

# Introdução ao Drupal

v2.0 2016

# Introdução ao Drupal



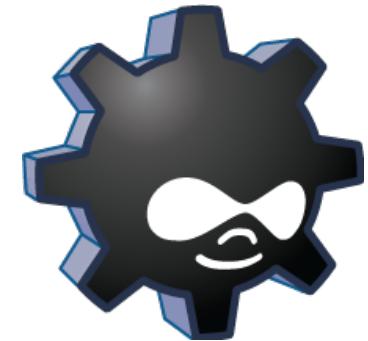
# Introdução

# Introdução: O que é o Drupal?

O Drupal é uma *Content Management Framework* (CMF): um sistema que permite a reutilização de componentes para gerir conteúdo web – construir sites (e muito mais).

Como tal, partilha conceitos de:

- *Web Application Framework* (WAF)
- *Content Management System* (CMS)



É *Free and Open Source Software* (FOSS), com licença GPLv2, *cross-platform* e foi escrito com a linguagem de programação PHP.

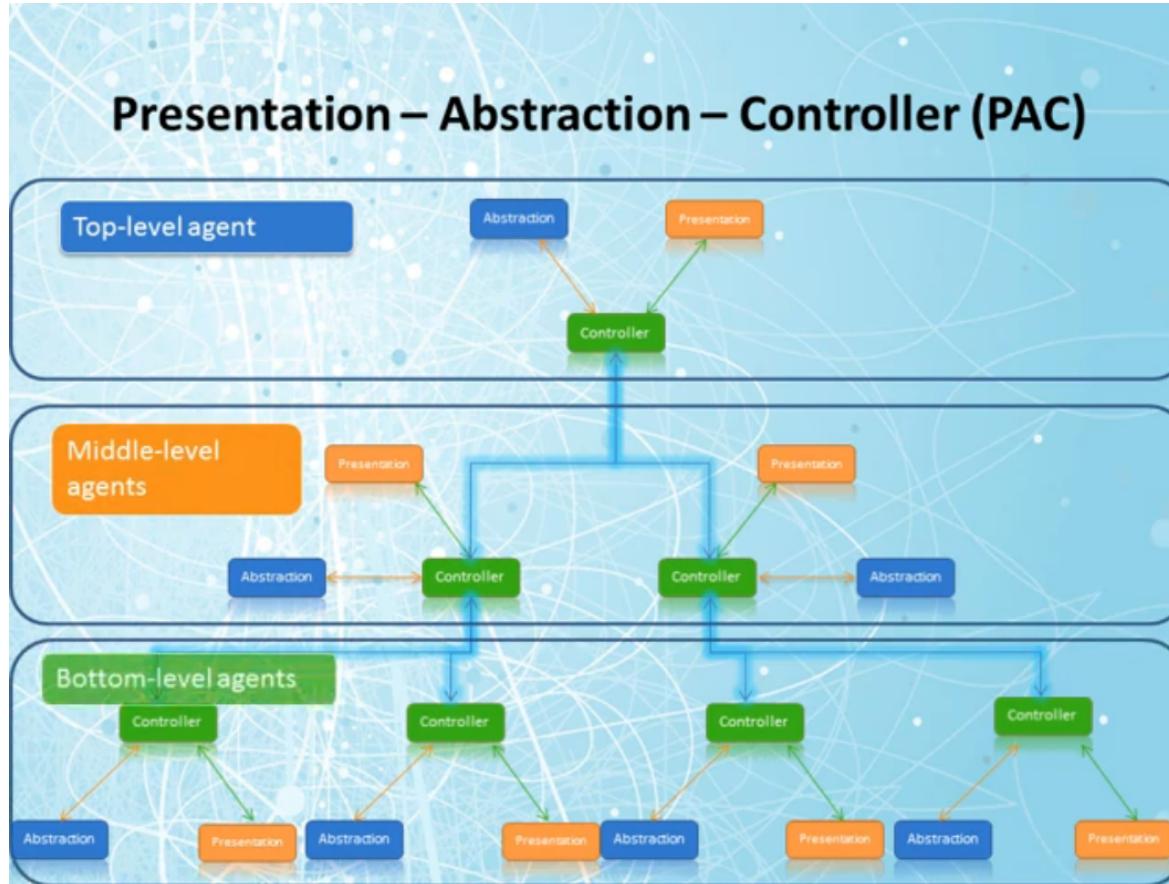
# Introdução: Porquê o Drupal?

Para os **developers**:

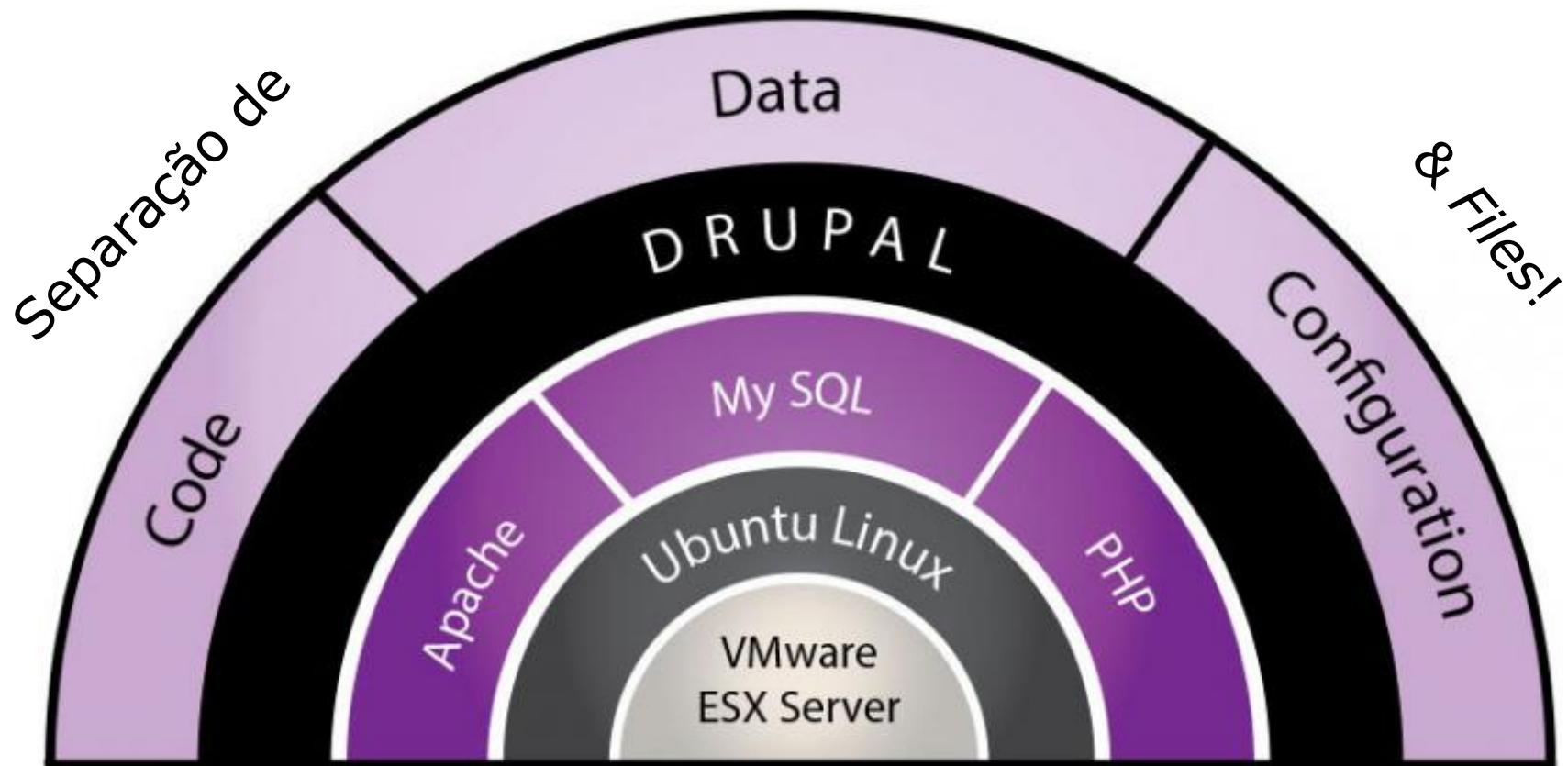
- 2-em-1: *Procedural & Object oriented programming*
- *Presentation-Abstraction-Control* (PAC) como padrão
- API excelente, extensível com sistema de *Hooks*
- *Content e Data Modeling*, via API e UI flexível
- *Coding standards* rígidas, com auditorias frequentes
- Separação de código, configuração e dados
- Eco-sistema com paradigma de dependências



# Introdução: Porquê o Drupal? (cont.)



# Introdução: Porquê o Drupal? (cont.)



# Introdução: Porquê o Drupal? (cont.)

Para os ***site builders*** e ***themers***:

- *Interface* muito funcional e intuitiva
- Sem requisitos de programação especiais
- Módulos para todo o tipo de funcionalidades
- Re-utilização de código e *interligação* de módulos
- Temas configuráveis e com suporte para tecnologias *Web 2.0+*, ie, HTML5, SASS/SCSS, *Grids*, etc
- Distribuições pré-preparadas para variados fins



# Introdução: Porquê o Drupal? (cont.)

Para as **empresas** e **clientes**:

- Parcerias com a [Associação Drupal](#), [Acquia](#) e outros
- Todo o tipo de [soluções universais e à medida](#)
- Sem custos de [licenças](#)
- Rede global de *developers*

**Acquia**<sup>TM</sup>



**Drupal  
Portugal**

**Drupal**<sup>TM</sup> Association

# Introdução: Porquê o Drupal? (cont.)

Para **todos**:

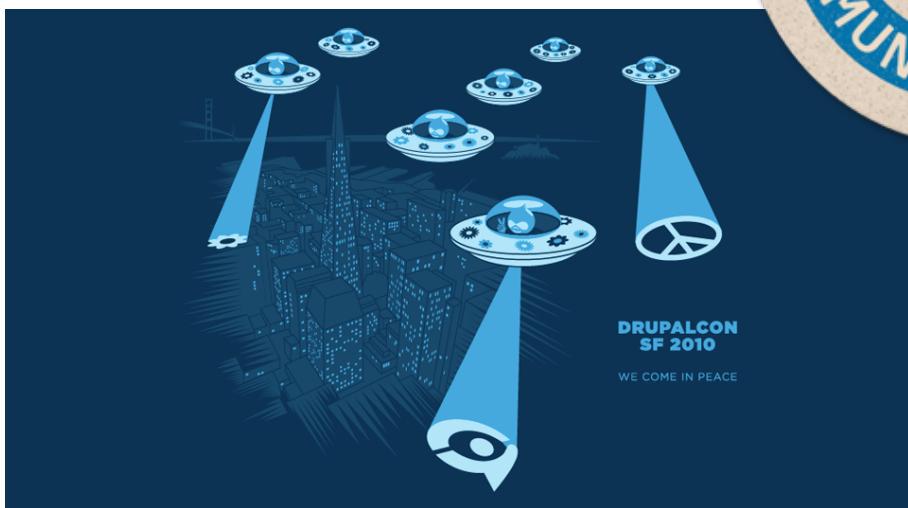
- Comunidade global muito grande
- Eventos internacionais (DrupalCons e DrupalCamps)
- Suporte e formações disponíveis em todo o mundo



# Introdução: Porquê o Drupal? (cont.)

Eventos anteriores:

DRUPAL CAMP  
PORTUGAL 2012



23-27 SEPT 2013  
DRUPALCON  
Prague

DRUPALCAMP  
LISBOA 2011

# Introdução: Porquê o Drupal? (cont.)

Eventos anteriores:



# Introdução: História do Drupal



Desenvolvido por Dries Buytaert, o Drupal teve as suas **origens** em 2000. Versões **mais recentes** do *Core* do Drupal sofreram *rewrites* significativos:

- **4.7.0** (2006) – Rewrote Forms, added Tagging & PHPTemplate
- **5.0** (2007) – Rewrote Admin, added Installer, Status & jQuery
- **6.0** (2008) – Rewrote Menu API & UI, added Schema API
- **7.0** (2011) – Rewrote Core with PDO & Entities, improved Scalability and UX, added SimpleText, Fields and many modules
- **8.0** (2015) – Rewrote Core with Symphony plus Twig and YAML, added PHPUnit, Web Services, HTML5, Views and more...

# Introdução: A comunidade

A [comunidade](#) do Drupal hoje em dia é global, com participantes de todo o mundo, de todos os estratos sociais e dos mais variados ramos e especializações:

- Grupos regionais, tais como o [Drupal Portugal](#), [Drupal Galiza](#) e [Drupal Spain](#)
- Grupos para organização de eventos, tais como o [DrupalCamp Portugal](#) e [DrupalCamp Spain](#)
- Sites de comunidades, ex, [Drupal Portugal](#)
- Twitter, IRC, Facebook, Google+ e outros



# Introdução: Eco-sistema

O Drupal tem crescido muito, e em todo o mundo [1][2]. Segundo estatísticas, hoje em dia conta com:

- 35000+ Módulos
- 2300+ Temas
- 1000+ Distribuições
- 104k+ Contribuidores (3800 D8)

O mercado de trabalho do Drupal tem **crescido** sistematicamente e cada vez há mais procura.



# Introdução ao Drupal



**“In depth”**

## In depth: Funcionalidades

O céu é o limite da imaginação. A que ponto podemos construir soluções com componentes já existentes?

- *Data modeling* com a criação livre de **campos**
- Criação livre de **listas** de conteúdos
- *Visual modeling* com a criação de **layouts**
- Criação de **regras** com gatilhos, condições e ações
- **Multi-língua** e internacionalização
- **Web services** e acessibilidade



## In depth: Funcionalidades (cont.)

E outras...

- *E-Commerce*, com multi-moeda
- *Newsletter* e diversos
- Integração com / soluções de raiz de CRM, ERP, *Intranet*, etc
- *Data sources* externos, *feeds* e migração de dados
- Exportação e gestão de configurações
- Possibilidade de criar *distros* de Drupal e muito mais!

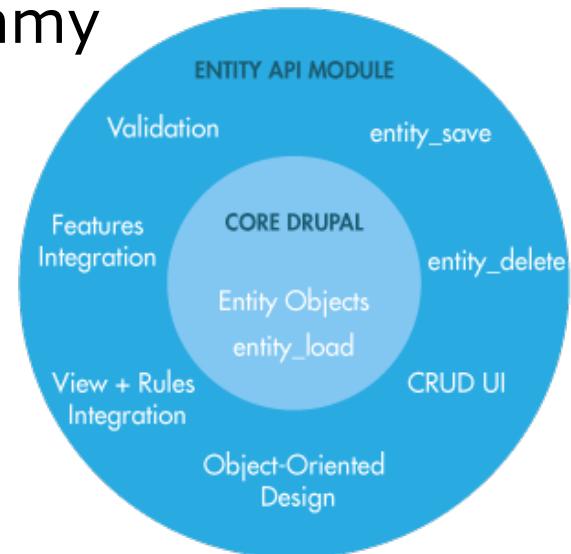


# In depth: Conceitos e terminologia

O Drupal tem alguns termos fora do vulgar [1][2][3]:

- **Entity types**: abstração de **funcionalidade** que serve para agrupar **objetos** semelhantes ou relacionados. Ex: node, user e taxonomy
- **Bundles**: agrupamento de **fields** e **configurações** pertencentes a um **entity type**.  
Ex: node types e vocabularies

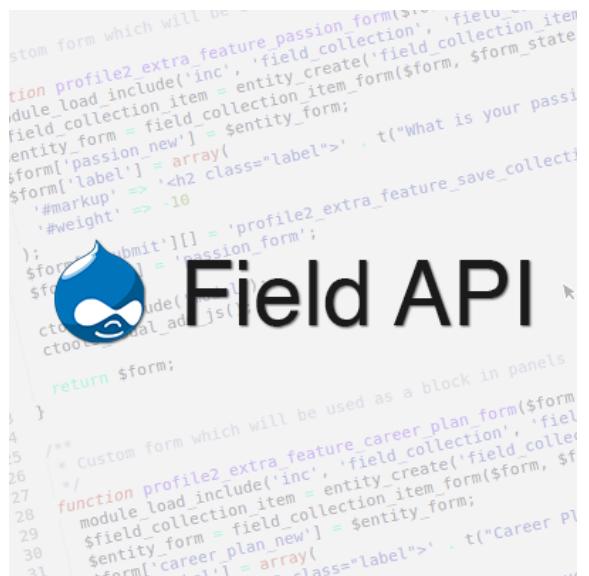
Nota: Existem alguns **entity types** sem **bundles**. Ex: user



# In depth: Conceitos e terminologia (cont.)

- **Fields:** um pedaço de **conteúdo** re-utilizável ou “data type”, que são adicionados a **bundles** ou **entity types**. Ex: body e field\_image
- **Entity:** uma instância de qualquer **entity type** ou **bundle**, ou seja, o conteúdo em si. Ex: conteúdo “hello world”, utilizador “admin”

Assim sendo, qualquer campo pode ser acoplado a qualquer objecto, e estes podem ter relações.



## In depth: O que mudou com o Drupal 7

O Drupal não foi sempre tão genérico e flexível como é hoje em dia. Com a release 7.0, 50+ módulos *contrib* juntaram-se ao *core* [1][2], tornando-o em algo inédito:

- **Entity API:** permitiu-nos criar uma abstração de estruturas de dados
- **Field API:** trouxe o potencial do CCK, Content Construction Kit do *contrib*
- **PDO:** resultou na portabilidade de BDs
- **Cache e APIs** melhoradas



# In depth: O que mudou com o Drupal 7 (cont.)

**Drupal 7 Release Party!**

Home All countries About Drupal 7 Login

Drupal 7 is released!  
And there were hundreds of parties to celebrate

326 parties planned in 96 countries. Add yours!  
Then spread the word on Twitter and don't forget to include the hashtag #d7rp

Mapbox | Terms of Service



# In depth: As novidades do Drupal

## 8

O Drupal 8.0 traz um grande *refresh* do código, com melhoramentos significativos de código e da UX:

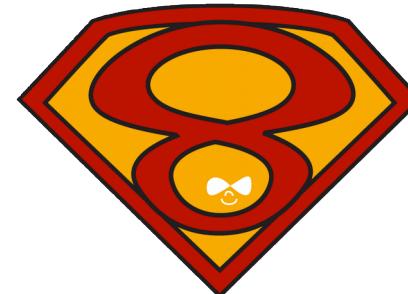
- **Modern OOP:** aplicado em quase todo o *framework*
- **Symphony framework:** substitui alguns mecanismos do core
- **Twig template engine:** substitui o PHPTemplate para simplificar os *themes*
- **YAML component:** a configuração é exportável no novo formato, o YAML



**Drupal™**

# In depth: As novidades do Drupal 8 (cont.)

- **PHPUnit:** *framework* de testes incluída, juntamente com o SimpleTest
- **Unified Entity & Field API:** agora tudo usa a mesma API, incluindo o sistema de menus
- **Translation API:** redesenhada para aproveitar novas ideias e tecnologias
- **Form API:** os formulários foram melhorados e agora todos são editáveis no *core*, incluindo o do registo de utilizadores e edições



# In depth: As novidades do Drupal 8 (cont.)

- **Views:** o módulo famoso do *contrib* juntou-se agora ao Fields API, para máxima flexibilidade
- **Web Services:** incluídos no *core*, a pensar na interoperabilidade
- **Accessibility, Mobile e Web 2.0:** tecnologias como o HTML5, *Responsive design* e *Mobile first* fazem agora parte do *core*
- **UX:** toda a interface foi redesenhada para melhorar a experiência do utilizador



# In depth: As novidades do Drupal 8 (cont.)



Mais uma release party!

# In depth: Requisitos

O Drupal tem necessidades muito próprias. Como tal, não pode ser corrido em qualquer *hardware*, *software* ou configuração. Podem ser consultadas no site oficial.

## Requisitos mínimos:

- Servidor *web*: Apache, nginx ou IIS
- Servidor de bases de dados: MySQL, MariaDB, Percona ou equivalentes, PostgreSQL ou SQLite
- Interpretador PHP



# Introdução ao Drupal

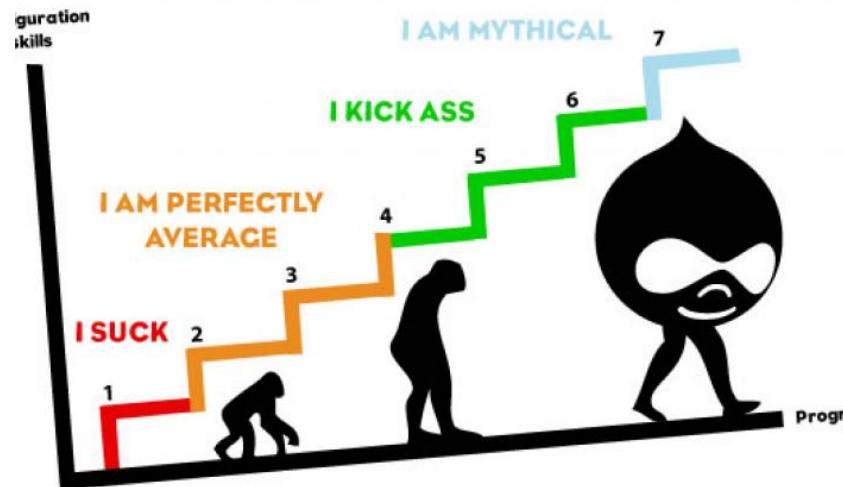


**“Hands on”**

# Hands on: Os primeiros passos

Como começar com o Drupal? É preciso fazer a instalação primeiro. Para simplificar, num *desktop*, podemos utilizar um *stack \*AMP* pré-configurado, que inclui todo o *software* necessário. O *workflow* é simples:

- 1.1) Instalar um *stack \*AMP*
- 1.2) Instalar o Drupal
- 1.3) Configurar
- 1.4) Construir
- 1.5) Desenvolver!



# Hands on: O stack WAMP da Acquia

O [Acquia Dev Desktop](#) é o *stack \*AMP* oficial do Drupal. É a maneira mais fácil de instalar o Drupal num *desktop*.

Características:

- Inclui os mínimos: servidor *web*, servidor de bases de dados, interpretador PHP e o Drupal
- Inclui extras: phpMyAdmin e XMail Server
- Disponível para Windows e Mac OS X
- *Freeware* e fácil de instalar



# Hands on: Instalação

Como instalar? Segue as indicações:

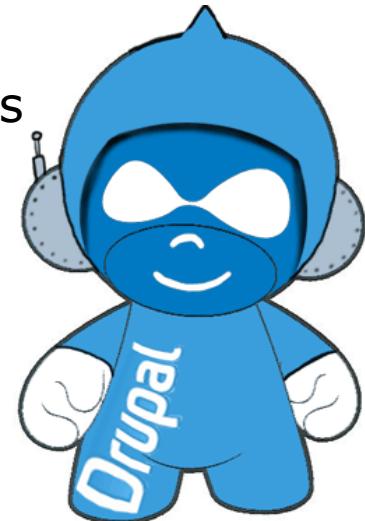
- 1) [Descarrega](#) o Acquia Dev Desktop 7.x com Drupal 7
- 2) Executa o instalador e confirma a licença
- 3) Define o *software* a instalar e o local de instalação
- 4) Define as portas de acesso do *stack*
- 5) Introduz alguma informação
- 6) Carrega em ***launch!***



# Hands on: Overview

O menu no topo permite-nos chegar a qualquer secção administrativa do Drupal:

- **Content e People**: administrar conteúdos e utilizadores
- **Structure**: construir o site, ie, *site building*
- **Appearance e Modules**: escolher módulos e temas
- **Configuration**: configurar parâmetros do sistema
- **Reports**: relatórios do sistema, ex, erros
- **Help**: manual dos módulos do *core* e *contrib*



# Hands on: Configuração básica

/admin/config

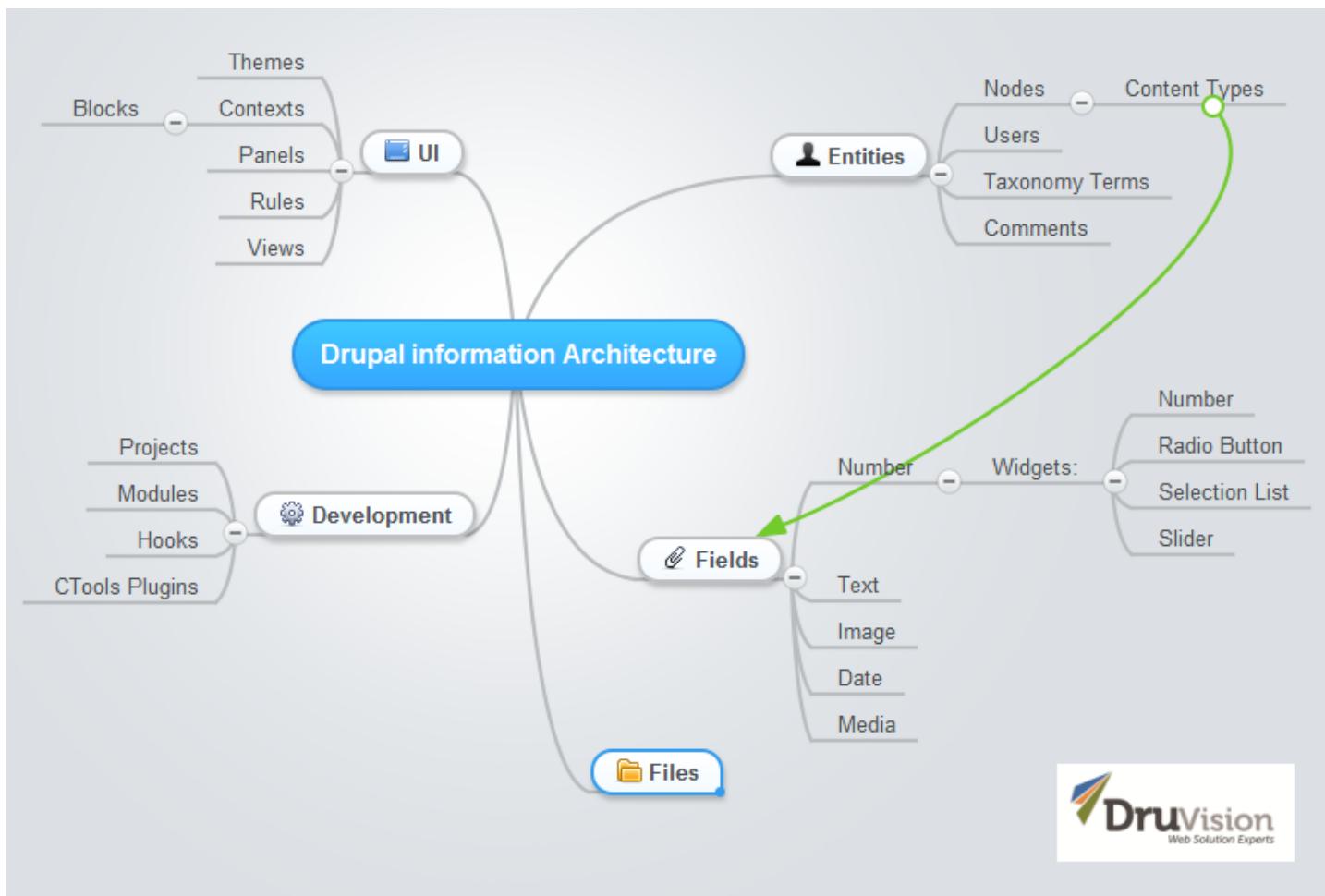
A página de administração contem uma secção para configuração de parâmetros do sistema:

- **System:** nome do site e endereço de email
- **Media:** definição dos caminhos do sistema de ficheiros
- **Development:** configurar as caches do core
- **Regional and language:** línguas e formatos



Para além destas, existem muitas outras configurações e normalmente este é o local onde módulos *contrib* colocam as suas páginas de configuração.

# Hands on: Up next...



# Hands on: Nodes, os objetos conteúdo

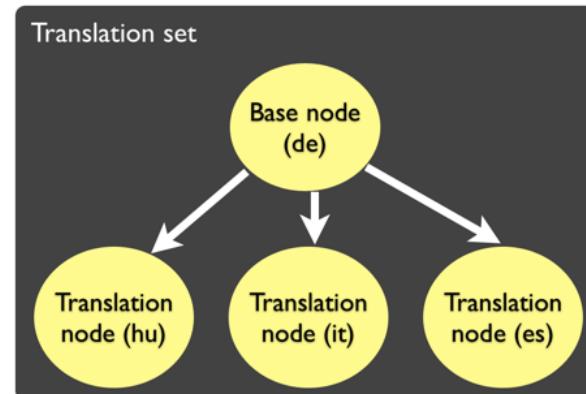
/admin/structure/types

A página de administração de tipos de conteúdo é o local onde podemos criar configurações de conteúdo:

- **Node type ou Content type**: armazena configuração relativa a **nodes**, aos quais também são agregados “date types”, em **fields**
- **Node ou Content**: armazena **conteúdo**, e também é um nome de um módulo do *core*

Os **node types** permitem-nos fazer a diferenciação de conteúdo, como um registo de *blog* e um *banner*.

Os **nodes** são também páginas, e estão sujeitos a traduções e a outras propriedades, como as revisões.



# Hands on: Fields, os campos dos objetos

/admin/structure/types

Cada tipo de conteúdo pode ter campos associados.  
Na verdade, qualquer tipo de entidade pode ter campos:

- **Field**: armazena um determinado “data type” e sua configuração.
- **Field Display**: armazena a configuração de visualização do **field**.
- **View Mode**: define um configuração de visualização de uma **entity**, à qual estão associados **field displays**.

Cada **field** no Drupal pode ter uma configuração de visualização para cada **view mode**.  
Ex, Print, PDF, Full, Teaser, etc.



# Hands on: Taxonomy, a arte de catalogar

/admin/structure/taxonomy

A página de administração da taxonomia é o local onde podemos criar ou associar blocos existentes a regiões:

- **Taxonomy**: a arte de catalogar objetos, é na prática um nome de um módulo do *core* onde se faz a gestão de **vocabularies** e **terms**
- **Vocabulary**: armazena configuração relativa a **terms**
- **Term**: armazena a **categoria** e outros dados

Um **vocabulary** é nada mais que um conjunto de **terms**. Tal como os conteúdos, os **terms** também são traduzíveis e podem opcionalmente ter **fields**.



# Hands on: Views, as listas personalizáveis

/admin/structure/views

A página de administração das visualizações é o local onde podemos criar listas de objetos:

- **Views**: é o nome do módulo contrib onde se faz a gestão das **views**
- **View**: armazena a configuração de uma **visualização**
- **View Display**: configuração de uma **variante** da **view**

Exemplos: listas de utilizadores, conteúdos, categorias, produtos de loja, etc. Para discerni-los são usados **filtros**, **ordenamentos**, **contextos**, **argumentos** e outras propriedades, que podem ser fixas ou dinâmicas.



# Hands on: Menus e ligações

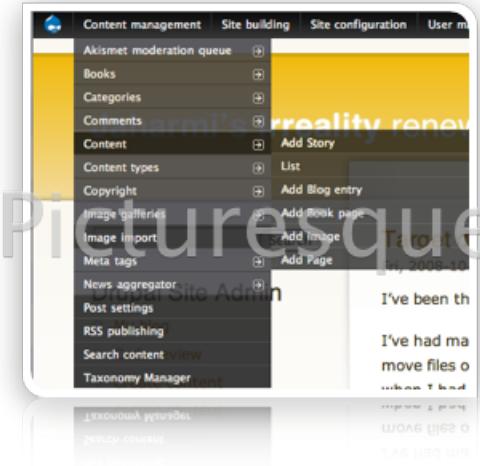
/admin/structure/menu

A página de administração de menus é o local onde podemos criar novos menus e ligações:

- **Menu**: objeto para armazenamento de conjuntos de **menu items**, e também um nome de um módulo do core
- **Menu item**: armazena uma **ligação**

Um **menu** é declarado como um **block**, e pode ser utilizado nos temas, no sistema de blocos ou em módulos compatíveis do mundo *contrib*.

Os **menu items** também estão sujeitos a traduções.



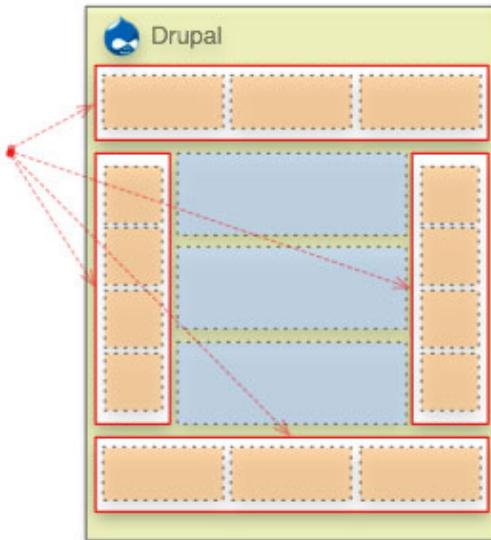
# Hands on: Blocos e Regiões

/admin/structure/blocks

A página de administração de blocos é o local onde podemos criar ou associar blocos existentes a regiões:

- **Block**: objeto especial para representação visual no site de **algo** produzido por um módulo e também um nome de um módulo do *core*
- **Region**: representação de uma **secção do tema**, aos quais são associados **blocks**. Ex: header, footer

Os **blocks** são criados na UI com HTML ou programados em módulos. As **regions** são declaradas no tema e inseridas nos *templates*, com recurso a HTML e PHP.



# Hands on: Utilizadores e Permissões

/admin/people

A página de administração de utilizadores é o local onde podemos gerir utilizadores, e não só...

- **User**: objeto para armazenamento de **credenciais**, e também um nome de um módulo do core
- **Role**: agrupa **permissions**, são utilizadas em associação com **users**
- **Permission**: definição de uma **permissão** individual para determinada função em determinado módulo

É através dos **roles** e **permissions** que se faz toda a gestão de direitos administrativos no Drupal.



# **Hands on: Let's have fun!**

## **Exercício:**

### **Site de notícias**

- Categorias temáticas
- Conteúdo notícias
- Lista de notícias
- Perfil de autor



# Introdução ao Drupal



## Créditos

# Créditos

Hope you enjoyed!

Este trabalho está licenciado com a licença Creative Commons Attribution-ShareAlike 4.0 International.

<https://github.com/lpalgarvio/drupal-training>



# Créditos

Luís Pedro Algarvio

[lp.algarvio@gmail.com](mailto:lp.algarvio@gmail.com)

<https://lp.algarvio.org/>

Fim

