Another Usability Proposal

(let' s not care about users; let's care about developers)

Usability for Drupal

Contents

- Usability in Open Source (same problems all over)
- Usability in Drupal (some experience)
- Different solution: using our nature in our advance.
- How?

Usability and Open Source.

Nature - most projets - distributed - democratic - anarchistic

Hence - no coordination - successes mostly backend: apache, PHP, Linux etc.

Important is the nature of dev teams. Firefox and OpenOffice come from classic dev-teams.

Not **Open Source** is the problem,
the **nature of the project** is.

Some of the issues and problems are...

Every UI improvement gets huge discussions.

result: hardly any, frustrated UI developers and inconsistent solutions.

Volunteers take only one piece to improve.
Result: inconsistent UI;

Human Interface Guidelines (HIG): too hard to maintain. Improssible to enforce.
Result: unfinished, old and unused HIG.

Usability experts don't feel at home in development environments.

Nor should they.

Usability happens in a different **ecosystem**.

Give usability improvement its own **environment**.

Back to Open Source usability. GNU/L**inux**

- GNU/Linux is just a **kernel**. Usability is entirely different there: it's **technical** not UI
- Ubuntu is a **Distro**, using a.o. Linux. Lots of **UI**. Lots of usability

Ubuntu can afford to focus on home-users, disregard corporate, heavy server and/or embedded apllications.

No need to consider "**but what about running ubuntu on TomTom navigation devices**".

Drupal has the same problem: lacks focus.

you, on the other hand, have that focus. (**for your own/clients sites**)

Another problem:
I sell services.

- "If Drupal becomes too usable, I will sell less."

As a developer reward is negative for improving usability.

However: **I do care**.

Not For **John Doe Anonymous Blogger** to Use Drupal, but for **Wendy my Wealthy Client** to use my configuration.

Drupal can do this. **Distros.**

But...

Distros help, but solve only one problem in usability area: **Installation and configuration.**

important, but not important in daily use. Ditro's only help John Doe Anonymous Blogger to **get going**.

Joe Shmoe must be able to **use** a site.
And be happy when doing so.

Proposal:

Make it **easy to develop good usability**
(for Developers).

Developers make it **easy to use for users**.

Give usability improvement **its focus**.

Example: Corporate Brochure Site About Us.

- Book module is fine, yet putting in menu should be automated - menu follows book struct.
- Is true for corporate sites, but not for wiki- or documentationo or large-volume implementation. For them such features are "in the way".

Off course solution that solves issue for both is possible.

But that only adds complexity, makes a really simple issue a complex, hard one.

Looses focus: maintaining two types of UI often means having two imperfect UIs.

Proposal: The developer (or distro maintainer) implements extra, focused module (bookmenu.module).

That way, Drupal book module becomes general outliner/hierarchical content system.

Implementors create exactly **the UI they need** in their specific case.

When the case is covered for Wendy the Wealthy Client,

John the Anonymous blogger his case might be covered too.

OpenSource at the Rescue.

How?

Enabling developers to develop better interfaces for their **specific** cases.

Enabling you to create a usable site for your clients, sites or users!

“Usability experts” can then:

- Develop widgets (case by case)
- Implement widgets (for their case)
- Improve widgets (case by case)

How? and Where?

Here I would like some of your feedback.

Library?

- elements.module
- YUI elements
- Snippets

In Distro's?

On a subsite?

On Several offsites?

Let us try to find out how to:

enable contribution and improvement of widgets?

allow people to focus.

...you to build a perfect site for your case.