

Positioning and goals

- Daisy has been developed from the beginning to support a broad range of content management needs. Based upon the core repository server, many front ends and publishing systems are possible. Currently we have:
 - the Wiki. Through the extension mechanism and publisher requests, arbitrary custom publishings are possible, see e.g. the RSS feeds.
 - the book publishing: background generation of nicely structured and formatted 'books'.
 - but no real solution for websites (especially public, high traffic sites), though if needed today the use of a custom skinning and possibly wget gets you quite far.
- we need to further look into decoupling the publishing infrastructure from wiki
- we need more powerful publishing possibilities, for example:
 - more flexible publisher requests: e.g. when publishing a document of type X, include a query that retrieves related documents based on the value of a field. More general, allow to use specific publisher requests depending on properties of the document (also for included documents). -- **done in Daisy 1.5**
 - more control over URL space by allowing to mount sites at arbitrary nested paths (including just "/").
 - directly linking to part content for certain document types (e.g. Attachments) -- **done in Daisy 1.5**
 - make use of 'ext/' prefix for extension URLs optional
 - make inlining of content of non-Daisy-HTML parts possible -- **done in Daisy 1.5**
 - inclusion/embedding of multimedia: -- **done in Daisy 1.5**
 - flash
 - movies
 - custom HTML (for more interactive content that doesn't need flash, such as imagemaps, something javascript driven, ...)
 - documents that combine other documents, queries, etc. to publish as one web page (which is possible today using custom publisher requests, but then geared towards composition by content authors)
 - better staging/live separation: e.g. make collection membership and document retired also versioned.
- match between publishing possibilities in the wiki and publish-only so that authoring environment can closely match publishing environment (as far as makes sense, and as preferred)
- better support for editing in a "staging" environment: currently the Daisy Wiki defaults to displaying the 'live' versions of everything (including navtrees & query results), maybe a site-level option to work with 'last' versions would be a solution. -- **done in Daisy 1.5**
- "easy enough" styling/templating: shouldn't require a computer science degree.

Currently we are investigating

1. static site publishing

2. more advanced publishing possibilities in general (for wiki, static publishing, ...). See e.g. items listed above.

Further documents will go into the feature and implementation details of these two items.

Fields

Name	Value
Category	None