OPACIAL Technical Documentation

Version 1.0

Dimitris Gavrilis (gavrilis@gmail.com)

OPACIAL – Web 2.0 Online Public Access Catalog
http://opacial.sourceforge.net

Digital Curation Unit, Athena Research Centre
http://www.dcu.gr

Library & Information Service, Panteion University
http://library.panteion.gr

Contents

Introduction

The OPACIAL system is an interface to any Online Public Catalog system (OPAC) that has a Z39.50 interface. A working demo of OPACIAL can be seen at: http://library.panteion.gr/opacial. OPACIAL is open source (distributed under the Academic Free Licence) and can be downloaded from http://opacial.sourceforge.net.

It consists of two main subsystems:

- The user subsystem

- The management subsystem

The first is the interface that the users use in order to perform a search (Figure 1). The latter is the management interface the librarian uses to configure OPACIAL and maintain it.
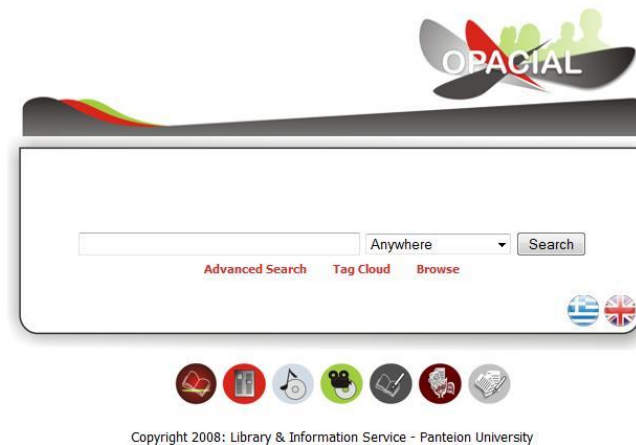
Figure 1: OPACIAL main page featuring a minimalistic interface.

Requirements

OPACIAL is web based system that requires a web server, (preferably Apache, version 1.3.x or later), PHP (version 4.x or later) and a MySQL (version 4.x or later) database. It is recommended that a PHP 5.x and a MySQL 5.x version is used. The php installation must be compiled to support the YAZ and GD extensions. This can accomplished either by compiling the extensions into the php installation or by using the PEAR extension to install them into an existing php installation.

- Apache: http://httpd.apache.org/
- PHP: http://www.php.net/
- MySQL: http://www.mysql.com/
- PEAR: http://pear.php.net/

Installation

In order to install OPACIAL in a Unix/Linux environment, follow the steps below:

1. Unzip the package and cd into directory

   #> unzip opacial_v1-0.zip

   #> cd opacial_v1-0.zip

2. Create mysql database

   #> mysqladmin create opacial

3. Create mysql tables

   #> mysql opacial < opacial.sql

4. Edit opacial configuration: config.php

5. Edit administration configuration: admin/configuration.php

6. Edit the apache httpd.conf

[Example of httpd.conf entry]

```
Alias /opacial/ "/srv/www/opacial/"
 Alias /opacial "/srv/www/opacial/"
 <Directory "/srv/www/opacial/">
    Options Indexes MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
 </Directory>
```

The user subsystem

The user subsystems comprises of the interface the visitors use to perform a search and locate items. The users are displayed with the simple search interface which can be used to perform a simple search on the catalogue. Alternatively, the users can use the advanced search option or they can navigate using a social tag cloud. The system's preferred language is set in the main search page (simple or advanced search). When a user performs a search, he/she is presented with the search results on the main pane (center). The interface layout is simplistic, and only the title, author, call number and status are shown. Also, a book thumbnail is retrieved from amazon.com (if it can be retrieved using the ISBN/ISSN). Alternatively, one can upload book/journal thumbnails onto the cache folder using the item's record number and a .jpg extension (e.g. 10346345.jpg).

On the right of the search results page, there are some helper blocks that can be used to provide additional functions. The implemented blocks are:

- Tag cloud: the tag cloud block displays the sum of the distinct tags found in the current search results (in order to improve performance the block searches for tags for the next 100 items starting from the current one). (Figure 2)

- Material type: the distinct set of materials is presented for all the items returned in the results. The user can use this option to filter out certain material types (e.g. only journals). Cancelation of the filter is performed using the [X] in the upper right corner of the block. (Figure 3)

- Year: The distinct sets of years found in the returned search results are presented and the user can browse the results using a specific year. Cancelation of the filter is performed using the [X] in the upper right corner of the block. (Figure 3)

- Subjects: the distinct subjects found in the returned search results are presented and the user can browse for items with specific subjects. Cancelation of the filter is performed using the [X] in the upper right corner of the block. In order to improve performance and aesthetics, only about 40 subjects are presented. (Figure 3)

- Related items: this block presents similar items from the digital library (DSpace repository). The matching is performed using the subjects found in the search results and the dc:subject field of the DSpace repository.

Figure 2: The search results page (see tag cloud on the right)

When clicking on a record, the full details of the record are presented. OPACIAL uses information abstraction in order to improve the presentation of the information and split this information logically. In the holdings view, the visitor is presented with a map of the current floor on which the book resided. The shelf that contains the book is marked red.

Figure 3: The search results page (see faceted browsing on the right)

The user can also view the item record in different formats like MARC, XML, BibTex. These options are available on the bottom right corner of the page. Furthermore, the user can annotate and share the record link using the various social tagging networks available by using Socializer (http://ekstreme.com/socializer/). This option is available on the bottom left corner of the page.

OPACIAL can interact with the users through its Web 2.0 features. More specifically, users can annotate each record with a set of tags. Additionally, users can write their opinion about a record and rank it using a 1-5 Likert scale.
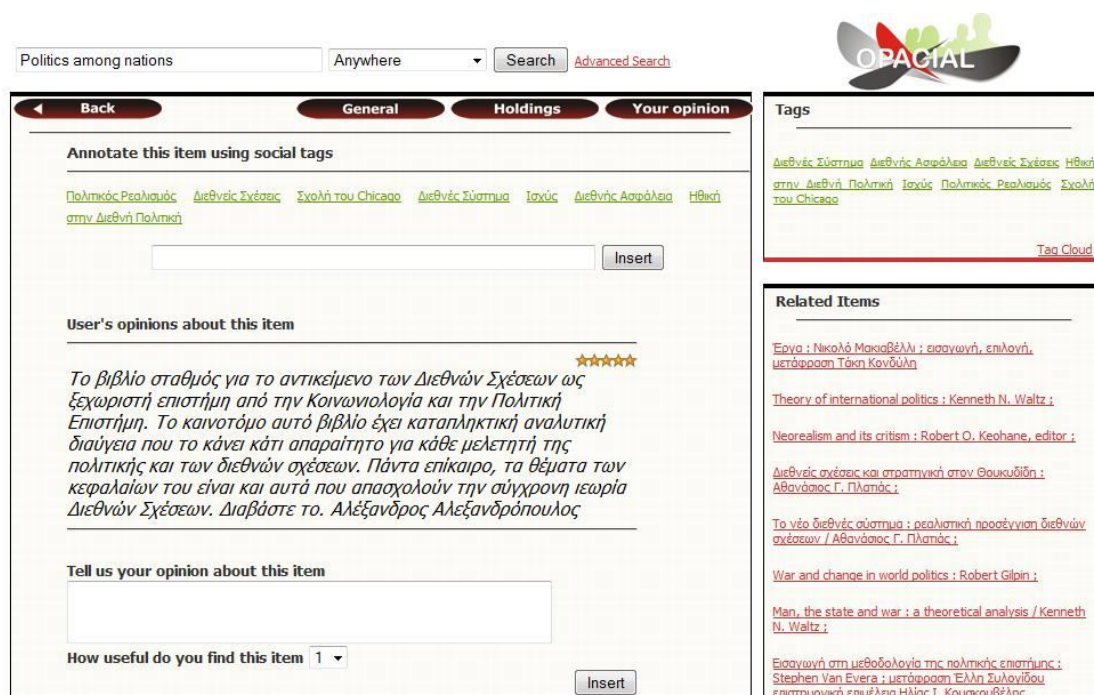

Figure 4: Users can insert their own tags, write their opinion and rate an item.

The management subsystem

The management subsystem's primary role is to configure the OPACIAL catalogues, views, MARC fields and subfields. OPACIAL can attach to any Z39.50 server. The Z39.50 server that

OPACIAL will use is located in the <u>opac_catalogs</u> table. For each catalogue, there exists a number of views which are used mainly for the information abstraction process. Example views are:

- Full view: this view is the main view of a result. It usually contains most of the metadata.

- Holdings view: this view contains information regarding the holdings of an item.

Each view displays a set of marc fields (Figure 5). These marc field sets are defined in the <u>opac_marc</u> table and can be administered through the administration management interface (<u>http://opacial.library.gr/admin</u>). Each row represents a field. The field code (e.g. 200) is set along with its value (e.g. Title). The field must be set Active in order to be shown. Additionally, the catalog must be set (e.g. Main Catalog), the View (e.g. Main view), the language (en=English, el=Greek) and the position (e.g. 10). The fields will be displayed ordered by position. For each field, the user can set up to 8 subfields that can be shown <u>simultaneously</u> (it is possible to add more subfields). For each subfield there is a prefix and a suffix that contains text/html that will precede/follow the text associated with the subfield. The Type field next to each subfield denotes the subfield as a normal text (-) or a role (R). When normal text is selected, OPACIAL displays the subfield value as it is. When Role is selected, OPACIAL replaces the subfield value with its corresponding translation found in the opac_roles table.

Figure 5: MARC Field administration.

The Roles can be administered through the administration interface (Figure 6).

Figure 6: MARC Roles administration.

OPACIAL can also display where a book resided on the library (Figure 7). In order to do that, the floor plans must be uploaded onto the server on the floor_plans folder. Each shelf must be marked onto the floor plan and the corresponding record using the inserted into the opac_maps table. There is also a web interface through the administration interface that can manage this. Each shelf is a rectangle on the image file and its upper left corner coordinates (x1,y1) along with its width,height (x2,y2) are used to mark the rectangle.

Figure 7: Map showing (red) the location of a book.

The mapping of the books onto the floor_plans is performed using the book dewey number (Figure 8). This association is inserted into the opac_maps_assoc table. This table can hold dewey number ranges (e.g. 105.1-107.6) that are mapped to a specific rectangle on the floor plan. So one could mark a rectangle that contains many shelves and map a wide range of dewey number onto this rectangle. This allows for maximum flexibility since the librarian can map the library books using adaptive accuracy.

Figure 8: Administration of call number mappings to map locations.

The database

opac_catalogs

This table holds the available catalogues that OPACIAL searches. Theoretically, it OPACIAL could search in parallel in different catalogues. An example configuration can be seen below:

Field	Value	Remarks
title	The name of the catalogue	
title_en	The name of the catalogue in english	
url	The url of the Z39.50 catalogue	e.g.: library.panteion.gr:210/ADVANCE
isActive	sets this catalogue active or not	1=active 0=inactive

opac_views

This table holds the available views that OPACIAL uses.

Field	Value	Remarks
view	The name of the view	
view_en	The name of the view in english	

opac_fields

This table holds the available fields that appear in the search box (e.g. Anywhere, Title, Author, ISBN, etc.). It is also used to link opac_marc fields and make them searchable by clicking on them (e.g. when clicking on an author name, to automatically perform a search for all the authors with the specific name).

Field	Value	Remarks
catalog_id	The catalog_id associated with this field	
language	The language for which it will be displayed	e.g.: en=English, el=Greek
field	The displayed title	e.g.: Anywhere
value	The bibset attribute	e.g.: @attr 1=1035
position	The position that fields will be sorted by	
isActive	Enable / Disable this search field	

opac_roles

This table holds the translation for the various roles found in MARC. For example when a marc subfield contains 005 and it corresponds to a role, it will be translated to whatever 005 is set to on opac_roles – in this case to Actor.

Field	Value	Remarks
code	The code of the role	e.g.: 005
role	The displayed title	e.g.: Actor
role_en	The displayed title in English	

opac_maps

This table holds rectangles that represent shelves or sets of shelves on the library floor plan. The user needs to define the appropriate rectangles on the floor plan image.

Field	Value	Remarks
floor_img	The filename of the corresponding floor plan file.	e.g.: floor_1.jpg
rect_x1	The x coordinate of the upper left	

	corner of the rectangle	
rect_y1	The y coordinate of the upper left corner of the rectangle	
rect_x2	The width of the rectangle	
rect_y2	The height of the rectangle	
display_text	The text that is displayed in the holdings view on the results page.	
remarks	Some remarks about the rectangle	e.g.: which shelf represents

opac_maps_assoc

This table holds the association of the dewey number ranges to the opac_maps records defined in the opac_maps table. The user can define a single dewey number (e.g. 105.11) or a range (e.g. 105.1-107-25) and associate it with a map file (e.g. a specific rectangle on the floor plan image). With this way, all books are mapped onto the shelves of the library using adaptive accuracy. The user can define a shelf image file that is displayed next to the floor plan and displays the exact position of the book on the shelf.

Field	Value	Remarks
dewey_start	The start of the dewey number	
dewey_end	The end of the dewey number	
opac_map_id	The opac_maps.id	
shelf_img	A shelf image filename	optional