

Open Source Wikis

New platform for technical publishing

peter.dykstra@metaphorx.com

www.metaphorx.com

Overview

- An intro to Wikis
- Wikis vs CMSs
- Relevance to technical publishing
- Usage scenarios / demo
- Implementation issues

About MetaphorX

- Information design consultancy
- Solutions for technical publishing
 - Assessment
 - Strategy
 - Design & Implementation
 - Training
- Specialize in open source
- Available to help

What's a Wiki?

- First introduced in 1995
- Server-based Web software for discussion / collaboration
- Distinguish
 - Wiki
 - Blog
 - Forum

Original definition

“The simplest online database that could possibly work.”

- Ward Cunningham – 1995

www.wiki.org

Wiki characteristics

- “If you see it you can edit it.”
 - Allow users to freely create and edit Web content
 - Use any Web browser
 - Simple text syntax (HTML not required)
 - Cross links

(source: www.wiki.org)

Effect of Wikis

- ❑ Powerful and subtle effects of Open Editing
- ❑ Everyday users can edit any page
- ❑ Encourages democratic use
- ❑ Promotes content composition by nontechnical users

(source: www.wiki.org)

Why of interest to tech pubs?

- Corporate use is increasing
- Technology is good and getting better
- CMS-based models for editorial work
- A way to move toward Web 2.0
- New roles: who manages Wikis?

Issues

- Learning curve
- Geek factor
- Establishing ROI – specific applications, goals and rationale
- Implementation strategy

A look at Wiki history

Original Wikis

- ❑ Simple storage, files stored in server directories
- ❑ One copy, no versioning
- ❑ Issues
 - Sprawl
 - Hard to navigate random structures
 - No edit history / tracking

Newer Wikis

- ❑ Thousands of pages

Wikipedia English version = 1,688,000+ articles (3/16)
2,039,000+ (10/9)

- ❑ Formal governance models

- ❑ Database back ends

- ❑ Versioning, tracking, and rollback

- ❑ Role-based access control

- ❑ Specialized administrator roles

Increased corporate use

- Discussion forums, blogs
- Workgroup collaboration
- Complement to formal Intranets
- User groups
- Communities of practice

Goal: collaboration

- “Content management systems usually won’t work as collaboration tools. Although a CMS does ease content creation, the resulting Web sites have rigid navigation and formal publishing workflows. Wikis, on the other hand, *empower users to create new pages ad hoc as well as create links easily within the content.*”

- InfoWorld, 2007 (emphasis added)

2005: Information Week's Prediction

- Content management is likely to hybridize with the Wiki into a new, more robust application that combines the strengths of both tools.

- Information Week, August 2005

2007: CMS-based Wikis are common

- ❑ High-end Wiki platforms are essentially browser-based Content Management Systems
- ❑ Mature products, active development
- ❑ Main CMS/Wiki difference at this point is the governance model

Wiki vs CMS Governance

Wiki

- Bottom-up approach to structure and navigation
- About empowering users

CMS

- Top down approval / coordination
- About managing content efficiently following a set of rules

Similarities

- ❑ Can use the same technology
- ❑ Wikis require SOME central management
- ❑ Similar user needs
 - Edit rights
 - Ease of use

Differences

- ❑ Social and governance model
- ❑ DEGREE of control --
continuum
- ❑ Need to identify the source /
speak with one voice?
- ❑ Specialized roles?

Why Open Source?

- ❑ “Golden Age of Open Source software” – Information Week
- ❑ OS and Wikis have grown up together
- ❑ Often created to support other OS projects
- ❑ 2000+ Wiki projects

Common attributes

- Open to any participant
- Standards-based
- Virtual communities of use
- Technical orientation

Two main technology branches

□ PHP

- Many of the most widely used systems
- MediaWiki, Drupal, Plone, SocialText
- Resource: <http://www.OpenSourceCMS.com>

□ Java

- Focus on enterprise use
- Lenya, Magnolia, Daisy; Apache Software Foundation
- <http://java-source.net/open-source/content-management-systems> [sic]

Commercial vendor trends

- SaaS
- Emphasis on solutions
- Features (e.g. DITA support)

Response to OS?

Some main trade-offs

□ Vendors

- Rich feature sets
- Support
- Knowledge of business context
- Attention to usability, user-centered design
- Buy a full solution

□ Open Source

- Access to apps and code
- Widely used, robust components
- Communities of users and experts
- Build a solution
- Buy help and support as needed

Option 3: open source vendors

- Developers or third parties
 - Strategy
 - Customization
 - Project management
 - Training and support
 - Writer/editor/administrators
 - Developers

Relevance to Tech Pubs

Three opportunities

1. Improve traditional publishing process
2. Enable Web 2.0 capabilities
3. Coordinate the two into a hybrid publishing model

These are all *essentially* editorial tasks.

1. Improve traditional process

- Meet growing demands
 - Formats, versions, speed, online delivery
- Eliminate current production bottlenecks

Three previous ages of technical publishing

- DTP Age – FrameMaker, Word, Quark
- Windows Help Age – Help Authoring Tools (HATs): Robohelp, DoctoHelp, etc.
- Web 1.0 Age – PDF, Dreamweaver, Webmasters

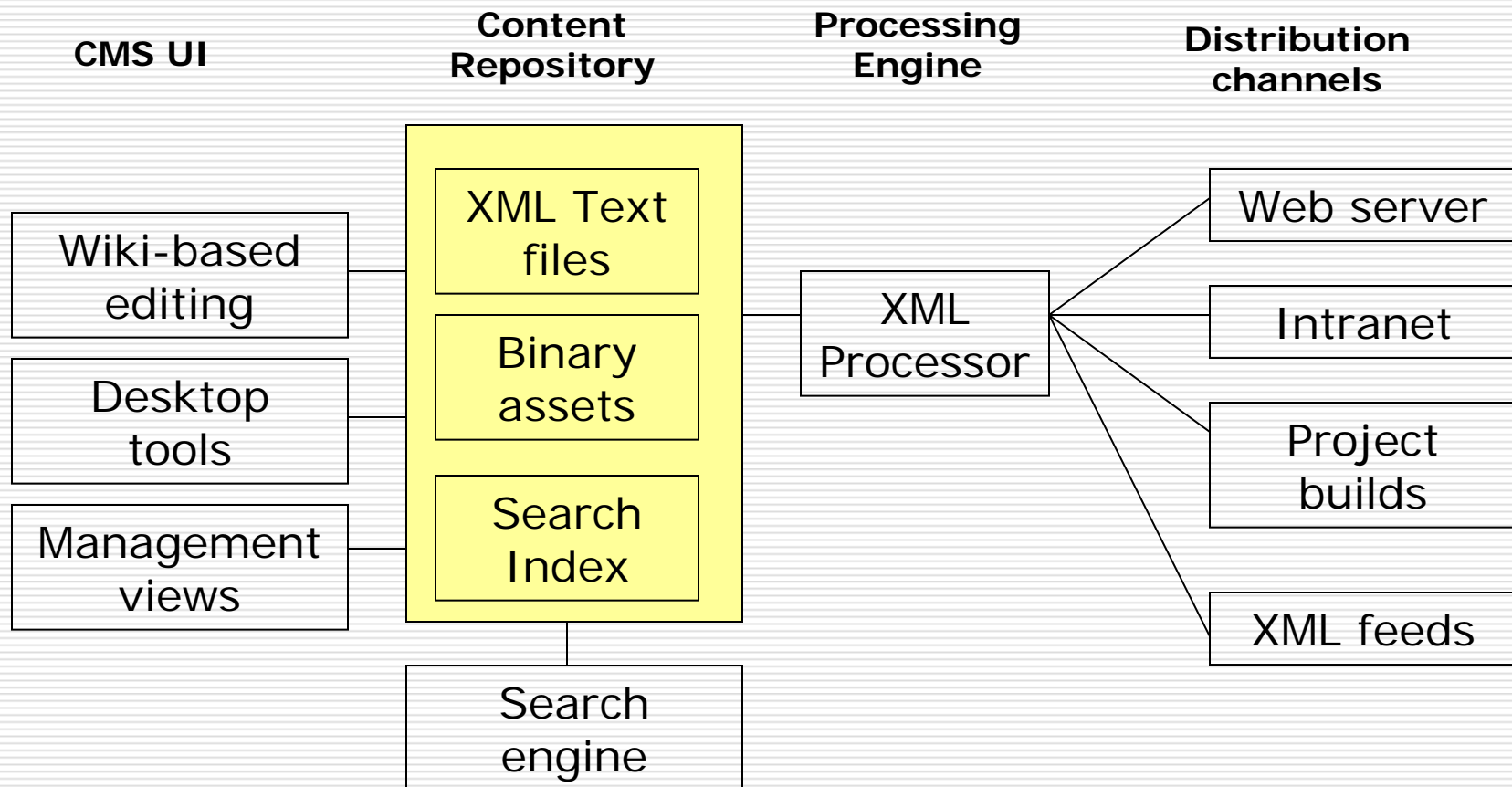
Typical TP environment

Edit tools	Storage	Processing / distribution
FrameMaker for PDFs Word for Help Dreamweaver for Web pages	Source files in directories (on LAN or in source repository). Manual file management.	Repetitive manual production processes. Writers do PDF and Help production. Hand off or post files for Web and build groups

Characteristics

- ❑ Multiple single-user tools
- ❑ Redundant processes
- ❑ Often project- or team-specific
- ❑ High % of time on manual production tasks and file management
 - Files, versions, dependencies
- ❑ Error-prone

High-level CMS architecture



Characteristics

- Integrated content model
- Defined workflow
- Repeatable automated processes
- Role-based task support
- Centralized management views

Results

- Improved quality
- Internal efficiency
- Ability to support richer user experience
 - Role-based customization
 - Richer UI
 - Multiple access channels

Additional benefits

- Content creation by SMEs
- Internal reviews
- Integrated Web delivery

Support for best editorial practices

- Structured writing +
- The Web +
- Standards-based Wiki/CMS

2. Enable Web 2.0 capabilities

- Monitor content use directly
- Feedback / comments
- Enable users to participate
 - Forums
 - Client Wikis
 - Blogs

3. Hybrid model

- Synergy between
 - Authoritative published content
 - Open or controlled contributions, blogs, etc.
- Moderated discussions
- Support relevant user communities
 - Product users
 - Support engineers, etc.

Natural fit / opportunity

- ❑ Organizations need dynamic, credible, online information sites
- ❑ Core competency is editorial management
- ❑ Tech pubs groups need direct access to information consumers

Roles for info developers

- Moderator
- Producer
- Online editor
- Synthesizer
- Architect
- Traffic cop

5 usage scenarios

- Internal doc sharing ('vanilla' Wiki)
- Traditional (pdf) doc delivery
- Web content management
- Online Context Help (DITA?)
- Hybrid publishing + Web 2.0

Implementation Issues

1. Getting oriented

- Browse online
- Download a package and play
- Do a small pilot project
- Read (e.g., Ann Rockley, *Managing Enterprise Content*)

2. Business focus

- ❑ Clear goals and ROI
- ❑ Think broadly, start small (crawl, walk, run)
- ❑ Don't plan on spontaneous behavior changes
- ❑ What constitutes success?
- ❑ Go for proof of concept, quick wins

3. Editorial focus

- Active CMSs or Wikis require editorial management
- Which editorial skills are required at what points? Consider:
 - Audiences
 - User expectations
 - Internal vs. external readers
 - Relevant quality measures

Content requirements

- Meet need + be credible
 - Known, authoritative source
 - Relevant, helpful to users
 - Fresh
 - Accurate
 - In synch

4. Technology

- Product selection
 - Every product has different strengths
 - Get advice, try a pilot
- Production environment
 - Hosting service vs. in-house
 - Initial planning and setup
 - Ongoing support - problems, upgrades
 - Directory management/authentication, notifications, backups

DITA Support?

- Some OS editors support DITA
- Possible levels of support
 - Export/extract to DITA
 - Basic DITA format
 - Full DITA support

5. Usability

- ❑ Usability for casual users in your environment is a key value add
- ❑ Plan to customize the UI
 - Consider work context, 80/20 rule
 - Configuration + some customization
- ❑ User-centered design
- ❑ Usability testing

6. Assess roles/skills

- CMS project management
- Information architecture
- Web/UI design
- Technical design/development (XSLT, CSS, XML, Java/PHP/Ruby, etc)
- Editorial/process management

Closing

- ❑ CMS: complex but critical area
- ❑ Open source Wiki/CMS products are maturing and improving
- ❑ Way to leverage resources
- ❑ Participating in OS communities can be personally and professionally rewarding
- ❑ Dive in!

Contact

- Peter Dykstra
peter.dykstra@metaphorx.com
www.metaphorx.com