



**DRUPALCON  
SYDNEY**

Business and Strategy | Mori Sugimoto | FEBRUARY 7 2013

# **Benefits and challenges of Drupal Adoption for Multinational Organisations**

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

# About me



**Name: Mori Sugimoto (dokumori)**

Using Drupal for: 6 years

Based in: London, UK

Nationality: Japanese

Works for: Acquia

# About Acquia



- Expert 24/7 Drupal support
- Optimised Drupal hosting
- Foster Drupal adoption



# We are hiring!

[acquia.com/careers](https://acquia.com/careers)



# About you?



# About you?



- What do you do for your job?

# About you?



- What do you do for your job?

# About you?



- What do you do for your job?
- What's your experience with Drupal?





# **This presentation is about:**

**Drupal adoption for  
multinational organisations**



# This presentation is **NOT** about

## multi*lingual* Drupal

# Outline:



# Outline:



- Benefits of adopting Drupal



# Outline:



- Benefits of adopting Drupal
- Challenges and solutions



# Benefits of adopting Drupal



# **...or disadvantages of NOT adopting Drupal**



# A typical scenario:



# Each country office has its own website



NZ

Australia

UK

Malaysia

Singapore

Hong Kong

Japan

USA

# Each country office has its own website



# Each website is built from scratch



1. Brainstorming
2. Requirements definition
3. Design
4. Development
5. Content population
6. Various tests
7. Launch

+100 meetings

+500 emails

(of which only 1/20 are  
actually read)

} \* *n*

*n* = # of offices

# If you are using proprietary CMSES



## License fees \* $n$

$n$  = # of offices





**Building each site from  
scratch and / or  
on different CMSes is  
  
inefficient**

Step in further



# Problems of having multiple CMSeS

Step in further



# Problems of having multiple CMSES

Poor reusability

Step in further



# Problems of having multiple CMSES

Substandard quality  
= low maintainability



Step in further



# Problems of having multiple CMSES

Multiple bugfix / security updates

Step in further



# Problems of having multiple CMSES

## Trainings for editors



**Maintaining multiple  
CMSes is  
  
inefficient**

# The cure: Build all the sites on a Drupal-based platform





# The cure: Build all the sites on a Drupal-based platform



**Drupal<sup>TM</sup>** - based platform

# The cure: Build all the sites on a Drupal-based platform



UK	NZ	Singapore	Australia
Hong Kong	Japan	USA	Malaysia

**Drupal<sup>TM</sup> - based platform**



# How can Drupal make it simple?

# Drupal can simplify it





# Drupal can simplify it

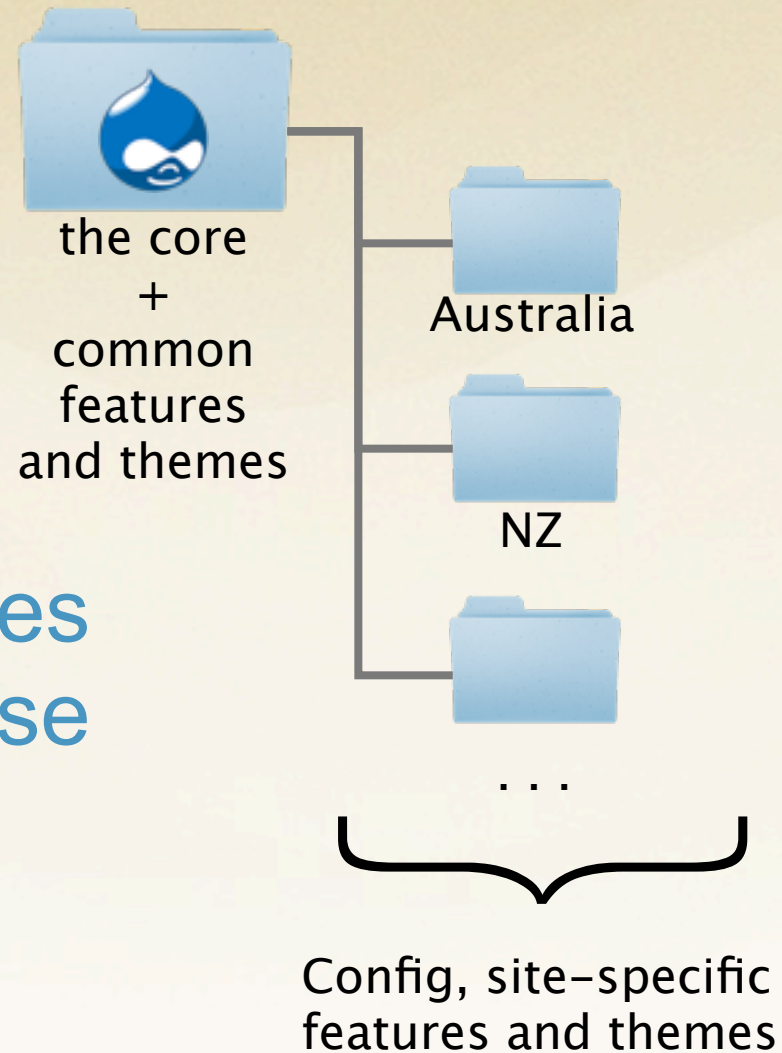


Drupal lets multiple sites  
share a single codebase

# Drupal can simplify it



Drupal lets multiple sites  
share a single codebase





# Benefits of a Drupal-based platform



# Benefits of a Drupal-based platform

High reusability





# Benefits of a Drupal-based platform

Consistent look-and-feel



# Benefits of a Drupal-based platform

Rapid construction of new sites



# Benefits of a Drupal-based platform

Lower maintenance overhead

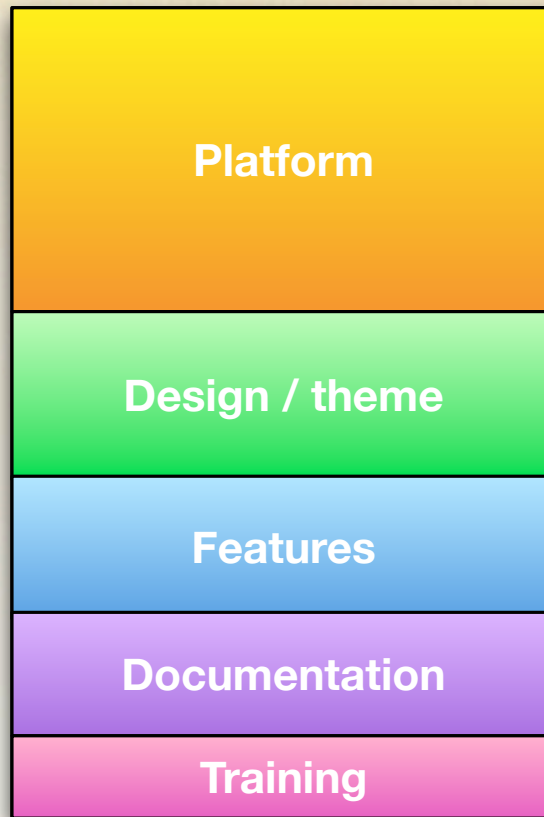


# Benefits of a Drupal-based platform

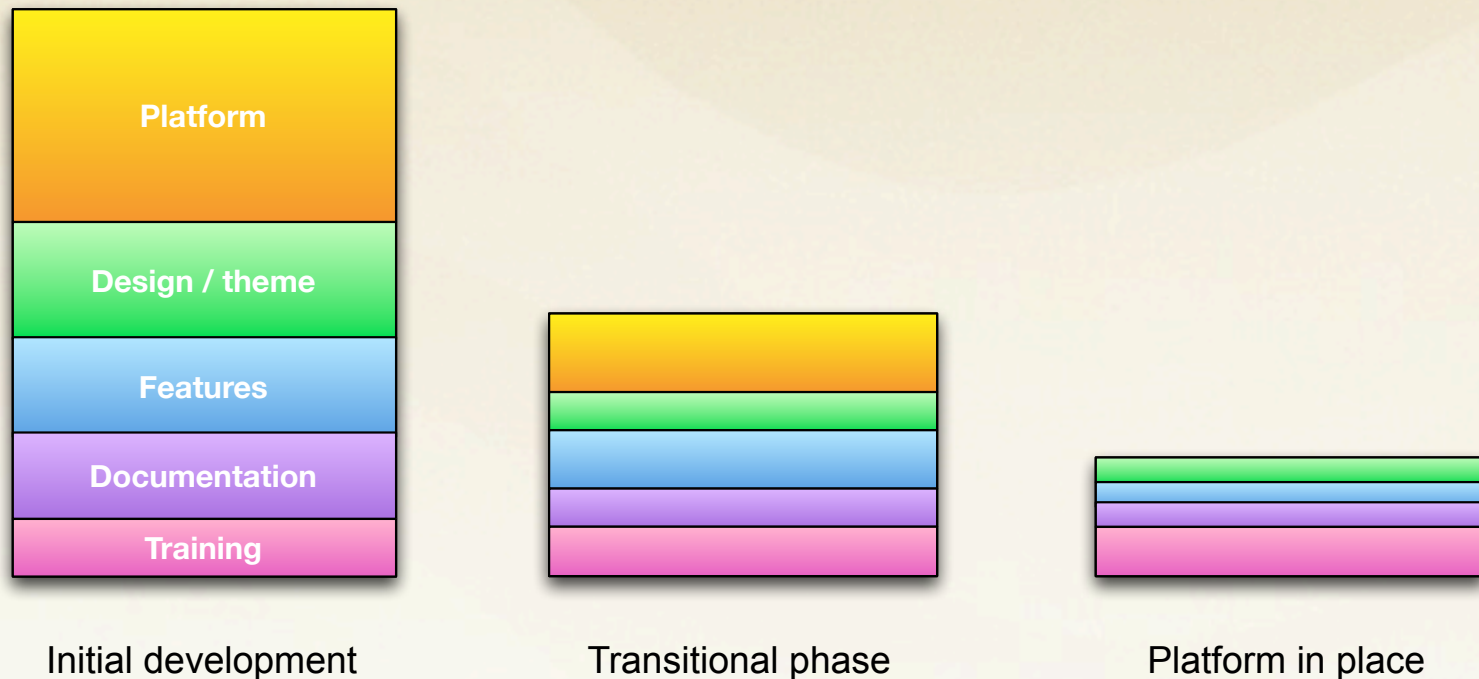
Simplified hosting requirements



# Cost efficiency



# Cost efficiency



The ratio of costs required for new sites



# Improved efficiency

## Better ROI



**Lovely. But is it easy and  
straight-forward?**



# Lovely. But is it easy and straight-forward?

Technologically yes.  
But there are other challenges.



# Challenges in different phases



1. Preparation

2. Development

3. Operation and maintenance



# 1. Challenges during the preparatory phase

- Consensus
- Selecting the right vendors



No one vendor is capable of  
developing all the sites

# Challenge:

# Consensus within the company

# Consensus



Each country office is autonomous

# Consensus





# Consensus



- Share the image of success

# Consensus



- Share the image of success

# Consensus



- Share the image of success
- Understand restrictions

# Consensus



- Share the image of success
- Understand restrictions

# Consensus



- Share the image of success
- Understand restrictions
- Understand the benefits





No one vendor is capable of  
developing all the sites

# Challenge:

# Selecting the right vendors

# Vendors



**The vendors must be  
highly competent**

# Vendors



**Main: builds / maintains the platform**

# Vendors



**Main: builds / maintains the platform**

# Vendors



**Main: builds / maintains the platform**

- Adequate resources



# Vendors



**Main: builds / maintains the platform**

- Adequate resources
- Highly capable in managing projects

# Vendors



**Main: builds / maintains the platform**

- Adequate resources
- Highly capable in managing projects
- Highly capable in managing communication and expectations

# Vendors



**Local: builds local sites**

# Vendors



## Local: builds local sites

- Experienced, standard-compliant



# Vendors



## Local: builds local sites

- Experienced, standard-compliant
- Good communication





# Ensuring vendors got what it takes...



# Ensuring vendors got what it takes...

- Clearly define a set of acceptance criteria for the end product



# Ensuring vendors got what it takes...

- Clearly define a set of acceptance criteria for the end product
- Request for references



## 2. Challenges during the development phase

- Project management
- Maintaining quality



# Challenge:

# Project management





# Challenge: Project Management

- Complex organisational structure
- Multiple stakeholders in multiple dept.
- Conflicting interests / requirements



# Challenge: Project Management

## Suggested solutions:

- Simplify communication
- Clarify responsibility
- Review frequently
- Provide better transparency
- Collaborate



In one word:

**Agile**



# Maintaining quality





# Challenge:

# Ensuring scalability



# Challenge: Ensuring scalability



- Loosely coupling features
- Minimise site-specific, tightly coupled features

**Reference:**

<https://www.sandcamp.org/features-they-work-youre-just-doing-it-wrong-or-loosely-coupled-features>



# Challenge:

## Ensuring high maintainability



# Challenge: Maintainability

Adhere to Drupal best practices

- Coding standards / commenting



# Challenge: Maintainability

- Configuration in repository
- Code driven development

## References:

<http://drupal.org/project/features>

[http://api.drupal.org/api/drupal/modules%21system%21system.api.php/function/hook\\_update\\_N/7](http://api.drupal.org/api/drupal/modules%21system%21system.api.php/function/hook_update_N/7)

<http://archive.org/details/CodeDrivenDevelopmentUsingFeaturesEffectivelyInDrupal6And7>



# Challenge: Maintainability



Control vendors' access to the repo





# Challenge: Maintainability

Control vendors' access to the repo

- Github - pull requests



# Challenge: Maintainability

Control vendors' access to the repo

- Github - pull requests
- Gitolite - directory access



# Challenge: Maintainability

Control vendors' access to the repo

- Github - pull requests
- Gitolite - directory access
- Gitlab - merge requests

# Challenge: Maintainability



**Avoid maintaining  
multiple versions**



# Challenge: Maintainability



- Automated tests / CI
- Site audit



# Maintainability



Sufficient documentation



# Challenge:

# Security



# Challenge: Security

- Regularly check for common errors in code and configuration
- Security audit



# Challenge: Performance

# Performance





# Performance



- Common sense stuff:

# Performance



- Common sense stuff:
  - Internal caching (e.g. Views)

# Performance



- Common sense stuff:
  - Internal caching (e.g. Views)
  - External caching (e.g. Varnish)

# Performance



- Common sense stuff:
  - Internal caching (e.g. Views)
  - External caching (e.g. Varnish)



# Performance



- Common sense stuff:
  - Internal caching (e.g. Views)
  - External caching (e.g. Varnish)
- Perform a load test on the entire platform as a new site launches



# Many of the checks can be automated...



# Many of the checks can be automated...



## So we did!



<https://insight.acquia.com/instant-insight>

# 3. Challenges during the operation / maintenance phase



- Change, release and deploy management (governance)
- Support



# Challenge: Governance

# Governance





# Governance



- Change management

# Governance



- Change management

# Governance



- Change management
- Release management

# Governance



- Change management
- Release management

# Governance



- Change management
- Release management
- Deployment management



# Governance



- Use RACI to clarify roles and responsibilities

RACI = Responsible, Accountable, Consulted, Informed

Example:

Category	Task description	Customer SA	Customer LO	Main Vendor	Acquia	Local Vendors
Guarding the CMS codebase	Accept pull requests	A		R	C	I
	Push the code to Acquia Cloud	R, A		I, C	C	I
	Deploy	R, A	I	I	C, I	I
	Test (dev)		A	C	C	R
	Test (stage)	I	R, A	C	C	I
	Test (prod)		R, A	I, C	C	I



# Challenge: Support

# Support



# Support



- Application-level support for editors

# Support



- Application-level support for editors



# Support



- Application-level support for editors
- Support for technical issues

# Support



- Application-level support for editors
- Support for technical issues

# Support



- Application-level support for editors
- Support for technical issues
- Support for hosting environment

# Support



- Application-level support for editors
- Support for technical issues
- Support for hosting environment

Make the reporting process simple for users



# Support



**Support should be  
comprehensive**





# Finally...

# Look out for:



- Legal requirements in different countries
- Latency



**That's all.**



**Thank you.**



# Questions?

**Feedback:**

<http://sydney2013.drupal.org/benefits-and-challenges-drupal-adoption-multinational-organisations>