира нов набор Таблицы и да започне Първоначална инсталация. Осъществяваме споменатите промени, навигираме обратно до нашия Базов URL адрес (Top-Level URL), и вече сме Log-нати като нов Admin-потребител на Първоначален екран, сходен с показания на Figure 1-8. Досегашните Таблицы все още фигурират в MySQL, откъдето ще трябва да ги изтрием ръчно.

Сега би следвало, ако искаме URL-адресът ни да е различен от този на локацията, където сме инсталирали WordPress, да изберем Settings/General от Контролния екран Dashboard и да

променим както нашия Базов URL адрес (Top-Level URL), така и Директорията на WordPress инсталацията. Ако не желаем URL адресът на Сайта ни да е свързан с Директорията на WordPress, ще трябва да преместим `index.php` файла на желанния Базов URL адрес (Top-Level URL), както и да променим пътя до съответната Поддиректория с WordPress в най-долния му ред.

Преди да създадем първия си Post, би следвало да уточним [Permalink](#) (Permanent Link – относително неизменен, перманентно актуален URL-адрес, по възможност неподатлив на [link rot](#): случаи, при които Хипервръзки в нечий Сайтове или в Интернет въобще, препращат към текущо неактивни или отсъстващи Web-страници, Сървъри или други ресурси; link-rot наблюдаваме и при неподдържани, остарели Web-страници, каквито изобилстват сред извежданите от Търсачките и всяват неизбежни объркване и затруднения) структурата и всичкият ни Код да е съобразен със следваните Конвенции на именуване (Naming Conventions), избрани от нас с цел относително да улесним посетителя при търсене, разделяне и указване на Връзки (Link-ове) към нашето Съдържание. Споменатата опция резонно е включена в Настройките (Settings) на Контролния екран (Dashboard); варианти за permalink именуване и следствията им върху Производителността (Performance) и Схемата на Базата-данни (Database Schema) са разгледани с повече подробности в следващата Глава.

Важно е, че след като сме отделили нужните пет минути или няколко часа за идентифициране на несъответствия при Имена на хостове (Hostnames), Потребителски имена (usernames) и Конфигурации на Базата-данни (database configurations), вече сме готови да публикуваме своя собствен първи Интернет Post.

Първи Post

При успешна WordPress инсталация, вече имаме публикувани първи Post и Коментар (Comment), като всички подвижни елементи се местят в синхрон и съставляват единно съдържание на нашия Web-сайт. За да добавим и няколко думи, използваме десния QuickPress панел (panel) на Контролен екран (Dashboard) където въвеждаме текста (при което първоначални ни се извеждат напътствия по отношение на новия Web-сайт), или пък отиваме на Posts и избираме Add New (Добавяне на нов), откъдето преминаваме към заложения WordPress редактор (editor). На Фигура 1-9 показваме процеса на въвеждане в QuickPress панела, при което Контролният екран (Dashboard) е актуализиран след вече успешен първи Post.

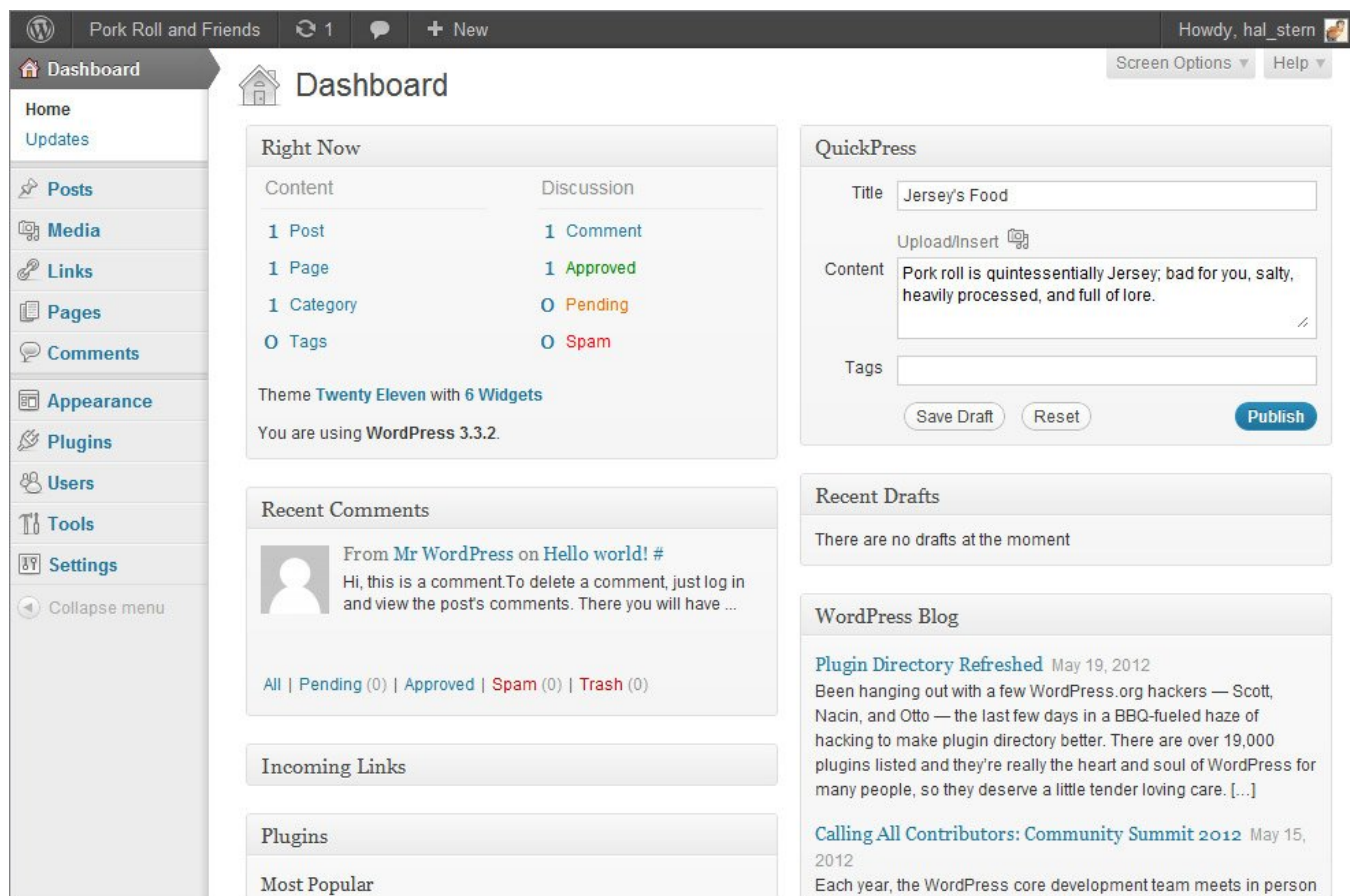


FIGURE 1-9: В ход е публикуване от QuickPress панела

Ако сме малко по-старомодни, можем да нахвърляме съдържанието в любимия ни Текстов редактор и после да го копираме в Панела за редактиране (Editing Pane). При работа с [WYSIWIG](#) Текстови редактори от типа на Microsoft Word или OpenOffice, при опит за пренасяне във WordPress HTML composition прозореца трябва да знаем, че в HTML-кода ще изобилстват множество излишни Tag-ове и форматиране. Има и множество автономни Редактори, от типа на Illumnix's Ecto, които експортират във WordPress посредством [Atom](#) Publishing Protocol или [XML-RPC](#). Опциите с възможност отдалечено да публикуваме Post-ове са поместени, както можем да очакваме, в Settings (Настройки) секцията на Контролния екран (Dashboard), под Настройки при въвеждане (Writing options).

Избираме Publish за собствения първи наш "Hello World". Множество подсистеми участват в извеждането на Панела за редактиране (Editing Pane), съхраняват съдържанието в База-данни, генерират и съхраняват релевантните Метаданни и впоследствие генерират елегантно изглеждащ HTML. Много от нещата, видими за Потребителя, се управляват посредством Контролния екран (Dashboard) и някои от въпросните функции ще разгледаме в следващите Глави

Обобщение

В настоящата Глава разглеждаме как WordPress извоюва днешното стабилно присъствие, направихме кратка историческа справка и разбрахме за повсеместната му популярност. Възходът на WordPress в сферата на Web-приложенията е обусловен от интуитивния инсталационен процес. В следващата Глава изследваме ядрото на WordPress, с цел да почерпим позитиви от неговите мащабируемост, интуитивен дизайн и функционалност.