The XIMS Project, 2004-05-21

Michael Kröll
Slides are available at http://xims.info/
The XIMS Project Overview
What is XIMS?

● XIMS Mission Statement:

"The XIMS Project aims to develop a web-based extensible Information Management System, built to especially suit the needs of academic and educational infrastructures."

● XIMS has been developed at the University of Innsbruck and released as Open Source (Perl Artistic License)

● Who is using XIMS?
  ◆ Universities of Innsbruck (~70 homepages, ~950 users)
  ◆ Management Center Innsbruck
  ◆ newSof Group
  ◆ Medienzentrum, University of Siegen
  ◆ ...

Major Features

- Customizable ACL-/Role system
- Extensibility with individual application modules
- Easy WYSIWYG-Editor integration
- Separation of application and presentation logic (XML/XSLT)
- Designed to flexibly integrate into existing infrastructures
- ...
Technical Implementation Overview
What do you need to get it working?

- Webserver with Apache AxKit support
- Oracle or PostgreSQL RDBMS
Scratching the surface...
Core Module Dependencies

- CGI::XMLApplication
- XML::LibXML, XML::LibXSLT
- DBIx::SQLEngine
- XML::SAX::Generator::PerlData
- AxKit
Application Classes Showcase
Putting OO in your CGI Application Logic
Class Diagramm Overview

Diagram showing class relationships and methods:

- **CGI::XMLApplication**
  - **XIMS::CGI**
    - +registerEvents()
    - +getDOM()
    - +selectStyleSheet()
    - +event_access_denied()
    - +event_default()
    - +event_edit()
    - +event_create()
    - +event_store()
    - +event_cancel()
    - +event_delete()
    - +event_publish()
    - +event_unpublish()
    - +event_obj_alllist()
    - +event_obj_activate()
    - +event_obj_activate()
    - +event_search()
    - +event_sitemap()
    - ...

- **XIMS::Exporter**
  - **XML::LibXML::SAX::Builder**

- **XIMS::CGI::File**
  - +registerEvents()
  - +event_default()
  - +event_edit()
  - +event_store()

- **XIMS::CGI::Document**
  - +registerEvents()
  - +event_default()
  - +event_edit()
  - +event_store()
  - +event_preview()
  - _resolve_annotations()
  - _set_wysiwyg_editor()
# This is XIMS::CGI::File::event_default()

```perl
sub event_default {
    XIMS::Debug( 5, "called" );
    my ( $self, $ctxt ) = @_;

    return 0 if $self->SUPER::event_default( $ctxt );

    print $self->header( -type => $ctxt->object->data_format->mime_type() );
    print $ctxt->object->body();
    $self->skipSerialization(1);

    return 0;
}
```
# This is XIMS::CGI::Document::edit()
sub event_edit { 
    XIMS::Debug( 5, "called" );
    my ( $self, $ctxt ) = @_;

    $ctxt->properties->content->escapebody( 1 );

    # event edit in SUPER does things like checking for
    # edit privileges and setting a lock
    return 0 if $self->SUPER::event_edit( $ctxt );

    # check if a WYSIWYG Editor is to be used based on cookie or config
    my $ed = $self->{_set_wysiwyg_editor}( $ctxt );

    # $ed will be '' if no WYSIWYG Editor is to be used
    $ctxt->properties->application->style( "edit" . $ed );

    return 0;
}
Further Resources

http://xims.info/