Overview

- Basic Ideas
- Advanced Concepts
- Includelets
- Weblog
- Prototypes
- Policies
Basic idea

- Wiki + Content Management + Dynamic Page Fragments = Basic Web Application

- Ease of Wiki
  - Named Pages/Links for Referencing and Content creation
  - Reference Management
  - Multi-Language support
  - Revision Management

- Extended Features
  - Typed Wiki Pages
    - Predefined: xowiki::Page, xowiki::PlainPage, xowiki::Object, xowiki::PageTemplate, xowiki::PagelInstance, xowiki::File
    - XoWiki pages are subclassable
  - Typed Links
  - Search with weighted Terms
  - Categories and Delicio.us like tagging (Folksonomies)
  - ADP-Inclusion for reuse of OpenACS packages, nested content
OpenACS Home Page (source)

>>left-col25<<

{{adp portlets/wiki {name de:News}}}
{{adp portlets/wiki {name de:Recent\ Posts}}}
>>

>>right-col25<<

{{adp portlets/wiki {name de:About\ OpenACS}}}  
{{adp portlets/wiki {name de:Recent\ Posts}}}
>>

OpenACS (Open Architecture Community System) is a toolkit for building scalable, community-oriented web applications. OpenACS is the foundation for many products and websites, including the .LRN e-learning platform. OpenACS is open source and is available under the GNU General Public License. ...

This is the Personal Wiki-Page of Gustaf Neumann. This shows, how to place arbitrary text into a kind of Portal Page developed as a wiki.

-gustaf neumann

{{adp portlet {title Groups portlet/packages/dotirm/www/dotirm-main-portlet}}}
{{adp portlet {title #forums-portlet.pretty_name# portlet forums-portlet}}}

>>

{{adp portlet {title #dotirm-calendar.Day|Summary# portlet calendar-portlet}}}

Path: body

OpenACS Edit

MySpace OpenACS Home Page (source)
Typed Links

- **Idea:** Define different kind of relationships between Pages
- **Realization:** A Link is a Class defining
  - Page lookup
  - Renderer for links for found pages
  - Renderer for links for unknown pages
- **Example**
  - The [[glossary:PostgreSQL]] Global Development Group ...
  - Page named PostgreSQL in an XoWiki instance named glossary
  - Use special symbol for denoting the link type on Link Label
  - Use a popup for rendering the glossary content (using overlib.js)
Aggregate Viewers

- View contents of a folder of XoWiki Pages
  - View all pages (maybe of a certain type)
  - Recently changed pages (maybe per category)
  - Recently visited pages, most visited pages, ...

- Weblog-Portlet
  - List multiple entries (XoWiki pages) in chronological order
  - Include Title/author/Modification Date
  - Full content or summary of pages
  - Filter by categories (single multiple), tags, date, tags, ....
  - Exclude items
  - Pagination
  - Renderer for Items + Renderer for Container, can be changed via mixins
Including content in XoWiki Pages

- **Adp-Includes**
  - Standard OpenACS include semantics; example
  - `{{adp portlet {portlet } /packages/dotlrn/www/dotlrn-main-portlet}}`

- **Include of other XoWiki content**
  - Full pages
  - Page fragments
  - `{{categories-portlet -count 1 -skin plain}}`
  - Similar to function calls returning HTML

- **Argument Resolving**
  - Portlets provide defaults
  - Content can be taken from URL
  - Invocation in page provides values
Prototypes

- Named after “Prototypes” in object based programming languages like SELF
  - Rather than creating fresh instances of some type, developers provide clone-able content
  - Example in architects software:
    - some designer creates a sofa via graphic tools, user can copy the sofa and put it into his room, maybe with slight modifications
  - Similar with web pages

- Need for shared pages between multiple xowiki instances
  - Index page, Weblog, ...

Prototypes (2)

- Prototype Life-Cycle in xoWiki
  - File named SOMENAME.page in xowiki/www/prototypes
  - Page name resolver checks prototypes
  - If found instantiates for the package instance denoted by url
  - Revisions are kept in the content repository
  - If page is deleted, it is recreated in a fresh query for SOMENAME

::xowiki::Page new -title "Index Page" -text {
  <p>This is the default start page of XoWiki. You can edit this page and save it to provide a personalized look of the XoWiki instance.....</p>
}
Policies

- XOTcl Objects to govern access control decisions
- Configurable via package parameter
- Used to control the visibility of links for certain users
- Define, what methods on objects of which classes can be used by which subjects
- Possible to differentiate on attributes of the objects
- Subjects:
  - None, swa, login
  - [read|write|create|admin] permission on the values of certain attributes

Policies (2)

Policy policy3 -contains {
  Class Package -array set require_permission {
    reindex {{id admin}}
    rss none
    delete swa
    edit-new {{has_class ::xowiki::Object id admin} {id create}}
  }
}

Class Page -array set require_permission {
  view {{item_id read}}
  revisions {{item_id write}}
  edit {{item_id write}}
  make-live-revision {{item_id write}}
  delete-revision swa
  delete swa
  save-tags login
  popular-tags login
}

Class Object -array set require_permission {
  edit {{package_id admin}}
}

Class File -array set require_permission {
  download {{package_id read}}
}
}
XOTcl primer

- Tomorrow: Friday morning
- Short XOTcl introduction
- XOTcl and aolserver (initialization, serialization)
- Xotcl-core
  - Oo-layer for content repository
  - Forms, templating, mixins
  - Threading
  - Async I/O; background delivery