

TopicMapDev

Status/RoadMap

- Where are we?
 - We have the [Categories](#) feature spanning across multiple tikiwiki features, which could be used for "Tagging" or "Topic-Mapping" as is. Even permissions on categories have been implemented in version 1.9 and above... [ang](#)
- Where do we want to be?
- Who is working on what? (Priorities/goals/majors issues/roles)

TikiTeam

Who is working here generally? [Link](#) [UserPage](#).

Trackers

- Bugs
- RFEs
- tech support
- patches

Standards and Competition

List of other products with similar/interesting/related features.

XML1.0

[eXtensible Markup Language 1.0 \(2nd edition\)](#)

OWL

[Web Ontology Language Overview](#)

RDF

[Resource Description Framework](#)

XTM

[XML Topic Maps](#)

XFML

[eXchangable Faceted Markup Language](#)

OPML

[Outline Processor Markup Language](#)

ETN

[Easy News Topics 1.0](#)

Here I would like to see some "editorial" content. How do our features compare to others?

Useful Links

- <http://www.topicmaps.org> — Home of XML topic map specification
- [The TAO of Topic Maps](#) — Nice introduction to topic maps.
- <http://easytopicmaps.com> — TM community site.
- [XFML.org](#) — Specification home page

- [Topic Maps For Repositories](#)
- [Building a Semantic Website](#)
- [Why Topic Maps](#)
- <http://www.ontopia.net/topicmaps/faq.html> — Explains how "facets" are built-in, and discusses topic maps for CMSs.

Tutorials

- [My first Tutorial](#) — easytopicmaps.com
- [Introduction to XFML](#)
- [Topic Maps Presentation](#)
- [Lessons on Applying Topic Maps](#)
- [A Semantic Integration Methodology](#)

Books

- [XML Topic Maps](#) — by Jack Park (editor) and Sam Hunting (technical editor)

Tools

- [GooseWorks.org](#) — tmtk, mod_topicmap, V.browser opensource projects (in development)
- [STMQL](#) — Query language used by gooseworks tools
- [TMHarvest](#) — Generate topic maps from existing databases
- [Dagon Lair Knowledge Base](#) — topic map driven
- [Accounting & Auditing Topic Map](#)
- [xSiteable](#) — TM static website generator.
- [TM4Jscript](#) — Client-side TM engine
- [The Protege Ontology & Topic Map Editor](#)
- [Omnigator Topic Map Browser](#)
- [Topic Map Designer](#) — win32 only

Example Implementations

- [Organizing CPAN with Topic Maps](#)
- [Bridge Web](#) — In development, more of a topic browser from a developer's view than a production interface.

Data Modeling

- [Mapping Topic Maps to RDBMS](#) — Recommended!
- [Topic Maps to Extend RDBMSs](#) — Recommended
- [Map DTDs to Databases](#)
- [DTD2XML](#)
- [XDB: XML Database](#) — PHP
- [Berkeley DB XML](#)
- [PEAR::DB_NestedSet](#)
- [Nested Set Explanation](#)
- <http://phptmapi.sourceforge.net/> — PHP port of a standardized TM API, may be a useful reference.

Heavy lifting theory stuff

- [RM4TM: Reference Model for Topic Maps](#)
- [Topic Map Programming Model](#)

REST Architecture Style

- [REST Wiki](#)
- [Roy Fielding's Dissertation](#) — The REST Bible
- [More rest resources](#)
- [RESTful Web Services](#)
- [sqlREST](#) — Exposes relational databases as REST style Web Service
- [Giving SOAP a REST](#)

CVS Doc section

This is where new features being developed and only in CVS are documented. When the CVS becomes RC/official release, the info in the CVS docs is transferred to update the official docs (FeatureXDoc).

Discussion/participation

Where ideas can be exchanged, debated, etc. Interested people can subscribe to the wiki page and/or to these forums as they would a mailing list.

An idea

Imagine a website that is navigable like DMOZ directory; but, instead of displaying categories as a directory, display them as the navigation of each page (e.g., child links on left, siblings on right, "see also" on right, parents on top crumb-trail style); the main section holds content links or the content object (leaf node). The result is context sensitive navigation. You can try this at home: pick any links management application and modify the display as described above ... Voila!

Add some bells and whistles, convert leaf nodes from content links to content display, enrichen the classification datamodel to handle facets and other topic map features, create a plugin system to connect other services (and their data) into the system, create a base template that lays out the navigational elements, and voila!, a possible Tiki 3.0? --pacoit

A stab at some definitions

Classification: the process of categorizing the subjects of a subject domain. Types of classification structures:

- hierarchical
- alphabetical
- chronological
- relational - this is more involved because it requires having an ontology that defines relation types and the topic types to relate. This is how knowledge can be powerfully represented, and where topic maps come in.

Taxonomy: A classification of a subject domain based on established relations among the subjects in the domain. Usually there is a most significant trait chosen as the basis upon which a hierarchical classification is derived.

Facets: Taxonomies of a subject domain derived from the classification of a secondary trait of the domain. Facets provide additional views or perspectives of the same domain (e.g., search for cars by price, class, or manufacturer--3 different facets).

Ontology: The definition types of topics and types of relations between topics, both of which are understood to be concept definitions. It is the vocabulary with which queries and assertions can be made. It also refers to the task of selecting concept labels (names) that are intuitive and most easily understood by the target audience. Concept definitions must be unambiguously defined so that identical concepts with differing labels (due to different language, synonyms, spelling, etc.) can be reconciled as identical either by humans or machine. This makes it possible to merge two different subject domains, enriching both in the process.

- Consider: There is Mozart the person (topic) and Mozart the subject (topic type). Mozart can't exist in the computer, but he can usefully be represented by a web page that provides a biography of Mozart the person (topic). However, it would also be useful to represent the subject of Mozart (topic type), including links to: his biography, list of musical works, books written about him, pictures, etc.. Mozart (topic) falls under the subject category of Mozart (topic type).

Thoughts on XFML vs XTM topic maps

XFML was developed as a simplified path to topic map capability. XFML is a means to define faceted taxonomies. This is a welcome addition for searching a directory; but it does not provide for the navigation of a whole website, i.e., the smooth context-sensitive navigation across multiple subject domains of a website (articles, documents, events, forum topics, comments, members, projects, tasks, etc.), with context-sensitive links to expand the current topic.

Faceted classification is ideal for repositories of specific topic types (e.g., articles, books, images, mp3s, events, cars, houses, etc.). However, topic maps provide the association types needed to bridge across different topic type repositories.

An application like [Flamenco faceted browser](#) is narrowly focused on one topic type (e.g., images). You can sort (facet) your navigation according to periods, location, style, etc. of art images; but there is no expansion of each topic: an image page does not provide links to information about the author, or the art style, or related writings, etc.. This might be desired depending on the application. But facets can't provide more than this. Flamenco improves greatly on information access; but it is still a walled-in application.

Whereas taxonomies (faceted or not) address a specific domain, topic maps could be described as domainless, in the sense that they provide smooth navigational transition from one domain to another. A good thing about XFML is that it can be translated into a simple topic map, which can subsequently be enriched as the Tiki CMS provides expanded forms for adding the additional information. ... to be continued