

Evergreen Documentation Issues (May, 2009)

- Much documentation is lacking and the Dokuwiki presence is a mess.
 - “How things work” trapped in the developer brain trust—subject to departures, retirement, plane crashes, etc.
 - Time-consuming for newcomers to figure out basic information
 - Using email lists to ask basic questions is not an efficient use of the querent's or list readers' time
 - Content added whimsically, not maintained, not always in wiki format
 - Reflects poorly on Evergreen project — “first face forward”
- Need a clear process for getting documentation added/maintained.
 - The "Procedures and conventions for contributing to the Evergreen project" document needs top-to-bottom review and revision
 - The structure it outlines is not being maintained.
 - The process described in the "Documentation Contribution" section does not match reality.
 - Guidance is lengthy, contradictory, developer-focused, and intimidating to newcomers
 - The bottom line is that the current system has not resulted in the creation or maintenance of adequate quantities of high-quality documentation

Core ideas

- Big-picture idea: establishing sustained, formal leadership, planning, and energy related to documentation.
 - Suggestion:
 - Establishing a small documentation committee responsible for...
 - Organizing the documentation that is out there
 - Identifying holes for official documentation
 - Coming up with processes for evaluating and accepting documentation into the main project (“core” versus “community”)
 - Clarifying licensing requirements for open source documentation
 - Recommending (and writing) procedures
 - Recommending, implementing, and enforcing documentation standards and toolsets
 - Communicating among the projects
 - Advocating for funded documentation projects
 - Or appoint a rotating Documentation Czar/ina who would maintain the structure of DokuWiki, making it clearer where community-written documentation could be plugged in/linked to.

Related Ideas

- Evergreen wiki/website needs community adoption/ownership
 - Need statement of IP
 - Need a “throat to choke” for updates
- Making documentation a priority in development.
 - Possible response: pairing a volunteer community member with each piece of development, and have the programmer and documentation communicate with each other throughout the development process.
- Making it simpler for people to contribute
 - Design a contribution matrix that makes it easy for people to contribute small amounts of good draft documentation
 - Have editors up the chain for editing/revising/reviewing/testing
 - Have a small, dedicated group of committers
- Possible goals for the site
 - Establishing access rights for documentation
 - Centralizing documentation location — one site, well-organized
 - The structure that is there is not clear, nor is it explained.
 - Need an agreed-upon structure that people can plug into.
 - Need central locations for “other” info — community contributions, historical information, etc.

Example Needs Document:

Basic documentation needs for libraries evaluating or migrating to Evergreen

- A good, general overview from a data perspective
 - form
 - This could be an image: entity-relationship diagram, or other high-level view.
 - This could be structured text: schema, data dictionary, etc.
 - I was glad to find the schemas, but they are basically undocumented, and therefore not as helpful as they could be.
 - basic information on
 - system structure
 - record types
 - fields
 - code lists
 - indexes
 - record linking
 - policies
 - templates
 - character encoding
 - display
 - data import
 - constraints on the above
 - how the above can come into the system (system-supplied? loadable? hard-coded? manual input?)
 - which of the above are modifiable, and how (manually individually? batch edit? overlay? developer only?)
 - normalization
 - validation
 - error handling
 - how various record types and fields are used in the system
 - examples = organizational unit, reference flag, tcn
 - which settings are settable in the staff client
- clearly written profiling forms
- account info for dev server
 - The download page tells which staff client to install to use with the dev server. That page has account names and passwords for the demo server, but not for the dev server. It took me quite a long time searching around on the Evergreen site until I finally found the info in a listserv message.
- recommended Linux distributions, and which versions of those distributions
- help inside the staff client