The Drupal-Powered Enterprise

Three ways to capitalize on Drupal as a unified platform for low-cost, central management of multiple websites

Delivering Extraordinary Web Experiences without Extraordinary Burdens

What an enterprise expects

Today, digitally savvy companies manage a growing set of web experiences, targeting specific channels, audiences, markets, and outcomes. When promising opportunities arise, smart firms move quickly to launch new experiences. But the willy-nilly addition of standalone sites within the enterprise – typically built on an array of technologies by disparate groups — leads to wasted efforts, unnecessary expenses, and problems sharing content or capabilities. Customers also feel the pain, as they are bombarded by mixed messages and disconnected websites.

When juggling multiple sites and experiences, enterprises face the dilemma of scale: how best to leverage IT investments for an economical yet sustainable solution. Here are a few problems that companies typically face:

- **A pharmaceutical company** invests considerable effort developing product information that meets regulatory mandates. To limit the risk of distributing incorrect or inconsistent information to various markets and segments, the firm needs to centrally manage the content. Yet it must also ensure that individual segment owners can easily produce compelling web experiences for their target audiences and update them at the speed of business.

- **A consumer products company** maintains several lines of business, each with a unique branded experience. The firm needs to control its IT infrastructure, while ensuring that brand managers can promote their products through websites designed to meet their own needs and publishing their own content on their own schedule.

- **A media and entertainment firm** fuels fan loyalty by producing targeted websites about its various stars. Rock fans want to groove to favorite tunes and performers, and track concert schedules. By comparison, film buffs are enthralled with storylines and video downloads. The company seeks the capabilities of a unified platform while maintaining multiple web experiences.

Every company needs to coordinate content delivery, control costs, and optimize investment across multiple websites. The solution is a consistent platform for content delivery — one that is sufficiently flexible to meet both current requirements and future growth opportunities. It is essential to rely on a scalable content infrastructure that can maintain multiple web experiences, while also reducing development and operating costs.

To date, this has been a need unfulfilled. Drupal, however, changes the landscape.

How Drupal delivers

Drupal provides an integrated platform for powering content-rich web experiences across the enterprise. These experiences, in turn, are maintained by one or more discrete websites. Drupal delivers the unparalleled flexibility and extensibility for supporting a diversity of business activities on the web. Being an open-source project, Drupal does all this at vastly lower costs than proprietary or customer-built solutions while harnessing the innovation of a global community of web developers.

Drupal relies on a contemporary, component-based architecture. As shown in Illustration 1, discrete modules encapsulate features, invoke actions, and perform functions. A Drupal-powered site manages content as self-contained nodes (or unique objects separated from "pages" or delivery channels) in a database. Nodes are tagged with extensive, context-specific metadata.

A Drupal-powered website then displays content through a series of views and themes. The same text, image, or video stored in Drupal can be delivered across multiple websites or mobile applications and "mashed up" with other content in almost any desired way. As a result, Drupal dynamically adapts web experiences to varied devices and business situations.

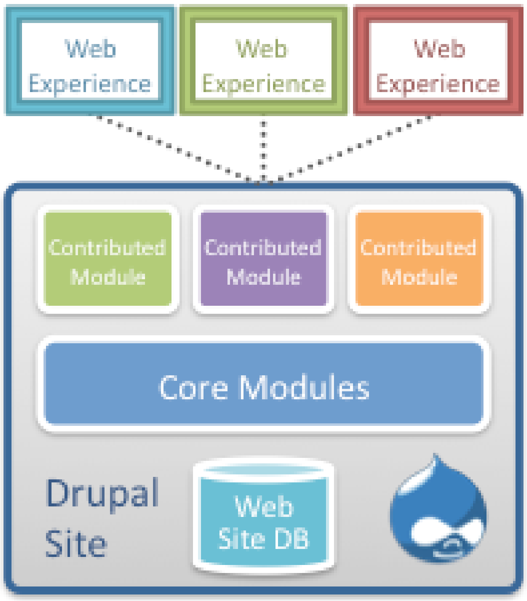

Illustration 1. A Drupal site includes a predefined set of core modules together with an extensible set of contributed modules. Drupal stores all nodes and other entities in a website database. Drupal then produces multiple web experiences on based on this content infrastructure.

Furthermore, Drupal capabilities and configurations are packaged into specific, reusable software “distributions.” With a Drupal distribution, website functions and services are tailored to focus on identified business purposes, and then rapidly reused and extended to suit new requirements, with little additional investment. A Drupal distribution includes a predefined set of modules and features, specified content types, tailored configurations, and a defined installation profile.

The platform for a scalable content infrastructure

As the platform driving a scalable content infrastructure, Drupal encompasses three distinct yet interrelated sets of capabilities. Drupal supports:

- Multiple interactive experiences produced by a single website
- Many websites spawned from a single Drupal distribution
- Various Drupal distributions managed by a deployment environment

Consequently there are three approaches a company can adopt to exploit the platform-level capabilities of Drupal:

- **Multi-tenant: One database, one Drupal installation.** Multiple tenants share both the software files and database In Drupal, this is sometimes known as the “Domain access” method.
- **Hybrid: Multiple databases, one Drupal installation.** Multiple tenants run on one installation of Drupal, but with a separate database for each tenant. This is the most common in practice in Drupal world, where it’s known as “Multisite”.
- **Multi-instance: Multiple databases, multiple Drupal installations.** Each tenant has its own Drupal installation and database, although they could be managed through a unified deployment environment.

Each approach highlights a different set of capabilities for the Drupal-powered enterprise, as summarized in Table 1. Each also provides a specific set of business benefits in terms of optimizing investments in an underlying content infrastructure, managing operating and development costs, and ensuring effective IT support for multiple web experiences.

Let’s examine how each of these three approaches work, and describe how best to optimize Drupal for particular business situations.

	Flexibility	Safety (from sharing)	Cost savings
Multi-tenant	■	■■	■■■■
Hybrid	■■	■■■	■■■
Multi-instance	■■■	■■■■	■
Multi-instance without code management ("Cowboy model")	■	■■■■	■

Table 1. Drupal supports three distinct approaches for producing multiple web experiences, shown here compared to a non-managed approach

Multi-tenant

How a single site powers multiple experiences

With a multi-tenant configuration, a single instance of a Drupal-powered website produces an extensible set of related sites, each addressed with independent domains (e.g. foo.com and bar.com) or sub-domains (foo.site.com, bar.site.com). Most important from a platform perspective, there is a single website database, running on a LAMP (Linux, Apache, MySQL, PHP) infrastructure. Thus a multi-tenant configuration features a shared content collection, organized and structured through a common schema and a single security model. Each related site can provide a unique web experience while being managed centrally.

Customers, stakeholders, and other groups access various websites, designed to meet their specific needs. Each site has the look-and-feel they expect and provides them with easy access to targeted information. But from an enterprise perspective, a "master" website produces multiple experiences on different related sites.

For instance, a local hospital can meet the needs of its multiple stakeholders by maintaining several websites — one for the general public, patients, and family members, and another for doctors, nurses, and support staff.

- The public- and patient-facing site is designed for easy browsing. The site includes a lot of content about preparing for a hospital stay, where to park, visiting hours, and steps for discharge. It also includes patient contact information, secured to be visible only to patients and their caregivers.

- The staff-facing site is designed for rapid retrieval and includes the hospital phone directory, links to department-level information, policies and procedure manuals, and other kinds of organizational information, all secured by authenticated access. Doctors, nurses, and support staff log into the site and automatically find the content for their departments and specialties.

With a multi-tenant configuration, the hospital's IT group only needs to deploy and manage a single Drupal installation. All of the content is stored in the website database, indexed by relevant criteria, and rendered on related sites to produce varied experiences. Content is developed once, managed centrally, controlled with multiple levels of permissions, and published where needed.

To continue the example, the hospital can prominently display transportation and parking information on the home page for the public-and patient-facing site; this same information can be accessed as a link on the staff-facing site. When a hospital administrator updates the parking information, he or she only needs to make the change once within the website database, and the new content is automatically distributed to all related websites.

Separating presentation from content storage and access

A multi-tenant configuration exploits a key capability of Drupal: separating the presentation and visual elements of the user experience from content access and storage. The Drupal-powered site includes a predefined set of modules and stores content within a website database, as shown in Illustration 2.

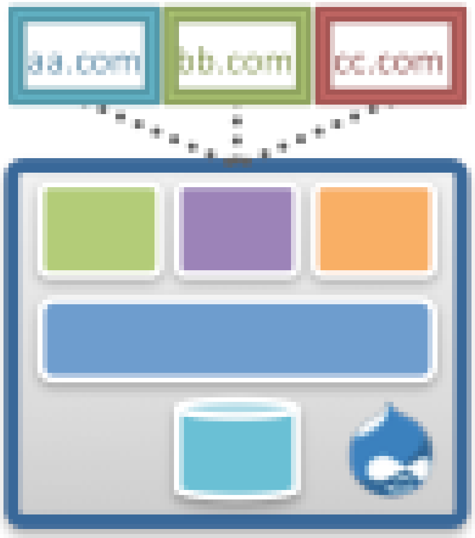

Illustration 2. With a multi-tenant configuration, there is a single underlying Drupal site and multiple web experiences, each accessed with a unique URL. The master site encompasses a predefined set of Drupal modules and a single web database.

Related sites can have their own unique views and themes – including distinctive visual designs, display layouts, and navigation capabilities that are primarily applied with Cascading Style Sheets (CSS). Related sites manage the presentations for multiple web experiences maintained by the master Drupal website. It is easy to configure and launch a new site, simply by designing the appropriate user experience and linking it to content maintained within the master site.

Drupal centrally stores and maintains all of the content for related sites within a website database. With a multi-tenant configuration, these sites rely on:

- A single set of content types
- A common taxonomy of terms
- A unique metadata schema
- An underlying security model for determining user authentication, access rights, and permissions

Thus an enterprise centrally manages all of the content published on multiple related sites. There is a unified content production process, a common editorial workflow, and a standardized way of tagging content. There is also a standardized way for managing website security and content permissions. It is easy for authenticated users within an enterprise to share content across sites. A single IT group manages access rights and permissions across the various sites and web experiences.

When a multi-tenant configuration makes sense

A multi-tenant configuration works well when an enterprise needs to centralize content governance and management across multiple websites. For instance, government agencies, pharmaceutical firms, and financial services companies frequently need to maintain an approval process for publishing authoritative content on multiple sites. Marketing organizations and advocacy groups often need to quickly launch new websites, to support new campaigns and promotional events. For these (and other) business situations, it is important to be able to rapidly deploy related sites, leveraging the content stored within a single website database.

With centralized storage and content management, a multi-tenant configuration helps to reduce the costs of business operations. An IT group can cost-effectively support multiple web experiences, particularly when there are benefits to sharing content among sites. Furthermore, when it comes to managing software updates, the changes only need to be applied once to the master site. Updates automatically propagate to all related sites. With a multi-tenant configuration, an enterprise can centrally manage content, and have the capabilities to redistribute it as needed across multiple web experiences.

Hybrid

Spawning multiple websites from a single distribution

A hybrid configuration provides another approach for maintaining multiple web experiences across an enterprise. With a hybrid configuration, a single Drupal distribution spawns many stand-alone websites. Each Drupal-powered website has its own website database, manages its own content, and maintains its own set of authenticated users.

A hybrid configuration also centralizes essential IT functions within the enterprise. All sites run within a common application environment, including a single instance of a LAMP infrastructure. All sites include an initial set of modules, templates, content types, and configuration settings, specified by the Drupal distribution. The IT group manages the systems-level activities of the application environment – including the overall security for websites and software updates. The IT group is also able to maintain the technical road map to determine the evolution of the environment.

Significantly, the IT group is able to distribute the day-to-day administrative control for content development and publishing to site-level administrators. Business users maintain the content on their own websites, within the parameters defined by the Drupal distribution. They can also add site-specific content types and configuration setting, if permitted to do so by the IT group.

For example, the United States House of Representatives is now in the midst of transitioning its approximately 520 member, committee, leadership, and administrative sites from a disparate collection of stand-alone websites to Drupal. Each Drupal-powered site provides a discrete web experience, within the hybrid-enabled content infrastructure maintained by the House's Chief Administrative Officer (CAO) and a small support staff.

The House has adopted OpenPublic (a Drupal distribution from Phase2 Technology) for its hybrid deployment. OpenPublic configures Drupal to manage government and legislative affairs. Each member and committee office is able to maintain a distinctive web experience; office-level administrators easily create, modify, and manage content for their own Drupal-powered sites without involving the central IT group. At the same time, the CAO and staff are able to centralize IT operations, deploy a flexible platform, provide overall technical direction and support, ensure website security, and optimize the costs of maintaining the multiple websites across the House application environment.

A Drupal distribution for assembling a web application

A hybrid configuration exploits another key capability of Drupal: the ability to assemble a specific Drupal distribution into a customized web application. With Drupal, it is easy to produce a content rich web experience, incorporate both structured and unstructured data sources, and blend published information with user-generated content.

A Drupal distribution assembles a predefined set of content- and community-enabled modules into an interactive application for a business purpose, as shown in Illustration 3.

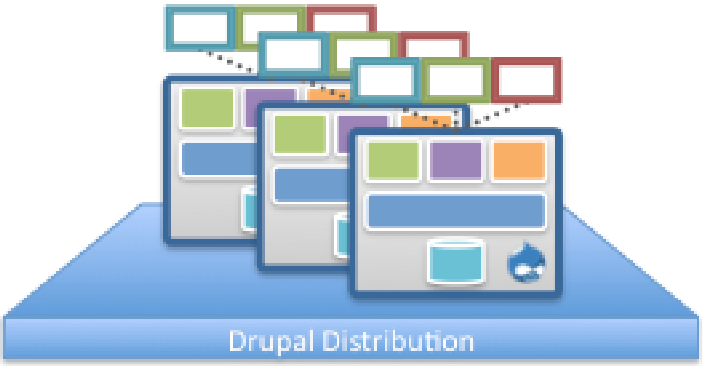

Illustration 3. With a hybrid configuration, a single Drupal distribution spawns an extensible set of Drupal-powered websites. All sites have the same set of modules and their own website databases.

A Drupal distribution includes:

- The core and contributed modules that create the application-level features
- The initial set of content types and taxonomy terms for defining and managing content within a website database.
- Multiple views and themes for displaying content and providing the user experience

A Drupal distribution also includes a predefined set of user profiles for defining access rights, roles, and permissions, as well as a standardized installation profile. (Individual site administrators can change these later on their own sites.) As a result, an IT group within an enterprise can easily roll out multiple Drupal-powered websites from a single Drupal distribution. The IT group can manage the underlying technical and systems infrastructure, including the overall site-level security.

But beyond the enterprise-level coordination, each Drupal-powered site functions independently. Site-level administrators have the flexibility to define the capabilities for their specific sites. They can manage access rights and permissions. They can tailor the user experience to meet their needs by adapting the graphic displays provided by particular themes and views delivered with the distribution. They can customize the content types and taxonomies provided by the distribution to create site-specific items. They can decide to turn on or turn off various features and modules, and thus tailor the capabilities of their own sites without IT support.

When a hybrid configuration makes sense

A hybrid configuration works well when an enterprise needs to manage core IT functions, while also supporting multiple websites. For instance, multiple marketing groups within a consumer electronics firm may want to promote new branding initiatives. Similarly, different business units within a financial services firm may want to feature their unique products and services on their own websites.

In these and other cases, the business groups within an enterprise operate autonomously from one another. They expect to maintain independent sites, at an affordable cost. But these self-contained and autonomous groups also rely on an enterprise IT group to provide technical support and policy-level guidance.

At the same time, the enterprise IT group needs to optimize investments, reduce operating expenses, and coordinate the overall growth of the web infrastructure. The IT group needs to focus on the systems-level operations: maintaining the security for an entire collection of sites, and centralizing their updates and evolution.

With a hybrid configuration, Drupal delivers the business solution for the enterprise: distributed content management with centralized IT support for essential system-level activities.

- Business units have the flexibility to deliver the engaging web experiences that drive their business strategies. They can administer and promote their own Drupal-powered sites at a fraction of the cost and management overhead of stand-alone sites.

- From an IT perspective, there is a single Drupal distribution with a consistent set of features and modules, running within a single application environment. IT support and management costs are centralized and optimized.

In short, from an enterprise platform perspective, there is rapid payback to spawning multiple Drupal-powered sites from a single Drupal distribution, with each site designed for a specific business purpose.

Multi-instance

Simultaneously supporting several distributions

In some situations, a hybrid configuration is only the beginning. Sometimes an enterprise committed to a web-wide digital presence needs multiple sets of Drupal-powered sites, generated by several distributions.

There is a third approach for producing the web experiences across the Drupal-powered enterprise. It is important to channel the flexibility of Drupal and focus on the overall deployment environment for building distributions. With a multi-instance configuration, developers working within an IT group are able to simultaneously assemble and evolve several Drupal distributions.

A multi-instance configuration includes synchronization tools to manage the build and deployment processes for a distribution, together with quality assurance (QA) tools and testing harnesses to verify and validate the various releases. The IT group thus maintains a unified deployment environment that supports the build, test, and release activities required for rolling out Drupal distributions.

Why should an enterprise consider a multi-instance configuration? Let’s consider how a media and entertainment company with several divisions can use Drupal to power different sectors of its overall web presence. It’s essential to focus first on the business purpose for a web presence.

- The studio division needs to continuously promote newly released movies while they are available in theaters. Movie sites seek to build the buzz of fan awareness with social media links and trailers.

- The home entertainment division needs to offer DVDs and digital downloads of the movies, ranging from recent releases to oldies. Home entertainment sites need to market digital products in multiple formats and languages, and include e-commerce capabilities for managing digital rights and doing business on the web.

Each division relies on particular sets of websites encompassing some different capabilities, deployed by various Drupal distributions. Each division may have independent internal or external development resources. At the same time, the IT group needs to maintain the overall enterprise architecture and infrastructure, optimize development activities, and reduce ongoing support costs.

Coordinating the assembly of drupal distributions

A multi-instance configuration leverages the modular infrastructure of Drupal. It coordinates the assembly of modules into a Drupal distribution through a unified deployment environment, as shown in Illustration 4.

Each Drupal distribution contains a predefined set of modules, content types, and features for powering various websites. Specifically, a distribution includes:

- A common set of core Drupal modules required to run within the enterprise application environment
- A varied set of contributed modules designed to support specific business tasks
- Site-specific modules and site themes that extend the capabilities of contributed modules in unique ways

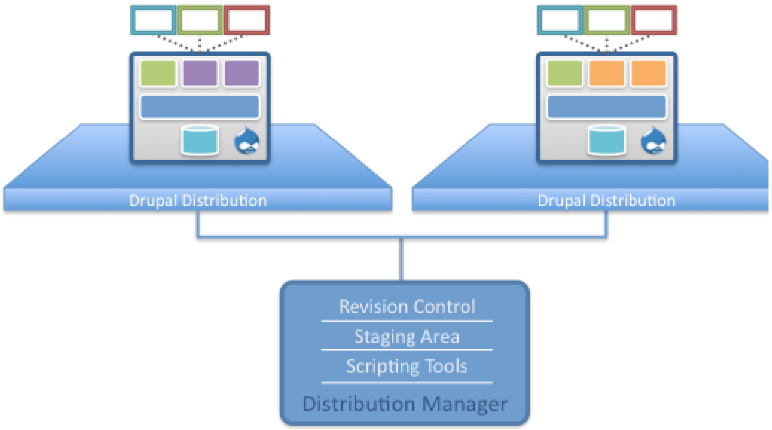

Illustration 4. With a multi-instance configuration, a unified deployment environment supports multiple Drupal distributions and manages the differences among them.

Synchronization provides the environment for coordinating the assembly and release of Drupal distributions. Each distribution incorporates the common set of core Drupal modules together with particular contributed modules.

A multi-instance configuration is designed to automate the build and deployment activities for discrete distributions. It leverages the capabilities of a revision control system (RCS) together with continuous integration (CI) tools.

Within this unified deployment environment, the tools perform such functions as:

- Maintaining an inventory of all modules
- Scripting the build and deploy processes for distributions
- Logging and registering all modules that are included in a build
- Managing the version updates to modules in the build list

Modules that developers check into the RCS-maintained repository are automatically incorporated into the build and deployment process for the relevant distributions.

With a multi-instance configuration, the IT group can automate and track many labor-intensive activities. The IT group is able to rely on the scripted build, test, and release processes to assemble modules and other elements into various Drupal distributions. The IT group implements predefined workflow sequences to control the steps for managing changes, and thus centralize the overall management and testing of Drupal distributions. As a result, the IT group is able to manage the release cycles of multiple distributions in a consistent fashion and improve its overall operations.

When a multi-instance configuration makes sense

A multi-instance configuration works well when an enterprise maintains a web-wide digital presence and needs to marshal resources for producing Drupal sites. With a multi-instance configuration, an enterprise can accelerate Drupal deployment and release activities, and produce unique web experiences for multiple business units, while also limiting dependencies among websites.

Specifically, the IT group can improve the coordination and automation of its build and test processes. Developers have the added resources of a unified deployment environment. They can work on projects over the Internet and easily add their modules to the shared repository within the enterprise.

In addition, the IT group no longer needs to track dependences among Drupal distributions and release cycles. With a multi-instance configuration it's easy to keep track of multiple versions of modules and to upgrade distributions on their own schedules. The IT group can automate the routine and repetitive tasks of the build, test, and deployment processes, and can more efficiently assemble Drupal modules into unique distributions. Fewer developers are needed to complete routine tasks and activities. They can spend more of their time creating new capabilities that meet business needs.

With a multi-instance configuration, an enterprise builds business value from its ongoing Drupal investments and harnesses the underlying flexibility of Drupal. An enterprise is thus able to reduce deployment costs while streamlining key business processes. In short, when making a web-wide commitment to a digital presence requiring distinctive experiences, a multi-instance configuration provides the fuel for the Drupal-powered enterprise.

Calculating the Drupal Difference

For large-scale deployments across an enterprise, Drupal supports diverse web experiences. With its modular infrastructure and its attention to all things content-related, Drupal combines an easily extensible platform with a broad set of content services to rapidly assemble a compelling web presence. Drupal reduces the costs and complexity of application development while optimizing ongoing management, maintenance, and support investments.

There are three approaches for building the Drupal-powered enterprise: multi-tenant, hybrid, and multi-instance. Each encompasses different sets of capabilities to leverage Drupal in unique ways.

- **Multi-tenant** centralizes content management with Drupal by maintaining a consistent set of web experiences within the enterprise. With a multi-tenant configuration, an enterprise can easily develop content once and rapidly distribute it to multiple audiences and devices.

- **Hybrid** distributes multiple Drupal-powered sites across an enterprise to produce a consistent experience, while centralizing the administration, security, and control of key IT activities. Content can be managed locally while web site deployments are coordinated centrally. With a hybrid configuration, an enterprise can rapidly develop a comprehensive web experience, easily redistribute it to various websites, and continue to manage and evolve the underlying IT infrastructure.

- **Multi-instance** extends the distributed capabilities of Drupal by maintaining the multiple experiences produced by several distributions. The content and interactivity of Drupal can be further distributed across an enterprise, and customized for specific business situations. With a multi-instance configuration, an enterprise can easily manage the deployment of multiple Drupal distributions, each encapsulating unique web experiences, while optimizing and expediting the work of Drupal developers to produce them.

Using Drupal, an enterprise solves its dilemma of scale in a flexible fashion based on business requirements. As a result, an enterprise can profit from an easily extensible content infrastructure, maintain multiple web experiences, rapidly integrate with disparate enterprise systems, and substantially reduce both development and operating costs. In short, the Drupal-powered enterprise can readily capitalize on the continuing opportunities for doing business in today's digitally driven economy.

About Acquia

Acquia empowers enterprises with the open-source content-management system Drupal. Co-founded by Drupal's creator in 2007, Acquia helps customers manage their growth and scale their online properties with confidence. Acquia's software, consultation, cloud infrastructure, and services enable companies to realize the full power of Drupal while minimizing risk, as it's done for Twitter, Examiner.com, Al Jazeera, and over 700 others. See them at **http://showcase.acquia.com** and learn more at **http://acquia.com**.

Acquia, Inc.
150 Presidential Way
Suite 310
Woburn, MA 01801
USA

www.acquia.com
sales@acquia.com

+1-978-296-5250