
Introduction to the Open Clip Art Library

For the SVG Open 2005 conference

Jonathan Phillips, Center for Research in Computing and the Arts
[<http://crca.ucsd.edu/>] <jon@rejon.org>

The Open Clip Art Library (OCAL) [30] is an on-line community-driven project aiming to create an archive of user contributed clip art that can be freely used. This document presents an overview and then explains the project as a library, a set of tools used to interface this collection, and as a supportive community-users, artists of varying skill-levels, and "librarians." This document will be concluded with a scan of current development practices and a general discussion of the project's direction including where help is needed for improvement.

Table of Contents

Overview	1
History	2
Views of the Project	2
Library: A Public Domain SVG Collection	2
Tools: Interfaces to the Library	4
Community: Users, Artists and Librarians	6
Development Practices	7
Roadmap of Future Features	7
Large Scale Goals	8
Conclusion	9
Bibliography	9

Overview

What does the "Open Clip Art Library" mean? Basically, clip art are graphical components which can be used to compose other graphics. Other definitions add more nuances to this and imply certain media, such as dictionary.com which states that clip art are "Ready-made pieces" of "illustrations, borders, and backgrounds." [8] Wikipedia's definition sheds light upon the origin of the term clip art as pieces having been "physically cut from pre-existing printed works". Overall, these definitions show their age as clip art is now found on various inexpensive collections like Nova Development's Art Explosion 300,000 [28] which contains an ample amount of varying qualities of clip art repackaged on several CD-ROM disks for around \$30~80 USD. Graphic designers and artists use these art fragments to make compositions rapidly rather than having to build images tediously from scratch. As with software development, why reinvent the wheel? While the roots of clip art are established in more traditional media -glue, paper and scissors - the metaphors of the modern computer now expand "clip art" to be more contextually-rich with data embedded inside which describe an image's content. Thus, clip art becomes bite sized media useful for composing rapidly, traditionally, and/or algorithmically.

So then, what is so open about the Open Clip Art Library? The project is open because of its affiliation with the Open Source [35] software and cultural movement. Thus, this project attaches to the core values of Open Source culture like releasing early and often, making the source available, and rewarding contributors with praise. [38] However, where this project differs with Open Source is that it is not develop-

ing one piece software. Rather, development is on the infrastructure which supports user submissions of clip art including the website, tools for manipulating the clip art, and the actual content, clip art. Also, the use of Open also links in with Open Content, or media in which the content is publicly accessible and modifiable. [49]

The project title also latches onto the library metaphor because libraries exist "to collect and organize information, make access to knowledge more democratic, and preserve the record of ideas for future generations." [2] Generically, a library is a place where media is stored, also called a collection. While this could be an individual's collection, the library as an institution is most accurate as it refers to publicly accessible collections of media. Similarly, the Open Clip Art Library is a publicly accessible collection of clip art. A further extension of this metaphor is to map the role of librarians onto software developers. Traditionally, librarians help organize, repair, check-in, and check-out media. Likewise, the Open Clip Art Library's core developers, or librarians, assume this role in the tending of clip art.

Therefore, recombining these concepts composes the primary goal of the project: to create an archive of user contributed clip art that can be freely used. While continuously working towards meeting this goal, the project has grown over the last year to collection of 3400 clips by over 200 artists. The monthly package released now is about 50-60 megabytes and is carried by several major Linux distributions like Debian, Gentoo, and various Red Hat RPM packagers. [40] Also, the project is affiliated with the Open Office.org and their 20-30 million users [32], KDE's KOffice [22], the Open Source desktop publishing tool, Scribus [43], the Open Source drawing program, Inkscape [17], freedesktop.org [10], who graciously provides hosting to the project, The Open CD [45], and the Creative Commons. [3]

History

The threads from the project were initially sewn by the vector editor Sodipodi's own collection of user submitted images, and most importantly by the Sodipodi Flags project, spearheaded by Christian Schaller [42], where United Nations flags were gathered together as one package. As this package grew on the Sodipodi website, the interest to create other packages increased. Another thread that helped to start the Open Clip Art Library is the dynamic development of Inkscape. As this project grew during its first year of development, the amount of artwork generated by users increased as well as the addition of new clip art to new releases. Once several full slide shows were added, thus bloating a coming release by several megabytes, the clip art needed to be extracted from Inkscape into another form.

So, Bryce Harrington, Ted Gould and the author of this paper brainstormed about what to do over a few months while Inkscape's development continued. Prior to committing to a new project these core developers did look around at the other projects in the landscape. KDE had the Klip Art project [21] which appeared to be inactive. Then, they found a few photo-based sites like PD Photo [34] and OpenPhoto.tk [33]. Deviant Art's [9] vibrant community enticed them, yet the licensing did not seem clear for user submissions and many of the works were finished compositions – far from the raw nature of clip art. These developers also looked at closed projects like the Microsoft Office Clip Art website [26] and the aforementioned Art Explosion 300,000, but these projects did not solve the need for a fully user-driven clip art library which would make it easy for artists to submit their work. This led to starting a fresh project allied with freedesktop.org, Inkscape, Scribus and Sodipodi. After incubating the project as clipart.freedesktop.org, Chad Smith, an OpenOffice.org developer donated the domain name openclipart.org which in turn shaped the direction of the project. After developing many iterations of the current logo, the community voted to have Andreas Nilsson's design as the primary project logo and the name for the project to be formally titled "Open Clip Art Library."

Views of the Project

Library: A Public Domain SVG Collection

The initial decision to make a clip art library focused on several issues such as what kind of clip art would be allowed – should the library accept any file format of any size by any artist? Also, what could a small team of developers actively maintain and develop? Having learned from past projects, the core

developers decided it best to do one thing well rather than doing many things halfway as Brian Kernighan in the *Software Tools* states as “controlling complexity is the essence of computer programming.” [19] The core developers of OCAL learned from developing Inkscape that contributors often burn out from trying to conquer the world by doing everything when they could have done one thing well. This is the case for groups of people because they “function more effectively if their goals are clear to members.” [23] Adhering to this principle has been the path to success for OCAL.

File Formats

Rather than support all graphics formats, including the popular, JPEG, GIF, and PNG, the core developers decided that the World Wide Web Consortium's (W3C) Scalable Vector Graphics (SVG) [41] would be the primary format for the project. It serves as the mother format (“master”) because SVG files allow for extremely descriptive content which makes managing a library of files much easier. Also, the modern day requirements for scaling of one type of file for multiple media such as print, web, and video is extremely important to content providers. An example “scalable media” would be a file format containing a movie preview which would playback on cell phones, laptops, large flat panel displays, and on movie screens. Similarly, the SVG format has several features like the aforementioned scalability, reproducibility, modularity (object-oriented), precision, and abstraction (graphical vectorization), as has been shown elsewhere [37], which allow for the clip art to be scaled, stretched, and re-composited while maintaining the high-quality nature of the original. Finally, after all this conversion, then this format can be used elsewhere, or downgraded in quality to be used as a JPEG, GIF, or PNG.

SVG is the standard file format for OCAL because it is an open standard. While Adobe/Macromedia's Flash is installed on around 98% of all web browsers [25], it is a closed binary format even though it is well documented. [24] On the other hand SVG, a World Wide Web Consortium (W3C) open standard format, [41] is not currently supported so well on web browsers. Yet, it has the functionality of Flash and is well documented on-line. Also, like its sibling HTML, it can be authored without an expensive monolithic piece of software. One may write SVG by hand, or in a simple text editor.

Beyond this standards war, SVG files are well suited for OCAL because they are text-based XML files which allow meta-data to be internally embedded. One of the project's early developers, Richard Worth, investigated [50] what type of meta-data should be embedded inside the Open Clip Art Library's files, balancing features with file size. Ultimately, the project went with Creative Commons' meta-data standard. While Adobe built the Extensible Meta-data Platform, XMP, a strong standard for enterprise-level media, the specification is far too large for OCAL's needs. [4] The project only needs a basic meta-data standard that doesn't bloat the overall size of each SVG file. Also, the Creative Commons meta-data combines standard Resource Description Framework (RDF) XML, the standard Dublin Core meta-data [6], and Creative Commons' own usage.

Entire Library

To zoom out from an individual SVG file and scan the entire project, one will see that the Open Clip Art Library is really comprised of many meta-data rich SVG files. The decision to use meta-data though is primarily to aid in the searchability and categorization of these files in bulk.

If the project is truly a library, then there needs to be a way to submit clip art into the library and a way for the public to check out the images. This is handled currently through the website, www.openclipart.org. There is a simple entry box where anyone can post SVG files. The primary criteria is that a submitted file is SVG, or a package of SVG files with meta-data fields embedded, or in an accompanying `metadata.rdf` file. Then, monthly, the clip art fills up in an incoming folder where it is parsed late in the month, broken files are flagged, and the entire repository is updated.

For clip art checkout, the web site also serves the public well, as there is a graphical browser written by Alberto Simões and a keyword search written by Nathan Eady which allows for the library to be searched by the embedded keywords in the clip art. Once a search is found, a user can then view a selected SVG file, or save it to disk. Currently, the categories in the browser are static, but this is being improved so that the categories are dynamically selected by users based upon the keywords in the files.

The keyword search however provides early functionality of the future system because files are returned based on their internal meta-data based upon a query.

Licensing

In addition to selecting SVG as the default file format, all clip art must be declared as Public Domain. Originally, the developers of the project thought that clip art could be governed under any license as long as that license or a URL to it were included in the SVG file's metadata. However, as the discussion continued, allowing all licenses violated the modicum of not doing one thing solidly well. Also, supporting closed copyrights did not interest the developers, as there already are many projects which do this. However, using the standard software licenses like Gnu Public License (GPL) and Lesser Gnu Public License (LGPL) [12] did not make sense to use for artwork as including the previous source of all artworks in a composition is not always possible nor is it always possible to include a web URL to where one could find the source of an image. And, if the project used the GPL users of the clip art would be forbade from from making money on that creation, any uses, or derivatives of it. Also, since promoting SVG is one of the goals of the project, OCAL had to match similar clip art projects like Art Explosion 300,000 and Microsoft Office's free clip art in terms of making royalty free or public domain artwork. In the end, OCAL's rationale is that it is better to get one's artwork out into the world, used, and seen rather than for it to sit idly on a web site unusable because of copyright restrictions: Users should deal with clip art, not licenses.

In order for this scheme to work properly, the submission process on the website requires a user to either have a link to the Creative Commons public domain declaration [5], or explicitly check a check box on the submission web form which states: "Yes, I agree to release my clip art to the Public Domain and assert that it is not legally encumbered by trademarks or reserved rights." This places the responsibility onto the submitter and also reminds this person not to submit illegal or trademarked symbols to OCAL. Still, unfortunately, some people do persist by submitting illegal imagery and is why the project must have people and tools to check the work at the end of the month.

Tools: Interfaces to the Library

This section will first discuss the primary interfaces to the Open Clip Art Library, and then will end by outlining other tools used to maintain the collection. Currently there are three primary interfaces. There is the already discussed web site, there are packages that are released monthly that contain the entire collection, and there is a third new system called the Document Management System, or DMS, which will become the core of the web site enabling the creation of command line and graphical interfaces.

Website

This document has already discussed several features of the web-based client. Additionally, the importance of this project in supporting open documented standards like W3C XHTML 1.1 and CSS 2 is underlined. It is important to respect these standards in order to also promote another standard, SVG. Similarly, the underlying software of OCAL, the file types exported, and development tools used reflect this belief in open standards. Thus, the most visible content on the site is dynamically generated from PHP while the heavier parts are written in PERL, both well documented languages in the Open Source community.

The web-based interface is the primary way that the public comes into contact with the project, the artwork, and the development community. So, much time is spent on the design, layout, and style of the site. If artists are using the site, then the quality of the site very much shapes the type of people attracted to the project.

The main way that users submit artwork is from the main page of the site through the banner: "Read the Guidelines. Submit A File." What follows is a file selection box and a "Send File" button. The simplicity of this is to provide at minimum, one simple function to users. A visitor receives no hassle in submitting their file, or a package of files. What is hidden by this form is the script 'upload_svg.cgi' which parses a submission and checks to make sure that the basic meta-data fields such as author, title, and a

link to the creative commons public domain declaration are included. If these tests fail, then the user is sent to another page where this data can be entered. After the required fields are filled in and a user submits the file again, the SVG is patched with the new information and is then put into the incoming clip art folder for later review by a librarian. This process pushes much of the work onto the author of the file, yet provides an expanded interface if more information is needed. Something else to note is that this simple input field also accepts large packages with the meta-data from the files extracted into a file, `metadata.rdf`, which at release time, is parsed and the meta-data from that file are inserted prior to release.

Packages

Another interface to the project is the monthly Open Clip Art Library release. Currently, this consists of the entire collection in static directories, parsed and checked for validation, and then compressed into a tarball, zip file, and a Microsoft Windows-based NSIS installer. [29] While the long term plan is to build packages on the fly based upon user queried dynamic categories, it is necessary to make the traditional package release monthly in order to both keep the project on a steady development process and to release the newest clip art to the public regularly. There are many release schedule systems such as Gnome's six month time-based cycle or Inkscape's feature-based schedule, but the rate of clip art submissions and lack of technical implementations demands a monthly maintenance release.

Keeping on this metronomic release schedule helps to sort out concerns with tools in active use and development. Also, releasing often helps catalyze realizations about how to deal with so much content and technical decisions like where standard installation of clip art should be on different platforms. Recently, this last issue became an issue as consolidation of content shared among applications, like Abiword, Open Office and Inkscape, has become important. Each of these applications has their own clip art folders within their own install directories. Because OCAL is external from these projects, the standard clip art directory of `/usr/share/clipart` is being made the default among various Open Source applications on UNIX distributions so that systems know where to look for images thus saving space for a user. This problem though did not arise until the project actually released around 14 packages which is about one year and two months of releases.

While this system has many strengths, there are downsides. The current release processes are way too complicated. With each release, as new issues arise and are solved, the release process is expanded. Currently, the process involves 23 steps which are heavily taxing the release process. Also, with each release the package size increases several megabytes which becomes a problem for distributions who package and release on CD-ROM media. Already Fedora Core and others require greater than three CD-ROM disks for installation. If the Open Clip Art Library releases a 300 megabyte package, distributions will deny this size of package as one standard data CD holds approximately 700 megabytes and project packages are growing quite large with recent graphical interfaces. This package size will eventually become a problem for the project's bandwidth at freedesktop.org.

Document Management System (DMS)

The good news is that the plan from early development has been to shift the entire system to use what this writing calls the third interface, the Document Management System (DMS), spearheaded by Bryce Harrington. [14] Bryce writes:

DMS is a system to provide a daemon-based service that encapsulates a document management system. Think of it like an email server, but instead of sending emails with to, from, and subject headers to it, you send documents with meta-data. This is invoked via the 'dmsd' daemon, which runs continuously, accepting connections from clients, processing requests, and maintaining the document store.

The protocol used for communicating with dmsd is the 'Simple Object Access Protocol', or SOAP. SOAP is an XML-based protocol and is supported by a wide variety of languages. This means you can construct client interfaces in Perl, PHP, Java, Python, C++, or any other language that has a SOAP implementation. Perl's SOAP im-

plementation is called SOAP::Lite and is being used for creating simple command line tools. [14]

This new service runs in the background on a server and manages document check-ins, check-outs, file structure, allows for file versioning and other concerns that users should not be bothered with.

DMS can be accessed in several ways. On the command line it can be used like a version control system [47], or it can be accessed by the hidden code behind the simple web based search form on the project's main web page. Graphically, it can be connected to in order to display clip art on the fly.

Like other tools that are used by OCAL, DMS uses another of Bryce's inventions, SVG::Metadata, a Perl module which parses an SVG file for Creative Commons meta-data and can either get or set this type of meta-data. [15] This allows for a client to connect to DMS to validate new SVG files on input before being placed into a DMS repository.

Other Tools

In addition to these three main interfaces, there are also several tools, some of which are based upon SVG::Metadata, which are used to filter content and update the library at the end of the month. As clip art fills up in the incoming folder each month, it has already been validated with necessary meta-data fields. If there are any problems however with a file, the meta-data, or general structure of the document, then this needs to be discovered before inclusion in the collection. There are several scripts, primarily created by Nathan Eady to check the incoming clip art. Also, there is a script, clipart-authority-control, which is "for managing the keyword meta-data on the collection as a whole... This script also produces author-submission and keyword-usage statistics and the keyword index for the on-line search tool," writes Eady on the Open Clip Art Library wiki. [46]

There are also other scripts like 'parse-svg-validate-log.pl' which produces a log that shows which SVG validate in the library and 'generate-thumbnails.pl' that uses Inkscape to create PNG thumbnails for the web-based graphical browser.

Also, there is another class of tools aiding the primary interface to OCAL. An example is the new clip art request system. In addition to collecting any type of SVG submission, experience gained by studying Wikipedia.org's request page, [39] how that project allows anyone to get a username in order edit Wikipedia.org, and then set placeholders for requesting content, exemplifies that it is good to provide a mechanism to allow for requests. The original system used to be hosted on the project's wiki, but after it filled up with too many requests it became quite unusable as discussions ensued between requesters and artists. This new web based system eases that information disorder so that needs can be filled easily and visibly.

Community: Users, Artists and Librarians

Over 200 people from the United Kingdom, United States, Romania, Germany, Canada and several other countries daily assume roles as users, artists, and librarians to form the community of the Open Clip Art Library. This community is responsible for submitting the approximately 3400 submissions that comprise the Open Clip Art Library [44] where the top submitter in terms of quantity, Danny Allen, has posted 952 pieces and a close second, Nicu Bucelei, a core developer, has submitted some 472 images. The community is what truly powers the Open Clip Art Library, and as such is a dynamically shifting group of people always changing roles based upon individual and group needs.

All members of the community are users of the project. They use the website, download clip art, use the clip art, or install the clip art from a package or an installer. A user uses the website to fulfill some want or need. Consequently, becoming a user is the general entry point into the project.

Through use of the various interfaces outlined previously, and primarily the web-based interface, users who submit clip art, or who intend to submit clip art assume the role of artist. This is not a static defini-

tion, but a simple guideline as generally anyone who creates or modifies artwork for the project is an artist.

When users and artists need assistance with the project's tools and interface they either assume the role of librarian or work with a librarian. The librarians are what are traditionally called developers in software projects. However, with the shift of this project from purely Open Source to Open Content there is a need to differentiate between the Open Source developer and the Open Content developer. Hence, the librarians assist all types of users, work on the infrastructure of the project, fix bugs, develop strategies, write documentation, make timely releases of new clip art, and function as the traditional developers of software code.

Development Practices

The actual process of development on the project consists of approximately 5-10 active librarians. [48] And, in keeping with the keep it simple philosophy of the project, the development style keeps “low social barriers.” This means that the walls in order to gain access to the project's code and development tools are as low as possible to keep structure for the project, yet encourage new development and empower people to transition to other roles in development. One place this has been learned is from Inkscape where the policy for a new developer to receive source code access is to provide “one patch” in order to get direct code access.

This style of low stress development places much faith in the community and relies on the reputation of the people involved. In the Open Clip Art Library there is also not a huge stressful monthly push to conquer the goals set forth on the project's road map for each release. This is due to the project's infrastructure which collects clip art automatically during a month. If goals for the release are not met the project is still successful because a package is ready to be released monthly with the newly collected clip art. However, this system could lend itself to complacency or stagnancy. To the contrary, the emphasis on the monthly release process and allowing clip art to build up rather than focus heavily on a goal-based development strategy has helped the project to steadily increase actual development while clip art continues to be submitted. This cycle also serves the project well because monthly distribution of press releases helps find new artists continually.

Roadmap of Future Features

While the current functionality of the Open Clip Art Library is presently sufficient, there are further features planned and future discussions that need to take place in order to improve the project. On the technical side, work on replacing the core clip art management software with DMS is where most development energy is directed. The old web site and clip art submission system have become too specialized with each release. The current site needs serious usability revisions and style updates to encourage more professional artists to submit work. In addition to the previously discussed benefits of DMS like an abstract SOAP client interface and the ability to check in and out clip art, the move to DMS will also enable easier development on making the library available through new interfaces. Thanks to Google's Summer of Code, a project launched by Google to fund 200 students with \$4500 in funding to work on proposed additions to Open Source projects for summer 2005, [13] there is funding to support a new developer, Greg Steffensen. He is commissioned to make the first Open Clip Art Library interface. His proposed work is to make a graphical browser of clip art for Inkscape. The goal is that this will also be usable as a standalone browser for Open Source desktop projects like Gimp and Scribus.

Other future features that are on the development roadmap are a user management system to help track artists and enable security so that users can assume a new role to review, comment on, fix, and update clip art all from a new web based interface. This is not meant to be necessary for submitting artwork, but is a service provided for people who would like to track their contributions, and also for those in the community who would like access enough to edit the current clip art in the project by inserting keywords, descriptions, make language translations, and would like to fill clip art requests.

Also, to address the current temporary categories, the use of keywords in the clip art meta-data will be

utilized through DMS to allow for dynamic hierarchies of content that will filter clip art based upon a user's search of OCAL's collection. According to the future development roadmap users will be able to have "My Favorite Images" saved that would use a set of preferences to build a customized and browsable set of categories. This system would then allow for someone to edit their own viewed images and provide their own sense of censored or banned imagery. After this feature had been thought of, the Linux distribution Debian specifically needed a way to edit out Nazi imagery from the current collection so that they could include the released clip art legally in Germany. The current solution solves this by providing a feature in one of the release scripts which can filter out illegal keywords. Until the better solution is implemented, Debian will continue to provide custom legal packages of the most up to date OCAL packages. Initially, Debian developers wanted OCAL to delete the "offensive" imagery, yet OCAL did not want to get into the habit of censoring or labeling content, so artists and users are now responsible to tag submitted imagery properly so that users can censor their own searches.

On the redundancy side, the project needs to develop a strategy for mirroring content as the size of the repository scales in volume of users, submissions, and downloads. The site hosted by freedesktop.org still is running strong, but OCAL needs to be prepared for throughput increases with reliance upon DMS's web services. When this does become a problem, several private individuals and UC Berkeley's Center for New Media have already made overtures to provide a mirror of the project when needed.

Another primary push that needs to happen is further interoperability with OpenOffice.org (OOo). Right now, one of the oldest (and most commented on) outstanding feature request for OOo is for SVG import. [18] There is already preliminary export of SVG, but a way to get the images into OOo is of importance, as several developers like Novell's Michael Meeks are actively working on the OOo gallery. Rather than implementing the best feature, an SVG gallery, Meeks has had to convert the collection of SVG files into fixed resolution PNG files, thus removing the benefits of a clip art library maintained as SVG files.

Along with adding SVG import should be an upgrade to the gallery which imports the clip art to OOo so that it can either be loaded from disk, or browsed via the DMS SOAP API. Ultimately, as OOo is one of the first supporters of the Open Clip Art Library there needs to be better integration and promotion between the projects to maximize networks.

Some other fun projects that are on the roadmap are to develop web spiders to crawl the web for old public domain clip art and automatically input this into the project. Several commercial distributions of clip art like the Nova Development Art Explosion 300,000 have built their products on public domain bitmap and vector graphics. This type of original public domain imagery needs to be sought out, converted to SVG, enriched with meta-data, and placed into the collection for use by all.

Further out on the roadmap, support for other parts of the SVG specification need to be thought about and supported. One developer, Benji Park has made an SVG Font and would like to contribute that to the project, yet there is currently not a strategy on how to support fonts. Also, there needs to be work on on-the-fly converters that will allow people to convert from SVG to PNG, BMP, WMF, PDF, and PS from the web-based interface. This will help to further provide a resource for users who might not have a way to import SVG, but that can accept a bitmap file.

Large Scale Goals

Beyond the technical roadmap, there are also some large scale goals and themes that the project has engendered. These are not all rigidly defined and several are more of questions than goals. With regards to the modicum of doing "one thing well, rather than many things halfway." While this might sound like foundationalism, this goal is a reflection of the thinking that as the project attains a higher status with abstracted technology like DMS, then other projects could take on similar efforts with different media. Just as easily as there is the Open Clip Art Library, there could be an Open Media Library, or an Open Sound Library. With some extra converters and ways to embed meta-data in or along side other media formats, the same type of wrapper as the Open Clip Art Library could be employed. The original thinking of the project was to do the basics well, and then apply these to the larger media which have far more concerns like storage size, time-based, and multiple-media needed support. Since 2D images are

considered one of the most basic media beyond text files, it is a great testing ground for how to maintain a larger library with more technically complicated media.

Also, a question and hope for this project is to actually create good and meaningful clip art. It is quite problematic to say, “Open Clip Art Library only accepts good art.” It is common rhetoric that the definition of “good art” has as many definitions as there are people. Rather, the importance of supporting a multiplicity of viewpoints is what is of the most vital concern for the project. Rather, this statement should be reshaped to say, the “Open Clip Art Library catalyzes creation of good diverse content.” This needs to be explored much further, but might not until the community of artists grows much larger.

Another abstract goal of the project is to help make the free desktop look better aesthetically. While not advocating Human Interface Guidelines like the Gnome Project or Apple, simply by giving a place for artists to submit artwork and encouraging others to fill requests and develop a community will raise the bar on the type of artwork gathered on the desktop. Already different projects like MultiSync [27] and various conferences come to the Open Clip Art Library asking for artists to develop logos and other project artwork.

Beyond inventing technologies needed for this type of on-line repository is to also support new standards which are viewed as positive to support and to promote. Helping to flesh out implementations of the Creative Commons licenses and use of their meta-data scheme not only solves a major organizational hurdle for the project, but helps the Commons. While this is not a defined goal, it is good for this project to help pioneer implementations so that as more media is developed, other projects have a good Open standard to use for their projects.

Conclusion

In closing, the coming year for the Open Clip Art Library is about growing. Now that the project has developed a community it is time to build the aforementioned “future” infrastructure to support the increasing number of submissions. It is also time to improve the interfaces to further empower the project’s users. As the first generation infrastructure is replaced with the new DMS-based core and the total amount of the clip art exceeds 5000, the project will face barriers to pierce, and in another year the project and community will be that much stronger.

Development on new infrastructure alone and collecting more submissions cannot alone grow the project. Rather, the project needs new users. New user’s should get involved in the project because it is small enough to allow a new user, artist, or developer great latitude in contributing. There are many entry points for all different types of users of varying skill levels. If one can create images, find others, or modify pre-existing images, then submission of this work is simple. Whereas large software projects like the Gimp have huge learning curves as prerequisites for development – coding standards, specific languages, and then the architecture of the software – OCAL is primarily a web-based interface and a collection of scripts. The tools in OCAL are similar to what new developers start with in learning programming. Hence, participating in the Open Clip Art Library has a low barrier for learning, joining the community, and is highly rewarding because of the low-stress monthly clip art releases.

As with many Open Source and Open Content projects, the Open Clip Art Library is open to all types of users and developers as different opinions generate innovative solutions and help to find errors in a group. [36] The more diverse the developer the better as this will help the project represent different types of artwork and vary the ways in which the content is accessed. As the Open Clip Art Library states, “put your clip art into circulation today at <http://www.openclipart.org>.”

Thanks to Bryce Harrington, Alan Horkan, Nicu Bucelei, Nathan Eady, and everyone else in the Open Clip Art Library community.

Bibliography

- [1] Art Explosion 300,000. *Nova Development*. 2005. University of California, Berkeley. 23 Jun 2005 <<http://www.novadevelopment.com/Products/us/ahw/default.aspx>>.
- [2] "British Library's strategy 2005-2008 ." *British Library*. 2005. University of California, Berkeley. 22 Jun 2005 <<http://www.bl.uk/about/strategy.html>>.
- [3] *Creative Commons*. 2005. University of California, Berkeley. 5 Jun 2005 <<http://www.creativecommons.org/>>.
- [4] "Creative Commons Metadata and XMP." *Creative Commons*. 2005. University of California, Berkeley. 18 Jun 2005 <<http://creativecommons.org/technology/xmp>>.
- [5] "Creative Commons Public Domain." *Creative Commons website*. 2005. University of California, Berkeley. 15 Jun 2005 <<http://creativecommons.org/licenses/publicdomain/>>.
- [6] "DCMI Metadata Terms." *DublinCore.org*. 2005. University of California, Berkeley. 19 Jun 2005 <<http://dublincore.org/documents/dcmi-terms/>>.
- [7] "Debian - The Universal Operating System." *Debian*. 2005. University of California, Berkeley. 1 Jun 2005 <<http://www.debian.org>>.
- [8] "Definition of word clip art." *Dictionary.com/clip art*. 2005. University of California, Berkeley. 14 Jun 2005 <<http://dictionary.reference.com/search?q=clip+art>>.
- [9] "deviantART: where ART meets application." *Deviant Art website*. 2005. University of California, Berkeley. 10 Jun 2005 <<http://www.deviantart.com/>>.
- [10] *Freedesktop.org*. 2005. University of California, Berkeley. 1 Jun 2005 <<http://www.freedesktop.org/>>.
- [11] "Gentoo Linux." *Gentoo*. 2005. University of California, Berkeley. 1 Jun 2005 <<http://www.gentoo.org>>.
- [12] *GNU website*. 2005. University of California, Berkeley. 16 Jun 2005 <<http://www.gnu.org/>>.
- [13] "Google Code: Google Summer of Code." *Google.com*. 2005. University of California, Berkeley. 6 Jun 2005 <<http://code.google.com/summerofcode.html>>.
- [14] Harrington, Bryce. "DocumentManagementSystem ." *Open Clip Art Library Wiki*. 2005. University of California, Berkeley. 15 Jun 2005 <<http://openclipart.org/cgi-bin/wiki.pl?DocumentManagementSystem>>.
- [15] Harrington, Bryce. "SVG::Metadata." *CPAN*. 2005. University of California, Berkeley. 16 Jun 2005 <<http://search.cpan.org/dist/SVG-Metadata/lib/SVG/Metadata.pm>>.
- [16] Held, Georg, et al. "Comparing .SWF (ShockWave Flash) and .SVG (Scalable Vector Graphics) file format specifications." *Carto.net*. 2004. University of California, Berkeley. 13 Jun 2005 <http://www.carto.net/papers/svg/comparison_flash_svg/>.
- [17] "Inkscape . Draw Freely." *Inkscape*. 2005. University of California, Berkeley. 1 Jun 2005 <<http://www.inkscape.org/>>.
- [18] "Issue 2497." *OpenOffice.org*. 2005. University of California, Berkeley. 1 Jun 2005 <http://www.openoffice.org/issues/show_bug.cgi?id=2497>.
- [19] Kernighan, Brian and Plauger, P.J. *Software Tools*. New York: Addison-Wesley, 1976.
- [20] "Keyword Organization." *Open Clip Art Library Wiki*. 2005. University of California, Berkeley. 15 Jun 2005 <http://openclipart.org/cgi-bin/wiki.pl?Keyword_Organization>.

- [21] "Klipart." *Koffice Project*. 2005. University of California, Berkeley. 5 Jun 2005 <<http://users.skynet.be/kborrey/klipart/klipart.html>>.
- [22] "The Koffice Project." *KOffice*. 2005. University of California, Berkeley. 1 Jun 2005 <<http://www.koffice.org/>>.
- [23] Locke, E.A. and Latham, G.P. *A theory of goal setting and task performance*. Englewood Cliffs, NJ: Prentice Hall, 1990.
- [24] "Macromedia Flash File Format (SWF) Specification License." *Macromedia.com*. 2004. University of California, Berkeley. 13 Jun 2005 <<http://www.macromedia.com/software/flash/open/licensing/fileformat/>>.
- [25] "Macromedia Flash Player Statistics." *Macromedia.com*. 2004. University of California, Berkeley. 17 Jun 2005 <http://www.macromedia.com/software/player_census/flashplayer/>.
- [26] "Microsoft Office Clip Art and Media Home Page." *Microsoft.com*. 2005. University of California, Berkeley. 7 Jun 2005 <<http://office.microsoft.com/clipart/>>.
- [27] "MultiSync - A Synchronization Tool." *MultiSync*. 2005. University of California, Berkeley. 13 Jun 2005 <<http://multisync.sf.net>>.
- [28] Nova Development. *Art Explosion 300,000*. Nova Development, 2004.
- [29] "Nullsoft Scriptable Install System." *NSIS website*. 2005. University of California, Berkeley. 16 Jun 2005 <<http://nsis.sourceforge.net/>>.
- [30] *Open Clip Art Library* :: www.openclipart.org. 2005. University of California, Berkeley. 1 Jun 2005 <<http://www.openclipart.org>>.
- [31] "Open content - Wikipedia, the free encyclopedia." *Wikipedia*. 2005. University of California, Berkeley. 1 Jun 2005 <http://en.wikipedia.org/wiki/Open_content>.
- [32] *OpenOffice.org*. 2005. University of California, Berkeley. 1 Jun 2005 <<http://www.openoffice.org/>>.
- [33] "Open Photography." *Open Photo website*. 2005. University of California, Berkeley. 5 Jun 2005 <<http://www.openphoto.tk/>>.
- [34] "PD Photo - Public Domain Photos (stock pictures, wallpaper, royalty free, clip art, etc)." *PD Photo website*. 2005. University of California, Berkeley. 1 Jun 2005 <<http://pdphoto.org/>>.
- [35] Perens, Bruce. "The Open Source Definition" in *Open Sources: Voices from the Open Source Revolution*. ed. Chris DiBona, Sam Ockman, & Mark Stone. Sebastopol: O'reilly & Associates, 1999.
- [36] Peterson, R.S. and Nemeth, C.J. . "Focus versus flexibility: Majority and minority influence can both improve performance" in *Personality and Social Psychology Bulletin*, 22, 14-23, 1996.
- [37] Phillips, Jon. "Vector Aesthetics." *Rejon.org*. 2005. University of California, Berkeley. 20 Jun 2005 <<http://www.rejon.org/writings/>>.
- [38] Raymond, Eric. *Art of Unix Programming*. New York: Addison-Wesley, 2004.
- [39] "Requested articles." *Wikipedia*. 2005. University of California, Berkeley. 16 Jun 2005 <http://en.wikipedia.org/wiki/Wikipedia:Requested_articles>.
- [40] "RPM Package Manager - Wikipedia, the free encyclopedia." *Wikipedia*. 2005. University of California, Berkeley. 1 Jun 2005 <http://en.wikipedia.org/wiki/RPM_Package_Manager>.

- [41] "Scalable Vector Graphics (SVG)." *Worldwide Web Consortium website*. 2005. University of California, Berkeley. 15 Jun 2005 <<http://www.w3.org/Graphics/SVG/>>.
- [42] Schaller, Christian. "Personal info for Uraeus." *Advogato.org*. 2005. University of California, Berkeley. 13 Jun 2005 <<http://www.advogato.org/person/Uraeus/>>.
- [43] "Scribus :: GPL Desktop Publishing for Linux and more." *Scribus*. 2005. University of California, Berkeley. 1 Jun 2005 <<http://www.scribus.net/>>.
- [44] "Statistics.txt." *Open Clip Art Library website*. 2005. University of California, Berkeley. 15 Jun 2005 <<http://openclipart.org/clipart/statistics.txt>>.
- [45] *TheOpenCD.org*. 2005. University of California, Berkeley. 5 Jun 2005 <<http://www.theopencd.org/>>.
- [46] "Tools." *Open Clip Art Library Wiki*. 2005. University of California, Berkeley. 15 Jun 2005 <<http://openclipart.org/cgi-bin/wiki.pl?Tools>>.
- [47] "Version control system." *Wikipedia*. 2005. University of California, Berkeley. 16 Jun 2005 <http://en.wikipedia.org/wiki/Version_control_system>.
- [48] "WhoIsWho." *Open Clip Art Library Wiki*. 2005. University of California, Berkeley. 15 Jun 2005 <<http://openclipart.org/cgi-bin/wiki.pl?WhoIsWho>>.
- [49] Wiley, David. *OpenContent*. 2005. University of California, Berkeley. 15 Jun 2005 <<http://www.opencontent.org/>>.
- [50] Worth, Richard. "MetadataDiscussion." *Open Clip Art Library Wiki*. 2005. University of California, Berkeley. 15 Jun 2005 <<http://openclipart.org/cgi-bin/wiki.pl?MetadataDiscussion>>.