

GT2003HackathoncocoonCms

Raw notes taken collaboratively during the talk.

See also [Sylvain's weblog](#).

Stefano on the quest for Cocoon's own Content Management

Introduction

How to have a Cocoon back-end which writes to a repository, and a front-end which is compatible with ASF architecture (i.e. ideally static pages).

Possible scenario

- Static pages on ASF sites
- Each page has an "edit this" link, pointing to `edit.cocoon.apache.org` which is a live Cocoon instance
- Edits cause mail to be sent to the list, including CVS diffs or something similar
- Committers review and incorporate edits

Requirements

- Accountable docs, like today from CVS
- Easy editing, mouse-less
- Easy to add images
- Metadata, page history, workflow
- Traceability of edits, who did what
- Offline editing must be possible (as currently for code), for example allowing Wiki-like input format as a secondary format. Ideally, bidirectional xhtml-to-wiki conversion
- Non-committers must be able to add new documents easily, requiring a doc-committer to check the submission

How does Lenya fit into the picture

Michael Wechner: sees a new (Lenya?) document type made out of main content and annotations. Committers or reviewers would take the annotations and integrate their content into the main content.

This project could be a good way for Lenya to get more attention from the Cocoon community.

Community aspects:

- lower the bar for contribution to the community
- lower the bar for recognition by the community

Learning objects

Future thoughts...

A learning object has

- Prerequisites: what you need to understand it.
- Knowledge: what you get out of it, which concepts it helps you understand.

Unstructured knowledge base, learning objects connected by their "synapses".

Trail view

A trail is a learning object itself, created by authors.

Highlighting paths through learning objects - need to visit an object from at least two different paths to actually understand it.

This concept can be applied externally to the documents themselves. This is useful because the documents we are making contain little or no meta-data or structure (because we want them to be easy to edit).