

Recommend that relevant parts be [merged](#) into the [TikiWiki Manual of Style](#) page and this one [deleted](#).

The discussion on this page pertains mainly to Documentation Structure actually.

MoS page should cover the details and fine points, can be expanded to cover the proper use of inpage menus, screen shots, faqs, etc.

doc site needs some major css teaking. heading levels after 2 should be indented, h2 should be smaller, perhaps including a horizontal rule and edit by section.

Templates are/should be the major tool for style control. Registered users should be able to punch a button and add (below any existing content) a preformatted chunk of content.

### **Thacker's Laws of Documentation.**

1. Answer **my** question. Do not make me wade through reams of extraneous material
2. Optimize my searches. Get me as close to the right entry as you can with my keyword
3. Keep the pages short, do not make me scroll from Minneapolis to Dallas for an answer
4. Use clear, concise, business quality language.
5. Do not editorialize. Write from a Neutral Point of View
6. Always give an example
7. Screenshots rule

smaller, more targeted pages are preferable over large monolithic pages. They are easier to maintain, categorize, and will serve our target audience better. Since a page can belong to more than one structure, they can be re-used where necessary.

### **Laws of the knowledge base**

- nobody talks about knowledge bases, the point is only having one. (doc? dev? org?)
  - without a common vocabulary, you can't communicate.
1. define all significant terms
  2. related link to the terms to the objects/concepts
  3. related link to from the objects/concepts to the advanced objects/concepts.
  4. inline link back to terms, always.
- don't force structure on the learning (except to)
  - create lists of categorized objects.
  - the specific must always link to the general.
  - use hard redirects and (soft) disambiguators to restrict the vocabulary (did you mean?)

please explain a soft disambiguator?

- if someone linked it, it should exist. (do not remove open links, fill them in)
- knowledge is built always as a pyramid, from the broadest foundations (first) to the pointiest details. Teaching the details without the foundations is a mutual pain.

Sounds good. What's the bottom of the tiki pyramid

- (accurate) terminology matters.
  - the biggest problem i have had with learning trackers, i reckon, is that they are misnamed. They are a Forms and Reports feature.
- programmers (experts in general) are often the least helpful explainers. (fishes that can't see the water)

## **Content Segmentation. (proposal)**

Feature (for end users)

- overview (what and why)
- related links
- instruction (how)
- examples

- [faq](#)

Faq gets flushed back into instruction periodically?

Feature Admin (for configuration options and maintenance)

- [options overview](#) (what and why)
- [related links](#)
- [instruction](#) (how)
- [examples](#)
- [faq](#)

Feature Dev (for developers)

- [development status](#) (what, and when)
- [related links](#) (include developer userpages)
- [requirements and specifications](#) (exactly how)
- [code](#)
- [faq/bugs](#)