Status/RoadMap

- With the Tiki 1.7 Release a new feature of help links from the Admin section was delivered that are hardcoded to the corresponding pages on tiki.org. These links refer to many pages on tiki.org and represent core knowledge about the use and configuration of a Tiki system. While this is a fantastic step there are some potential problems with hardcoded links to any site in the long-term.
  - External usage of the tiki.org site
    - Bandwidth
    - Local usage could be faster due to network problems
    - tiki.org might be unavailable due to network, hardware, or software problems help would then be completely unavailable to the local site
    - The Tiki site is updated to a new version and the local site is still on an older version the docs are then potentially incorrect.
  - Intranet inability to connect for help/doc pages
  - Local Admin cannot edit these pages to reflect local information and choices
  - Format and style don't match originating site
  - Others (*please add any others not listed*)

- These problems already exist and must be addressed fairly rapidly. With some condideration to the future uses of this type of information (as it will only grow in the future) should be undertaken and then addressed. The base requirements are:
  - Local storage of these help (and maybe Doc) pages on a per site basis
  - Ability to update these pages as new features are developed and delivered (excluding CVS)
  - Local admin must be able to edit and update these pages based on local usage or to append notes on their configuration
  - Ability for the local site on a permission basis to create completely new pages that would perhaps have localized help for that site and stor them within the same table
    - Their changes should never be deleted unless they delete them
  - Ability to transfer Locally developed or expanded help pages from one local site to either))tiki.org((or to another local site
  - Ability fo a local site to subscribe to help/doc page updates from another site (either))tiki.org((or to another local site)
  - *Optional* - Some way to establish a cron job that would automagically keep the help/doc pages up to date on a regular schedule by only updating the pages that have changed
  - Others (*please add any others not listed*)

Who is working on what? (Priorities/goals/majors issues/roles)

- AlBrown is the working editor for this requirement and will attempt to assemble document/detail and achieve consensus for the implementation of this RFC/RFE.
  - All are welcome and encouraged to contribute.
  - This page will be where the RFE is developed and the requirements defined.
  - This RFC/RFE has some important long-term ramifications and must look into the future of the overall system.
  - When a near final requirement is achieved it will be submitted their review and may be amended to either clear understanding or revised to comply with what is possible to:
    - the senior devs
    - project management
    - the devel mailing list for the Tiki
  - After initial review of the final draft the RFE will be submitted
  - Project Management will prioritize and we will attempt to get a develoer (or few) to implement

1. Currently (02 Aug 2003) this should be concidered a Request for Comment (RFC)
2. The working ideas will be gathered together for a review prior to development, formalization and

submission of the RFE to reduce the coding effort and to ensure the long-term viability of the implementation.
3. No RFE has yet been developed or submitted
4. Others (*please add any others not listed*)

TikiGroup

Please Link your UserPage.
AlBrown
mose

Competition and standards

Typo3 uses "Extensions", which you can install/uninstall with one click — very modular. This is how they deploy their help system components along with all other sections and add-on features. It's very easy to use and I believe, worth a good close look. -colorado

There are no other competing products for this type of function that I know of. If you know of something please add it here and provide some detail about how and what they do this and if possible a link.

Distributed and P2P techniques
Author: SEWilco 2004-04-19 (everyone else feel free to chip in)

Distributed Wiki

- Files could have TikiWiki.Org signatures created by TikiDocGods.
  - These describe and checksum files so authorized files can be identified.
  - The signature files can then be distributed P2P, as the associated data files can be.
  - File identifier (ie: TikiWikiHelp:TikiSyntax) may be base, or might include version numbers.

- P2P tools such as BitTorrent only require such descriptive information and access to participating servers.
  - P2P type of networks reduce bandwidth demands.
  - P2P methods where anyone can create a server also allows importing to LANs, by having one peer able to import from "outside", then making those files available "inside" to others on the LAN.

- Web searches for **distributed wiki** show many existing proposals which can be tapped.
  - Are there any existing distributed Wiki tools?
- http://dmoz.org/about.html calls itself a distributed directory.
- Searching for DMOZ found Directory>>Social Responsibility>>Projects which includes DMOZ.
- http://rdf.dmoz.org/ describes DMOZ dump format.
  - DMOZ uses a few large files, including ones of many megabytes. Smaller files are better for dynamic use, so a new site can have quick access to just the data being requested when clicking on Help icons.
  - DMOZ uses RDF, a relevant web standard for information exchange by applications.
    - RDF spec at W3

- Suggestions
  - Use RDF to describe pages and groups of pages (so a group can be downloaded when desired).
  - Include signatures for positive identification of approved RDF.
  - Signatures should include non-variable information, with multiple

signatures for subsections.
- Support multiple approvers.
  - TikiWiki.Org would have TikiDocGods which could issue approvals for its Help and Doc pages.
  - **Others might issue their own information, whether alternate Help or other Wiki/Articles.**
  - Make the tools general, so multiple P2P distributed Wikis can used as desired.
- The author of an RDF could sign the entire RDF, except for an Additional Approval section.
- Additional Approvers sections would allow others to reuse existing files and merely indicate with their signature of the RDF that they accepted it.
- Initial discussion implies TikiWiki.Org would be issuing all articles, but it is possible that outside articles might be accepted by simply adding Approvals.
  - One RDF may have only TikiWiki.Org signature, while another may have added another Approval for its use in another collection of docs.
  - Site accepting TikiWiki.Org RDFs could accept one which also has approval from a different Authority, etc.
- P2P webs should distribute only items for Authorities which member sites recognize.
  - One site might be in many webs, with P2P to/from TikiWiki.Org web as well as others.
- Sites would be able to match versions of an RDF by the Author's signature.
  - Additional info, such as Approvals, could be merged together and offered to relevant webs.
  - A request on one web for an RDF might get any of several versions, which does not matter as long as there is a signature by an Authority for that web.
    - This implies that a site may get an RDF from SomeLinuxWeb: which includes a TikiWikiHelp: web approval, then distribute that RDF to a TikiWikiHelp web which it also is participating in. Thus bandwidth from many webs is used, but only based upon each site's configuration.
    - Files may also be shared between various P2P webs. The RDFs for them allow positive identification of them.
  - Should assumption be made that each P2P Web has only one Authority? Simpler that way.
  - Branches of multiple versions of files could be created.
    - How to request "any version of type X"? A 1.8.2 site might accept Help files for any 1.8.* version greater than 1.8.2 (not 1.8.1), yet accept 1.8.0 as a default (if 0 is default for 1.8.*).
    - Would Version:Itemname be a sufficient identifier?

Discussion/participation

Please jump in here and lets think this idea through together this is what))wiki((is all about! This is far too important a feature to just guess our way thru and not understand the *real* requirements and solutions.

-

For a feature like this the devil is in the details...

-

1. Establish a table in the database called tiki_help_pages
   - Like the table that holds wiki pages now
   - A new field will most likely be required in the table that would hold the version number of the "Offically" released page as identified by us. I'm not sure that the date time of the page would be the best answer here.
     - 1.7.0.0.000 or something simular to identify a page released for the 1.7 release
2. help/doc pages are stored in this table and referenced from this table
   - If/when one of these pages is edited then normal page versioning is handled the same way as it is for any wiki page except that the version number field would have to be managed.
     - Options for saving to this new table will require modification to add selection of storing in the help pages table
3. pages can be sent or received via the comm feature the same as for wiki pages
     - options for sending and receiveing will require modification to add selection of storing in the help pages table
4. If new pages are imported (due to feature changes or updates) then they should be imported with the next page version number of the same page name.
   - If the page being replaced has been modified locally then the modifications will not be lost but will be stored in the history
5. Special functions such as selecting the admin feature for the admin pages and selecting an option like update would pull a table update file from an official location and automagically update the table for any page that has been updated.
6. The tiki_help_pages should most likely have some minimum requirements for the number of versions that it will hold.
   - Is this number of versions the seperate for locally modified copies?
   - If there are at leaset 3 versions required prior to delete prior versions for these "Official" Help pages is that too much or not enough should this number be locally adjustable?
   - Locally modified copies would receive a local version number that would not interfere with the "Official" version number
     - While this number should hold some relationship with the "Official" Help/doc page this local would have its own "local" version number and page version number
     - There should be a locally selectable number of "local" versions of the page to be kept prior to deletion of the oldest versions.
     - Is this number the same as for standard wiki pages?

Questions

1. Should all Help/Doc pages be stored in this same table?
2. Should there be a seperate table simular to this for storing the Doc pages?
3. If all Help/Doc pages are stored in this database then:
   1. What new permissions would be required?
   2. Should there be an option for sending updated documentation back to us?
   3. Would the only the single latest "Official" version stored be the version of the Tiki that is installed? Plus any locally modified pages?
4. The notion of a collaborative help authoring system (not necessarily tiki specific) is very valuable IMO. In our case, JGraph.net would like to use tiki to document our application. This means we would like to be able to allow people to incrementally add documentation, allow people to filter out 'in-progress' pages and only show pages that have 'official' content, and to be able to package up the content in different forms (as proposed in OldTikiWikiDocumentationPlan with PhpDocumentor). Is there any benefit to thinking of this as a new module with specific help-like functionality rather than as enhanced wiki pages?
5. Doesn't this point to a CVS-like form for Wiki pages to manage different doc versions? *Chealer9*