



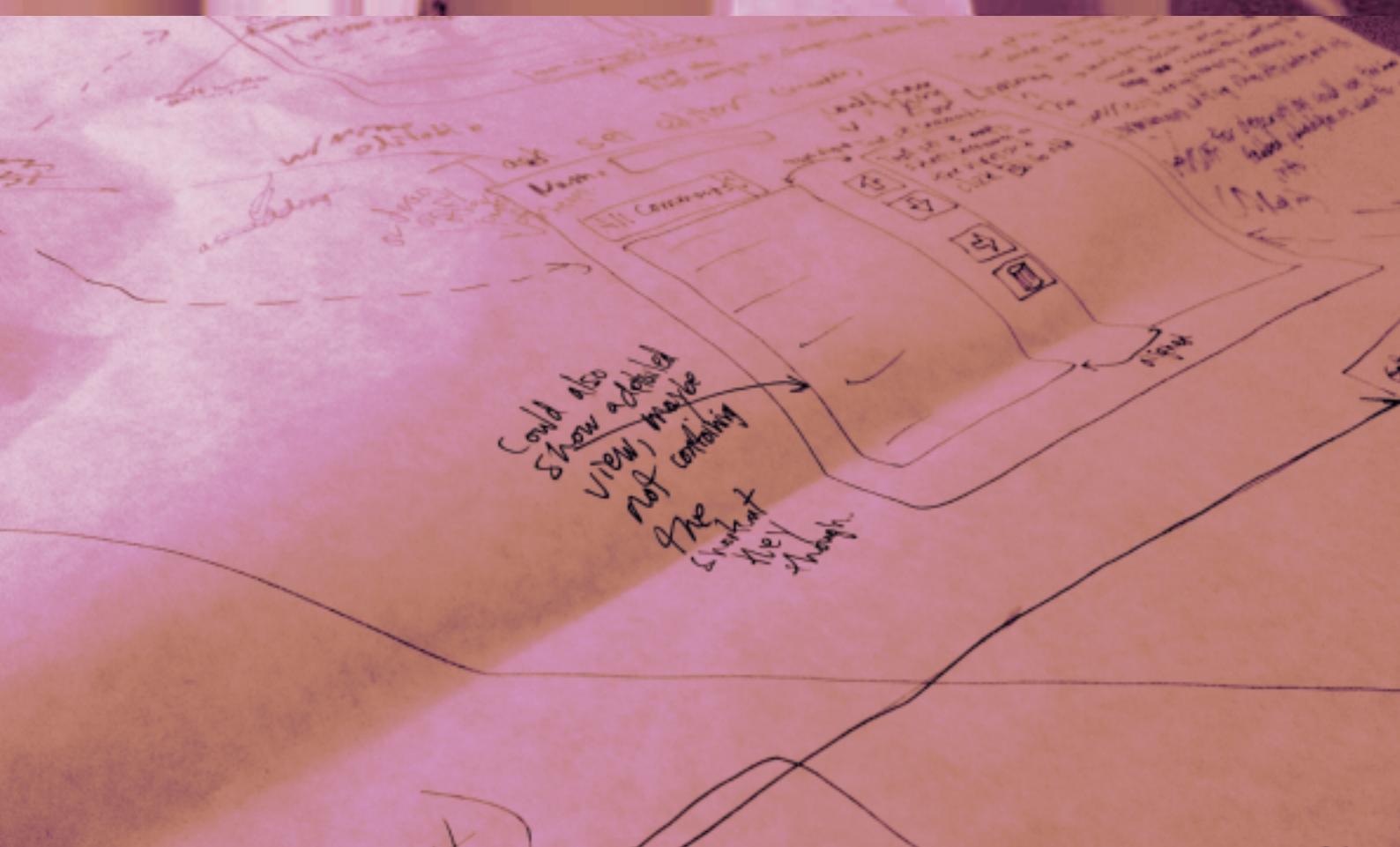
LIBRE GRAPHICS MEETING 2011

MONTRÉAL

10-13 MAY

JOIN
SPEAK
CONTRIBUTE

[HTTP://LIBREGRAPHICSMEETING.ORG/2011](http://libregraphicsmeeting.org/2011)



AdaptableGIMP: user interfaces for users

ginger coons

In 2007, Michael Terry and other members of the University of Waterloo HCI lab set out to learn just what GIMP users actually do. To achieve that lofty goal, they created something called `ingimp`, a variation of GIMP which tracked feature use. Four years later, they have an answer, in a way.

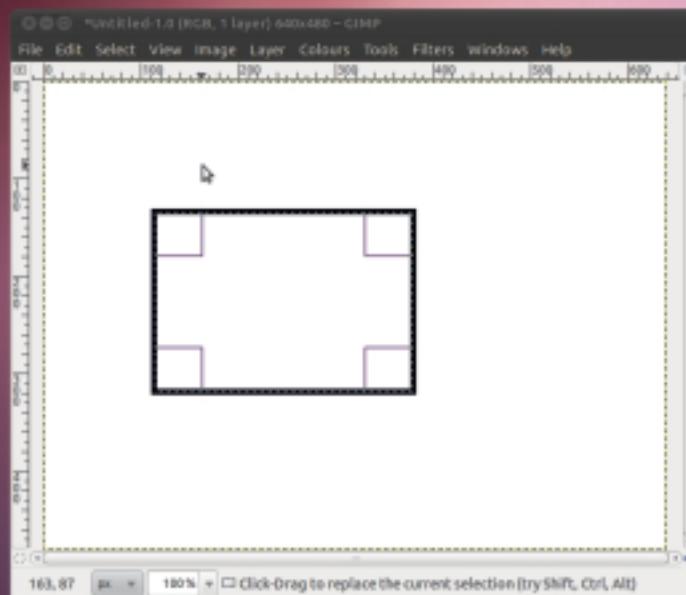
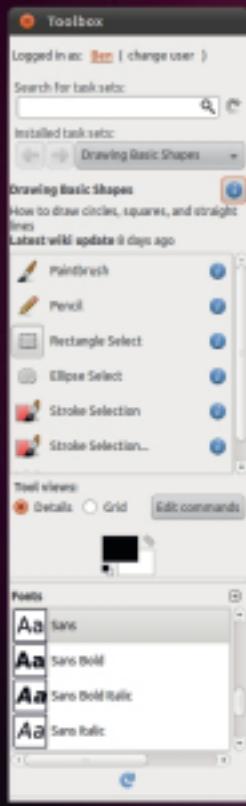
The answer, broadly, is what you might expect. It turns out that different users of GIMP do different things. Ben Lafreniere, a doctoral candidate in Terry's HCI lab, has combed through the data and come up with a more nuanced answer. Usage tends to be focused on small sets of tools, using only a tiny percentage of the actual capabilities of the program. The members of the lab refer to these groups as "corners."

According to Lafreniere, "not only do people use just a small corner of the functionality in the system, but they tend to use fairly distinct corners." Which means that there's no one-size-fits-all answer. With different users making use of small, distinct sets of tools, no one easy interface tweak will suit everyone and make GIMP universally more usable.

But never fear. There's another, far more exciting option. That option comes in the form of AdaptableGIMP. The premise of AdaptableGIMP, another project from the HCI lab, is that not only should users be able to customise the interface of their software, they should be able to share those customisations with others. Or, as Lafreniere puts it, crowdsourcing "the creation of customisations to the entire user community."

To do this, AdaptableGIMP relies on a modified version of MediaWiki. Task sets—customised collections of GIMP commands tailored to a specific use—are stored in a central repository, tied to wiki pages which are capable of both describing and controlling the mix of features in each set.

"It's like an infinite set of overlapping Microsoft ribbons. They try to do the same thing, they're trying to group functionality. But we're saying that it doesn't need to be the six that are defined by the people making the application, there can be a million. You can't only have the paintbrush in one. The paintbrush can be in 500,000 of them."—Filip Krynicki



[Drawing Basic Shapes](#)

TaskSet:Drawing Basic Shapes

How to draw circles, squares, and straight lines

Contents [hide]

- 1 Drawing Straight Lines
- 2 Drawing Rectangles
 - 2.1 Notes
- 3 Drawing Circles and Ellipses
 - 3.1 Pkrynic said ...

Drawing Straight Lines

- = Select the Paintbrush tool
- = Left click once on the canvas to create a dot at the starting point of your line
- = Hold down the Shift key on your keyboard. GIMP will display a preview line between the dot and the current mouse position
- = While holding Shift, click your desired end point on the canvas.

Drawing Rectangles

- = Select the Rectangle Select tool
- = Draw a rectangular selection on the canvas by clicking with the left mouse button, dragging, and releasing the mouse button
- = Click Stroke Selection
- = Click None to clear the selection

Notes

The beauty of this, according to Lefreniere, is that “when anybody creates a customisation to the interface, it’s immediately there, available to all the users of the application.” This provides all users with a collection of available task sets, just waiting to be used. Says Lafreniere, the intention is that a user “can sit down at the interface, type a few keywords describing what they want, searching things made by the community, select one, and then immediately have it.”

And who will build those task sets? According to Terry, there's already tangible evidence that some users are more than willing to create documentation, tutorials and other resources. “What we're doing,” he says, “is bringing that practice more directly into the interface.”

This community approach to building and documenting task sets has an added benefit: it makes the efforts of one person useful and valuable to all other users of the software. This means that different types of users can work to their own strengths and preferences, while benefiting from the preferences of others.

"What we're grafting onto the existing interface paradigm, is this task-centric view of computing where you say 'This is what I want to do' and the interface modifies itself to accomodate that particular task."—Michael Terry

The screenshot shows a Mozilla Firefox browser window displaying a GIMP task set page. The title bar reads "TaskSet:Drawing Basic Shapes - Adaptable Gimp Wiki-Mozilla Firefox". The page content is organized into several sections:

- Navigation:** Includes links to Main page, Community portal, Recent changes, Random page, and Help.
- Toolbox:** Includes links to What links here, Related changes, Special pages, Printable version, and Permanent link.
- Contents:** A sidebar with a tree view showing the structure of the task set:
 - 1 Drawing Straight Lines
 - 2 Drawing Rectangles
 - 2.1 Notes
 - 3 Drawing Circles and Ellipses
 - 3.1 Frkrynic said ...
- Commands:** A sidebar listing available tools:
 - Paintbrush
 - Pencil
 - Rectangle Select
 - Ellipse Select
 - Stroke Selection
 - Stroke Selection...
 - None
- Drawing Straight Lines:** Instructions for creating straight lines using the Paintbrush tool.
- Drawing Rectangles:** Instructions for creating rectangles using the Rectangle Select tool.
- Notes:** Additional notes about stroke selection.
- Drawing Circles and Ellipses:** Instructions for creating circles and ellipses using the Ellipse Select tool.
- Comments:** A section where "Frkrynic said ..." is mentioned.

"People are hesitant to stop the current task that they're working on to create a customisation" says Lafreniere. To Filip Krynicki, one of the HCI Lab's co-op students, this is one of the major benefits of the AdaptableGIMP approach. According to Krynicki, "in most interfaces where someone can make a customisation, that's where it stops." But in the case of AdaptableGIMP, if even one percent of users actually create customisations, all users benefit.

Users creating customisations may see some added incentive, too. Terry suggests that, given AdaptableGIMP's ability to collect usage data, task sets could well come along with information about how many users they've been installed by, how active their development is and even how recently they've been used. To Terry, this gives creators of task sets "some sense of feedback of the utility of the task set."

A NEW APPROACH TO INTERFACE DESIGN

Members of the HCI Lab see current interface design as something hierarchical and designed more to contain functionality than to help users accomplish their tasks.

According to Terry, one of the goals of AdaptableGIMP is to help users define their own workflows. This approach contrasts strongly with hierarchical interfaces, which he says are "designed in reaction to the large number of commands that are available and not designed around how people actually sit down and want to use the tool for a particular task."

This does not mean changing the entire functioning of the program or reinventing the wheel. To Terry, it's a case of "grafting onto the existing interface paradigm," adding in a "task-centric view of computing where you say 'This is what I want to do' and the interface modifies itself to accommodate that particular task."

Krynicki puts it into contrast with existing tactics: "It's like an infinite set of overlapping Microsoft ribbons. They try to do the same thing, they're trying to group functionality. But we're saying that it doesn't need to be the six that are defined by the people making the application, there can be a million. You can't only have the paintbrush in one. The paintbrush can be in 500,000 of them."



The future of AdaptableGIMP looks, at very least, exciting. Lafreniere suggests the possibilities presented by a built-in recommendation system, offering complementary task sets based on use patterns or even suggesting task sets which frame commands the user already knows, but to different ends. As Lafreniere puts it, “you know all these commands, you could do this task.”

Of course, it's not just GIMP standing to benefit from this work. Terry hopes to offer a core set of AdaptableGIMP components which would help developers of other software in implementing crowdsourced customisation themselves. Says Terry, “we hope that we can provide a tool set for them that they can plug in and start to use in their own application.”

AdaptableGIMP is available now, for users who don't mind compiling from source. Get it at [HTTP://ADAPTABLEGIMP.ORG](http://ADAPTABLEGIMP.ORG).

Resource list 1.2

BLENDER

A powerful r/LOSS 3D animation application for GNU/Linux, Mac os x and Microsoft Windows.



GIMP

A raster based image editor for GNU/Linux, Mac os x and Microsoft Windows.



IMAGEMAGICK

A raster image editing, creation and conversion suite for GNU/Linux, Mac os x, Microsoft Windows and iPhone, among others.

```
# Create a montage from a folder containing various png images
montage -geometry 400x300+0+0 *.png icon-montage.png

# Scale all jpeg images in a folder to a width of 640px
for img in *.jpg ; do convert $img -scale 640 $img; done;

# Rotate a batch of jpeg images 90° and convert them to png
for img in *.jpg ; do convert $img -rotate 90 ${img/jpg/png} ; done
```

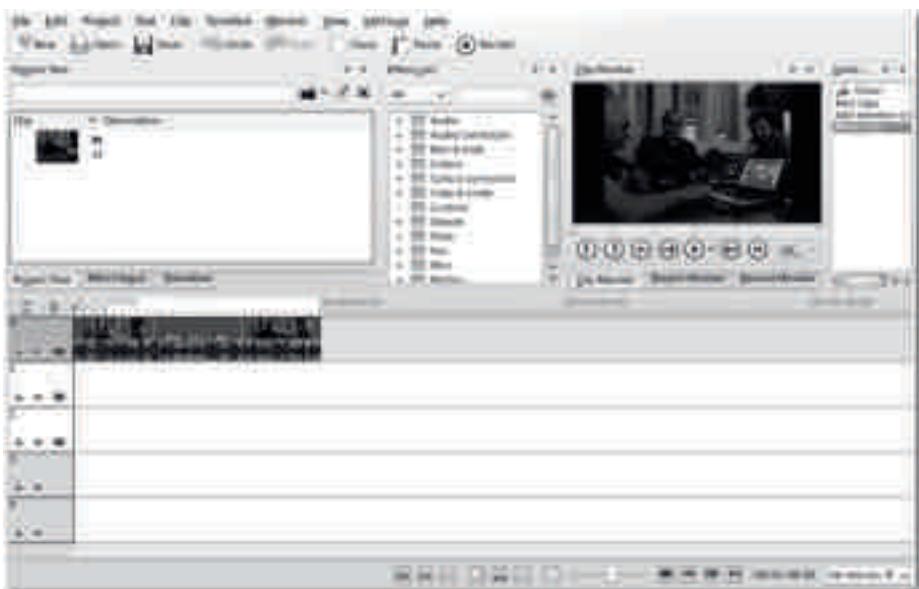
INKSCAPE

A vector graphics editor for GNU/Linux, Mac os x and Microsoft Windows.



KDENLIVE

A video editor for GNU/Linux, Mac os x, Microsoft Windows and FreeBSD.



MYPAINt

Graphics application focused on natural media simulation. Available for GNU/Linux, Mac os x and Microsoft Windows.



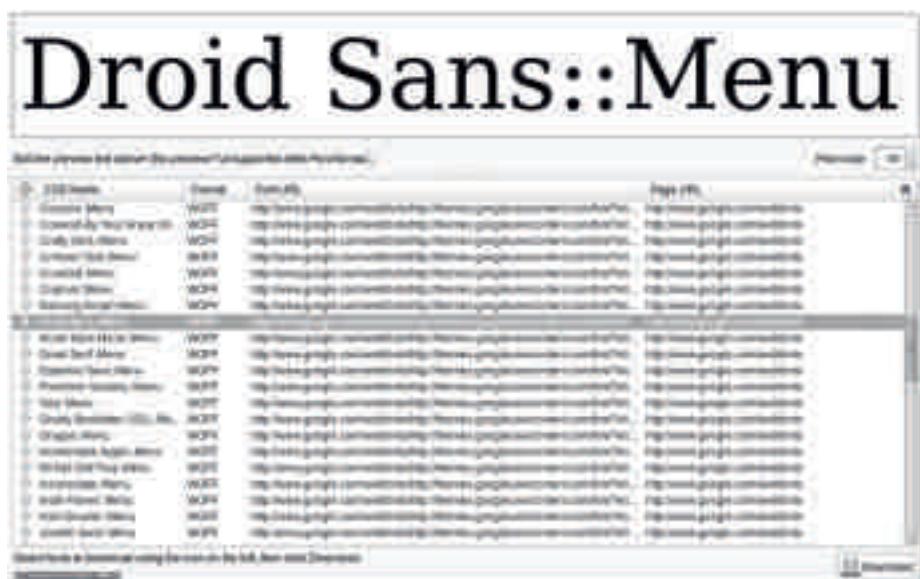
SCRIBUS

A desktop publishing program for GNU/Linux, Mac os x and Microsoft Windows.



WEB FONT DOWNLOADER

An extension for Firefox, allowing downloads of embedded web fonts.



Glossary 1.2

Alchemy:

A F/LOSS canvas drawing program, meant to encourage play and exploration. Available for GNU/Linux, Mac os x and Windows.

Audacity:

A F/LOSS sound editing application for GNU/Linux, Mac os x and Microsoft Windows.

Blender:

A powerful 3D animation application for GNU/Linux, Mac os x and Microsoft Windows.

command line:

A text-based interface for controlling a computer.

desktop environment:

A collection of tools and interface elements which style the visual and functional aspects of an operating system in a certain way.

Digital Rights

Management (DRM):

Technologies (of whatever sort) which prevent users from making certain uses of the larger technologies to which the DMR is applied.

distro/distribution:

A specific configuration of GNU/Linux, often designed with a particular purpose in mind.

Fedora:

A popular distribution of GNU/Linux, produced by Red Hat, Inc.

flavour:

Similar in meaning to distro/distribution, but more general. Simply means a specific version (normally of GNU/Linux).

Free:

As in freedom, or often, that which is or is of Free Software.

Free Culture:

A general term for activities and artistic works which fall under a similar ideological banner to the Free Software movement.

freedesktop.org:

A F/LOSS project which focuses on creating interoperable tools for GNU/Linux and other Unix-type systems.

Free/Libre Open Source Software (F/LOSS):

Software which has a viewable, modifiable source and a permissive license (such as the GNU GPL). It can be modified and redistributed.

GIMP:

A raster based image editor for GNU/Linux, Mac os x and Microsoft Windows.

Git:

A popular version control system, originally created to manage development of the Linux kernel.

GNOME:

A popular desktop environment for GNU/Linux.

GNU General Public License (GPL):

A license originally intended for use with software, but now used for other applications. Made famous the principle of Copyleft, requiring those using GPL licensed work to license derivatives similarly.

implement:

The act of integrating a feature or standard into a piece of software, rendering that software able to (for example) perform a task or use a specific file format.

Internet Relay Chat (IRC):

A popular form of internet-based real-time chat. Has a long history of use and is still popular among groups of developers and users.

KDE:

A community project which produces various FLOSS applications, best known as a popular desktop environment for GNU/Linux.

Libre:

A less ambiguous adaptation of the word Free. Implies liberty of use, modification and distribution.

mailing list:

An email-based forum through which subscribers may receive announcements, view or participate in discussion.

open standards:

A standard which is available for viewing and implementation by any party, often at no monetary cost.

Oxygen:

A project meant to develop a coherent and attractive visual identity for KDE.

proprietary:

A piece of software or other work which does not make available its source, which is not allowed or intended to be modified or redistributed without permission.

Scalable Vector Graphics (SVG):

An open standard for vector graphics, developed by the W3C.

SIL Open Font License (OFL):

A license intended for use with fonts and font related software. Dictates terms which allow modification and redistribution of fonts.

source code:

The human readable code on which software is based. Software distributed with its source code can be modified far more easily than software distributed without.

terminal:

A program which allows users to perform actions on the command line.

Ubuntu:

A particularly popular distribution of GNU/Linux, produced by Canonical Ltd.

version control:

Activities which have the effect or intent of distinguishing different versions of a work or body of work from one another.

Version Control System (VCS):

An application/collection of tools designed to facilitate version control. Tracks changes to files and allows a group of collaborators to share their changes as they are made.

W3C:

The organization responsible for setting web standards, such as HTML5 and SVG.



