Capturing and Modeling Open Content in One Laptop Per Child: A Framework and Case Study

Sila Chunwijitra¹, Marut Buranarach², Supranee Thirawatthanasuk¹
¹Opensource Development and Promotion Devision
²Human Language Technology Laboratory
National Electronics and Computer Technology Center, Pathumthani, 12120, Thailand
sila.chunwijitra@nectec.or.th, marut.buranarach@nectec.or.th, supranee@nectec.or.th

Abstract

The One-Laptop-Per-Child (OLPC) project is an initiative to create educational opportunities for children in developing countries by providing each child with a low-cost, low-power, connected laptop with content and software. Availability of open software and content for the OLPC platform is necessary to make the project succeed. This paper describes our development of a pictorial dictionary prototype for children on OLPC machine based on open content. We present a framework for capturing and modeling open content in the OLPC platform. A design of user interface that facilitates users’ browsing and access of pictorial content on OLPC machine is also provided. The initial evaluation results indicated that our prototype and user interface design are relatively effective for children. By promoting reuse of open content in OLPC platform, it can lead to a more diverse access to the content and allows it to have a greater impact on children education.

Keywords: OLPC, Open Content License, Creative Commons, User Interface Design

1. Introduction

People who live in rural or remote areas, especially of developing countries are currently faced by the problem known as digital divide. The digital divide problem can be generally described as the lack of computer equipments, network infrastructure and access to digital content. One of the initiatives in bridging this gap is the One Laptop Per Child (OLPC) project [1]. The OLPC project attempts to create educational opportunities for children in developing countries by providing each child with low-cost, low-power, connected laptop with content and software. One of the design principles of OLPC is its software architecture and content based on free and open source. XO, the current laptop model produced by the OLPC project, is built from free and opensource software, i.e. a Fedora-based Linux operating system. In the context of learning, content that is free for reuse and redistribution is promoted to allow for a more diverse use. Availability of open software and content for the OLPC platform is necessary to make the project succeed.

In this paper, we describe our initiative in developing a Junior Pictorial Dictionary (JPD) prototype for the OLPC platform based on open content and a wiki tool. Pictorial dictionary, such as Oxford-Duden Pictorial Thai & English Dictionary [2], augments traditional dictionary by using pictures to illustrate subjects. Pictorial dictionary helps the children to recognize things better by associating pictures of various subjects with terms and definitions. Today, there are a large number of digital pictures available with the open content license, e.g. those available from Flickr [3], Wikimedia Commons [4], etc. This permits authors to create new content by using these media as authoring components. In developing the pictorial dictionary, pictures were mainly collected from the existing sources with the Creative Commons (CC) license. The MediaWiki tool was then used in authoring the information associated with the pictures. Initially, 330 pictorial entries focusing on the subjects of plant and animal were created for the prototype.

The main focus of this paper is on a framework in capturing and modeling open content for the OLPC platform. We also present a design of user interface for the OLPC user interface environment, i.e. Sugar, that facilitates users’ browsing and access of pictorial content. Implementation and evaluation of the design is demonstrated in the JPD prototype development. By promoting reuse of open content in OLPC platform, it can lead to a more diverse access to the content and allows it to have a greater impact on children education.

2. Background

2.1. Open Content Licenses

Typical copyright gives a holder exclusive rights to make and sell copies of the work, to make
The OLPC was founded by Nicholas Negroponte and is a spinoff of the Massachusetts Institute of Technology Media Lab. The operating system for the OLPC is based on a Linux kernel with a unique interface called "Sugar". Because OLPC machine uses Linux as its core operating system, it includes a large number of open source software developed by developers from around the world. XO Laptop is the current laptop model developed by the OLPC. Figure 1 shows an XO laptop with the Sugar user interface environment displayed on the screen.

The physical design of the XO laptop is important in allowing it to be used around the world in different environments. Most importantly, the laptop must be durable and power efficient. The machine contains no hard disk storages. This allows the machine to be resistant to moisture and shocks. It uses a flash device for mass storage, i.e. 1024MB NAND flash. The 433MHz AMD Geode LX processor was used due to its low power consumption and integrated graphics controller. A number of peripherals are also included, such as a 7.5-inch dual-mode thin film transistor (TFT) display with 1200x900 resolution, an integrated 640x480 resolution video camera, an Analog Devices stereo sound chip, and an 802.11b/g compatible wireless Ethernet support. The wireless device enables mesh networking between laptops for information sharing and cooperation. It includes a moisture-resistant rubber-membrane keyboard and touchpad designed for small hands of children. Further, an enclosed battery pack provides between 16.5 and 22 watt-hours of operation with a minimum of 2,000 charge/discharge cycles.

2.2. XO and Sugar

One Laptop per Child (OLPC) is an organization whose mission is to develop a low-cost laptop with accompanying software to extend computer literacy to children around the world [10]. The OLPC was founded by Nicholas Negroponte and is a spinoff of
principles include simplicity, usability, reliability and recoverability.

3. Development of Junior Pictorial Dictionary

Dictionary is a reference book that provides alphabetical list of words and the information about them. Many forms of dictionary exist including those in book formats and online formats. Pictorial dictionary is a special kind of dictionary that augments traditional dictionary by using pictures to illustrate subjects. The Oxford-Duden Pictorial Thai & English Dictionary [2] is an example of pictorial dictionary in print. For children, pictorial dictionary allows them to associate pictures of various subjects with terms and definitions, and thus help them to memorize and learn about the subjects. In our project, we developed a prototype of online pictorial dictionary focusing on the subjects of plant and animal. We believe that visuals are critical elements that help the children to learn about these subjects.

Our approach used in developing the dictionary is based on open content. Specifically, photos and information were gathered from online sources with open content licenses. Today, there are a large number of digital pictures available with the Creative Commons license, e.g. those available from Flickr [3], Wikimedia Commons [4], etc. These electronic resources were made available by people from around the world and are essentially free for reuse and distribution. This permits authors to create and publish new content by using these media as composing components. Although we used some newly created photographs in developing the dictionary, the majority were digital pictures collected from existing online sources available under the Creative Commons license. Citations to the original authors were maintained.

The MediaWiki tool (http://www.mediawiki.org/) was then used in authoring the information associated with the pictures. Wiki is an excellent tool for authors to efficiently compose content that can later be shared with others through a website. With its available template, category and file upload facilities, the tasks of authors are significantly simplified. In addition, MediaWiki can facilitate collaborative content authoring. In developing the dictionary, template facility was utilized. Specifically, each pictorial entry has a specific attribute-value structure and thus can be defined as a wiki template. This significantly reduced the author's effort in composing the information for each entry. The created pictorial entries were then organized into categories. We defined categories based on several facets including by alphabets, by species, by classes, by habitat, etc. Initially, 330 pictorial entries that include terms, scientific terms, definitions, synonyms, etc. were created for the prototype. The created database was also made available under the CC license. Figure 2 shows a pictorial entry from the Pictodict wiki site (http://pictodict.opentle.org/).

4. Capturing and Modeling Junior Pictorial Dictionary Content in OLPC

4.1. Architecture and Implementation

This section describes a system architecture for extracting and encapsulating the created junior pictorial dictionary (JPD) database in OLPC machine. The architecture consists of three major components: MediaWiki Component, Extraction Component and OLPC Component. The architecture is shown in Figure 3.

1) JPD MediaWiki Component is used in constructing the dictionary content. In addition to content authoring, MediaWiki also allows browsing and access to the created content via a Web browser. It also provides collaborative support for content editing. The JPD content is stored in a MySql database, whose content is made available under the CC license.

2) JPD Extraction Component is used in extracting the data from the MediaWiki database and encapsulating the data in an OLPC database, i.e. SQLite (http://www.sqlite.org). It also includes
exporting and thumbnail creation of the image files. After the data extraction process, the JPD database is more compact and is suitable for the OLPC machine.

3) JPD OLPC Component allows storing and accessing the JPD content under the OLPC machine. It consists of the JPD database stored in SQLite, which is a database system bundled with the OLPC platform. The design of the JPD database in OLPC machine focused on compact size and simplified structure. It used SQLite version 3, which supports most commands from the SQL standard and provides a set of APIs to facilitate application development. This component also includes a JPD user interface, which allows browsing and access to the JPD content based on the Sugar user interface environment.

4.2. Data Extraction and Modeling
4.2.1. MediaWiki Data Extraction

We developed a data extraction software module that extracts the required information from the MediaWiki database. The extraction process is described in details as follows.

1) Obtaining list of categories. This step includes recursive traversing of the category structure by identifying a root node. The category titles and sub-categories are obtained from each category page. Duplicated categories, which are those appeared as sub-categories of serveral categories, are removed.

2) Obtaining list of pages under each category. This step includes identifying categories that are leaf nodes, which are categories that contain list of content pages instead of sub-categories. When a leaf-node category is found, the list of content pages under the category is collected. Duplicated pages, which are those appeared as content page under several categories, are removed.

3) Obtaining the information under each page. Each content page of the JPD is stored in form of a template instance. The use of template helped to reduce the author's effort in structuring and formatting the content. Thus in extracting the information from a template instance, template structure and format must be removed. Specifically, only attribute values are extracted based on the defined structure. A template entry begins with '{' and ends with '}' characters. Detecting for a template is done recursively in order to find sub-templates used under a template.

4) Exporting images files from the MediaWiki database. In order to permit offline access from OLPC machine, image files must be exported from the MediaWiki database and stored as part of the OLPC database. MediaWiki stores each image file in a separate directory. The directory name is based on an encoded name of the image filename. MediaWiki uses the MD5 hashing technique for the encoding. MD5 is a form of cryptographic hash, where the output length is fixed. For example, if an MD5 output of an image filename is 'f8dde4e45b2800f9a18de1072db2667', the directory location for the image file is typically '/f8'.

5) Creating a thumbnail for each image. Once the directory location of an image file is known, it can be retrieved from URL 'http://[Wiki_site_name]/images/{directory_location}/{image_filename}'. A thumbnail was created for each image with a size within 100x100 pixels. Our implementation used 'wget' utility for image downloading and ImageMagick's 'convert' utility for thumbnail creation.

4.2.2. Design of JPD Database for OLPC

The JPD database for the OLPC platform is stored in an SQLite database, which is a database engine bundled with the OLPC platform. The design of the JPD database for OLPC machine focused on compact size and simplified structure. It used SQLite version 3, which supports most commands from the SQL standard and provides a set of APIs to facilitate application development. Some extraneous information stored in the MediaWiki database is removed for simplification. The designed data model is shown as an ER diagram in Figure 4. The model can be transformed into four tables to store the following information: page information, category information, category-page relationships and category-subcategory relationships.

Figure 4. Data model of the JPD database designed for the OLPC platform

5. User Interface Design
5.1. Design and Layout

The main users of the JPD application for OLPC machine are children. Thus, the design of its user interface must be easy to use, attractive, interesting and include some assistive tools. The designed interface must utilize the Sugar UI environment such as layout and buttons. In addition, it must be designed to work well with the small screen of OLPC machine. The number of clicks and keyboard typing should be minimized. Vivid colors and graphics, such as image thumbnails, are largely utilized to make the software more attractive and interesting to children. The design layout for the JPD user interface can be divided into three main areas as follows. The layout design is shown in Figure 5.
1. Navigation Tool Bar
2. Path Bar
3. Working Space

5.2. Navigation System

The program allows three data access methods to the content, which are described as follows.

5.2.1. Tree-based Navigation

This form is the main navigation system of the JPD where the user can browse the content hierarchically based on the defined categories. There are two modes for tree-based navigation.

1) Graphical tree-based navigation uses the working space area to visualize the tree structure. The user can see up to 3 levels of category depth at one time: current, child and parent nodes. The current node is shown in the center of the working space area. The child and parent nodes are distributed around the current node. The connections are shown as directed arrows. Incoming arrows to the current node indicates links from parent nodes. Outgoing arrows from the current node indicates links to child nodes. Three types of nodes may be distinguished: root, category and page nodes. Starting from the root node, selecting a category node allows the user to drill down the category structure. While selecting a page node allows the information under the page to be displayed. Nodes are displayed in the working space by image thumbnails. Links to a current node are displayed as directed arrows. Incoming arrows to the current node indicates links from parent nodes. Outgoing arrows from the current node indicates links to child nodes. Figure 7 summarizes linking structure associated with each type of node.

2) Textual tree-based navigation uses the path bar to display navigation path from the root node to the current node. The user can view up to n levels of category depth at one time, where n is the depth of the current node from the root node. User can also click on any intermediate node for direct access to the node. This can be used efficiently in conjunction with the graphical tree-based navigation.

5.2.2. History-based Navigation

This form of navigation is similar to the back-forward browsing of a Web browser. The user can access each node that was previously visited. Up to 20 previously-visited nodes were stored as access history records. The user can use the navigation tool bar to go back and forward the records in the access history.

5.2.3. Keyword-based Search.

This form provides a direct access to a node if title of the node is known. The user can also use keyword-based search to see list of nodes whose titles matched with the search text.

5.3. Working Space

Working space is the focal point of the user interface that allows the user to browse and access the JPD content. Specifically, the working space area is used as a display for visualizing the hierarchical category structure. Our design allows the user to see up to three levels of category depth at one time: current, child and parent nodes. The current node is always located in the center of the working space area. The child and parent nodes are symmetrically arranged around the center node in a circular layout. A maximum of eight surrounding nodes are displayed per screen. Figure 6 shows the symmetric circular layout design with different number of surrounding nodes.

Figure 6. Symmetric Circular Layout Design for the OLPC JPD Navigation System

Three types of nodes may be distinguished: root, category and page nodes. Starting from the root node, selecting a category node allows the user to drill down the category structure. While selecting a page node allows the information under the page to be displayed. Nodes are displayed in the working space by image thumbnails. Links to a current node are displayed as directed arrows. Incoming arrows to the current node indicates links from parent nodes. Outgoing arrows from the current node indicates links to child nodes. Figure 7 summarizes linking structure associated with each type of node.
When a page node is reached, the information under the page is displayed. The page display is designed to minimize scrolling. Figure 8 shows the designed information page displayed.

![Figure 8](image)

**Figure 8.** Display screen for a page node

### 6. Evaluation

Assessment of the OLPC JPD prototype was conducted both in terms of usability and usefulness of the program. The prototype was evaluated by 28 end-users who were children aged between 8-15 years (Grade 3-10). The group consists of approximately equal number of male and female kids from various provinces and with different degree of computer skill. The purpose of the assessment was to ensure that our prototype and user interface design work well for the children. In evaluating the prototype, the children were asked to play with the program and to rate their satisfaction with the program in the scale of 1 (strongly disagree) to 5 (highly agree) given eight evaluation criteria: four related to program assessment, three related to user interface assessment and one related to overall satisfaction.

The evaluation results are summarized as shown in Table 1. The results are shown in terms of average rating score and percent of positive responses (rating score of 4 or 5). The children rated user-friendliness of the program in a high degree (avg. = 4.07, SD = 0.94) with 75% of positive responses. The rating score was also in a high degree for program attractiveness (avg. = 4.21, SD = 0.79, 79% positive responses), fun (avg. = 4.43, SD = 0.63, 93% positive responses) and gaining knowledge (avg. = 4.59, SD = 0.75, 93% positive responses). For the user interface, the rating score was in a high degree for the provided toolbars and icons (avg. = 4.21, SD = 0.88, 79% positive responses), font face and size (avg. = 4.11, SD = 0.96, 75% positive responses), and the graphical browsing interface (avg. = 4.25, SD = 0.80, 86% positive responses). Particularly, the children satisfaction level with the overall program is in a high degree (avg. = 4.67, SD = 0.55) with 96% of positive responses.

![Figure 9](image)

**Figure 9.** Evaluation results compared between children in primary school level and middle and high school level

### Table 1: Evaluation results of the OLPC JPD prototype

<table>
<thead>
<tr>
<th>Evaluation Criteria</th>
<th>Avg. Score (SD)</th>
<th>% of Positive response</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Program</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Program is easy to use</td>
<td>4.07 (0.94)</td>
<td>75%</td>
</tr>
<tr>
<td>2. Program has attractive appearance</td>
<td>4.21 (0.79)</td>
<td>78.6%</td>
</tr>
<tr>
<td>3. Program is fun and interesting</td>
<td>4.43 (0.63)</td>
<td>92.9%</td>
</tr>
<tr>
<td>4. Program helps me to learn more</td>
<td>4.59 (0.75)</td>
<td>92.6%</td>
</tr>
<tr>
<td><strong>User Interface</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Toolbars and icons</td>
<td>4.21 (0.88)</td>
<td>78.6%</td>
</tr>
<tr>
<td>6. Fonts</td>
<td>4.11 (0.96)</td>
<td>75%</td>
</tr>
<tr>
<td>7. Navigation</td>
<td>4.25 (0.80)</td>
<td>85.7%</td>
</tr>
<tr>
<td>8. Satisfaction with the overall program</td>
<td>4.67 (0.55)</td>
<td>96.3%</td>
</tr>
</tbody>
</table>

When comparing the satisfaction level of children between grade 1-6 (primary schools) and children between grade 7-12 (middle and high schools), the results shows higher satisfaction degree for the former group in most evaluation criteria as shown in Figure 8. This indicated that the program and user interface design works particularly better for younger children. From our observation, the younger children were particularly more excited and engaged to see pictures and the information of various animal subjects. The graphical browsing interface worked well for both the younger and the older children.

### 7. Conclusion

In this paper, we described our development of a junior pictorial dictionary prototype for the OLPC platform using the open content model. It focused on content creation method using a Wiki tool and utilization of resources with open content licenses. A data extraction framework for encapsulating the
Wiki content into an OLPC database was described. Finally, a user interface design for Sugar UI that visualizes category structure and allows efficient browsing of pictorial content was presented. Initial evaluation results confirmed that our prototype and user interface design was relatively effective for children. Together, these provide a framework and case study of reusing and representing open content to fit different audiences and platform that they were originally created for. Utilizing open content in OLPC machine can lead to a more diverse use of the content and allows it to have a larger impact on children education.

Some future works include expansion of the JPD database both in terms of number of subjects and attributes associated with the subjects. Integration of some advanced search tools is also planned. This includes searching based on subject attributes, such as color, size, habitat, etc., and combining image searching tools.

8. References


[9] Creative Commons Home Page: http://creativecommons.org/