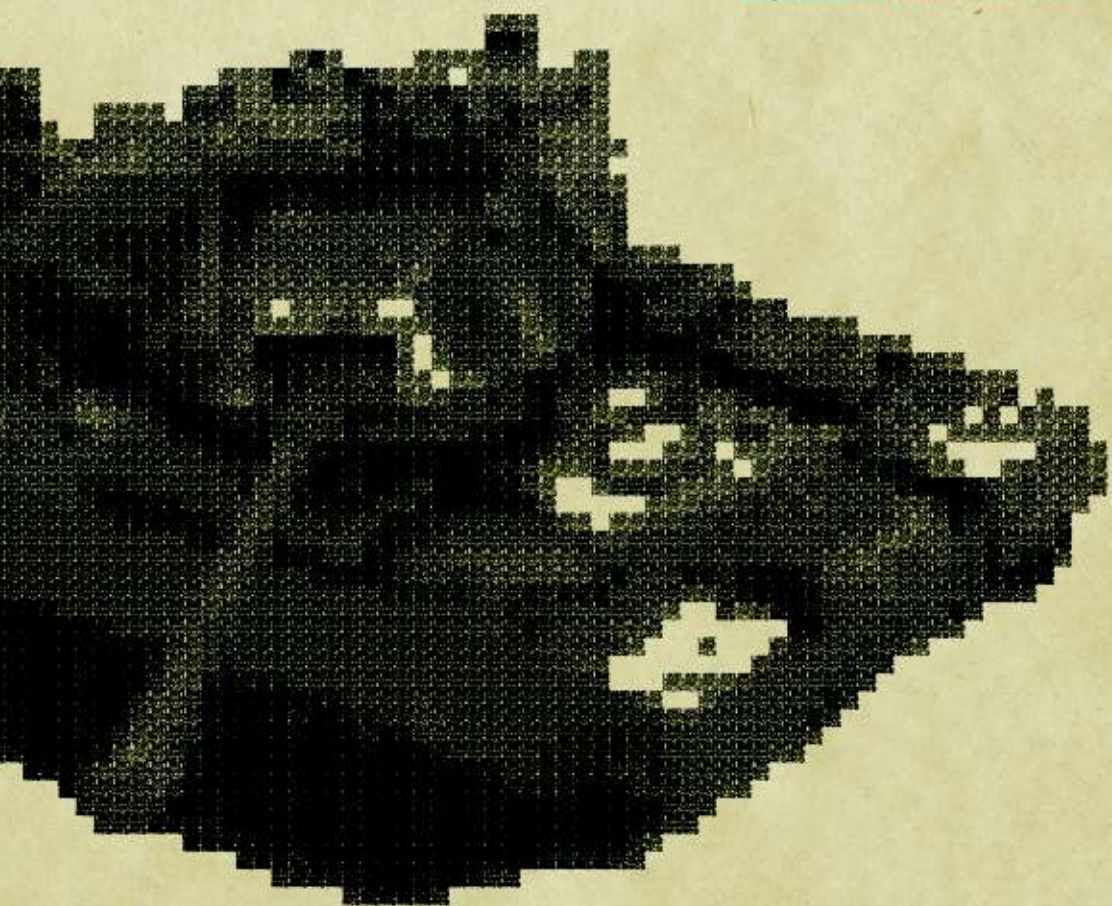


**LIBRE
GRAP
HICS
#0!**



This publication was typeset using the Junction, Titillium and Liberation Sans Mono typefaces.

Completed 25 May, 2010 using Scribus, Inkscape, GIMP and Shoebot.

Put together by ginger coons, Ana Carvalho, Ricardo Lafuente, Femke Snelting and Alessandro Rimoldi.

Printed in an edition of 1000 by Drukkerij Bulckens nv, Grensstraat 9, Herenthout, Belgium.

All works in this magazine are the property of their creators and are governed by the licenses chosen by those creators. Please go to the source if you wish to reuse the contents of this publication.

Special thanks to those who helped this project along, who are too innumerable to name, but know who they are.

VLAAMSE
GEMEENS
CHAPSCO
MMISSIE



With the support of
the Flemish authorities



```
#!/usr/bin/shot
```

```
# Here you can set your own parameters for this script.
```

```
SVG_FILENAME = 'cover-elements-parse.svg'  
IMAGE = 'cover-tank.png'  
ROWSIZE = 10*mm  
COLSIZE = 10*mm  
NUMBER_OF_SHADES = 7  
sequence = 'a'
```

```
size(900*mm, 700*mm)
```

```
fill(.1)
```

```
# background(9)
```

```
# Here be dragons!
```

```
# Lib importing hack -- move this onto main Shoebot code
```

```
import sys
```

```
sys.path.append('.')
```

```
svg = ximport('svg')
```

```
particleset = {}
```

```
def get_particle(name, value):
```

```
    global particleset
```

```
    particles = particleset[name]
```

```
    path = particles[value]
```

```
    return path
```

```
data = open(SVG_FILENAME).read()
```

```
paths = svg.parse(data)
```

```
# generate our particle dicts
```

```
for path in paths:
```

```
    # drawpath(path)
```

```
    name = path.id[0]
```

```
    value = int(path.id[1:].strip('-')) - 1
```

```
    if particleset.get(name):
```

```
        particles = particleset[name]
```

```
        particles[value] = path
```

```
    else:
```

```
        particleset[name] = {value: path}
```

```
# load an image
```

```
from PIL import Image
```

```
im = Image.open(IMAGE)
```

```
pix = im.load()
```

```
counter = 0
```

```
imagewidth, imageheight = im.size
```

```
for x,y in grid(imagewidth, imageheight, 1, 1):
```

```
    if counter > len(sequence) - 1:
```

```
        counter = 0
```

```
    name = sequence[counter]
```

```
    counter += 1
```

```
    value = NUMBER_OF_SHADES - 1 - pix[x,y]
```

```
    print value
```

```
    if value == 0:
```

```
        # white
```

```
        continue
```

```
    translate(x*ROWSIZE, y*COLSIZE)
```

```
    path = get_particle(name, value)
```

```
    drawpath(path)
```

```
    reset()
```

Time to Show and Tell

Neal Stephenson likens operating systems to cars. In his analogy, Windows is a station wagon and Mac OS is an expensive, attractive European-style car. The two are available in dealerships, along with all the normal service options. Linux, on the other hand, is a tank. Not only a tank, but a free one. It's a stronger, faster, more reliable vehicle with a personal approach to maintenance. But it doesn't have a dealership or ad budget.

Libre Graphics is in a similar situation. It's strong, fast, reliable and even diverse. It has great community support and investment. And like Stephenson's tanks, it's being cranked out and offered to anyone who will take it.

Both the Libre Graphics Meeting and this magazine exist to serve the Libre Graphics community. LGM, now in its fifth year, has been a venue for developers to meet, organize and work. In this magazine, we present to you the output of that work. Libre Graphics #0 showcases the work of developers, users, artists and people with any number of other titles. Some do performance art, some make films. In common, they have Libre Graphics.

The quick brown fox jumps over the lazy dog

The quick brown fox jumps over the lazy dog

The quick brown fox jumps over the lazy dog

The quick brown fox jumps over the lazy dog

The quick brown fox jumps over the lazy dog

The quick brown fox jumps over the lazy dog

ipsum dolor sit amet, consectetur adipiscing elit

ipsum ad minim veniam, quis nostrud exercitation

ullamco laboris nisi ut aliquip ex ea commodo

consequat. Duis aute irure dolor in reprehenderit

in voluptate velit esse cillum dolore eu fugiat

nulla pariatur. Excepteur sint occaecat cupidatat

no proident, sunt in culpa qui officia deserunt

Why make new fonts?

Dave Crossland (www.understandingfonts.com)

"Why make new fonts?" is the most common question I have been asked since I set out to become a typeface designer. When I mention I did a Masters degree in the subject, at the University of Reading, England, I sometimes meet genuine surprise that this subject is studied seriously. Often, people haven't ever thought about where fonts come from, since the fonts on their computer are just there, you know?

We see different fonts out in the world constantly, so we all know there sure are a lot of them. That's why the reasons for needing more aren't obvious. Perhaps the surprise also stems from the way fonts are subliminal. We focus on the meaning of what we read, and that directs our perception away from the underlying mechanics of reading. Common everyday things can be quite mysterious when you come to examine them for the first time, and reading seems to be one of them.

New fonts are made for many reasons, and often several are involved at the same time. Here are three:

Identity

Organisations need visual identities, and fonts can strongly identify who published what. To have a distinctive visual identity is to be fashionable. Fashion makes life more interesting, and is as much a part of typography as any other aspect of culture. While you can achieve a very distinctive visual identity with a very common font - the popularity of Helvetica is testament to that - I suggest that this is the prime motivation graphic designers desire new fonts today; it's why typographers license new retail fonts and why corporate branding projects commission bespoke fonts. Such commissions can fail to be enough justification for more than a single font, not a whole family - just extending a logotype into a full alphabet. Sometimes it can result in big families though, and this year Canonical commissioned a very big family of fonts from the Dalton-Maag foundry in London.

Art

While type designers cater to the desires of their customers, they also have minds of their own and those minds usually have an artistic bent. There's that basic urge for self expression, expressing emotions through letter forms as well as the whole George Mallory thing: "Because it's there."

The power of "I made this!" shouldn't be underestimated. It underpins why I became a type designer. Victor Gaultney, the designer of the libre font family Gentium, is also a trained musician and he said simply, "Why make new fonts? Why make new music?"

There's a business side to type design as artistic practice, too. I've heard seasoned professionals like Matthew Carter and Gerard Unger mention that during their careers they were constantly working on fonts for themselves, privately. When a graphic designer incubates a little secret stash like this, when a commission comes along and these private designs are relevant, they can be refined to meet the brief - boosting turnaround time and the bottom line.

Whatever kind of design you are doing, if you try to design without any constraints and only from self expression, you're not really engaged in design, you're engaged in Art. The initial motivation might well be indistinguishable

from the needs of users of the final design artefact, but I think it is a qualitatively different kind of process. What really anchors type design as a design process is this:

Design

Fonts can help or hinder the legibility of text. Raph Levien made Inconsolata for typesetting program code, a real niche. Typefaces for code are typically designed for reading on screens rather than in print since that's really where code lives. This typographic context directs the design in several ways, such as to make the brackets unconventionally big because they have a key role in code. Those brackets are not appropriate for, say, a telephone directory.

When Matthew Carter designed Bell Centennial for AT&T in the 1970s, he made a new typeface for telephone directories that would be used at very small sizes, which would get more text per line, and yet would remain just as legible as the old one. The typographic context is different, and the brackets' designs are different. This is subtle stuff, subliminal even to many graphic designers, but this is the stuff that really pays for type design - because it is what makes excellent typography, and it pays for itself many times over. When you're printing 50,000 copies, getting 10% more text into 50 fewer pages than the last edition saves a lot of money.

Freedom

And this illustrates why libre fonts are so valuable. If you're using a font and it doesn't quite work as you wish - if it takes up too much space, or if it doesn't feel right, or it could be made a bit more cool and contemporary - can you change it? If you're using a proprietary font, you can't. Well, you can always make a new typeface that has similar qualities, from scratch. But with a libre font, you can take it in new directions that the original developers never would have thought of straight away.

Why LGM matters

Alexandre Prokoudine (<http://prokoudine.info>)

The Libre Graphics Meeting is an annual conference for free graphics software developers and users from around the world. Having started as a meeting of GIMP developers back in the early 2000s, in 2006 it grew into an appealing event for people who produce design, typography, photography and 3D art and animations. The highlights of the conference are collaboration between software projects, discussion of innovations and usability, displays of photography and art, as well as development of standards.

Birds of Feather sessions are one of the most interesting features of LGM. At BOF sessions, developers meet and discuss ways for their respective projects to converge, like file format compatibility, common approach to user interfaces and reuse of code via shared libraries. The input from users is always most valuable.

Two of the most notable achievements to come out of LGM so far are the LensFun library, which is used by several photography related applications for automatic/nearly-automatic fixing of various lens distortions; and UniConvertor – a universal vector documents converter that is used by Inkscape and Scribus for opening a number of vector file formats including Corel DRAW files.

The question of complex script support in free software has always also been quite important. There are a number of advanced technologies now that aim to make free software accessible for minorities all around the globe. Achieving that goal would be rather difficult if it wasn't for collaboration between major projects. LGM has been the place to discuss complex script

support from day one – starting with talks on XeTeX back in 2006 and continuing with hosting annual Text Layout Summits where type designers and developers meet to discuss existing issues and work out solutions.

LGM is also a great place to present new interesting technologies. In the past there have been great presentations covering Panini, a rediscovered panorama projection based on works by the Veduta artists of the 17th century, and Lighttwist, a system to create 360 degree panoramic installations via immersive displays.

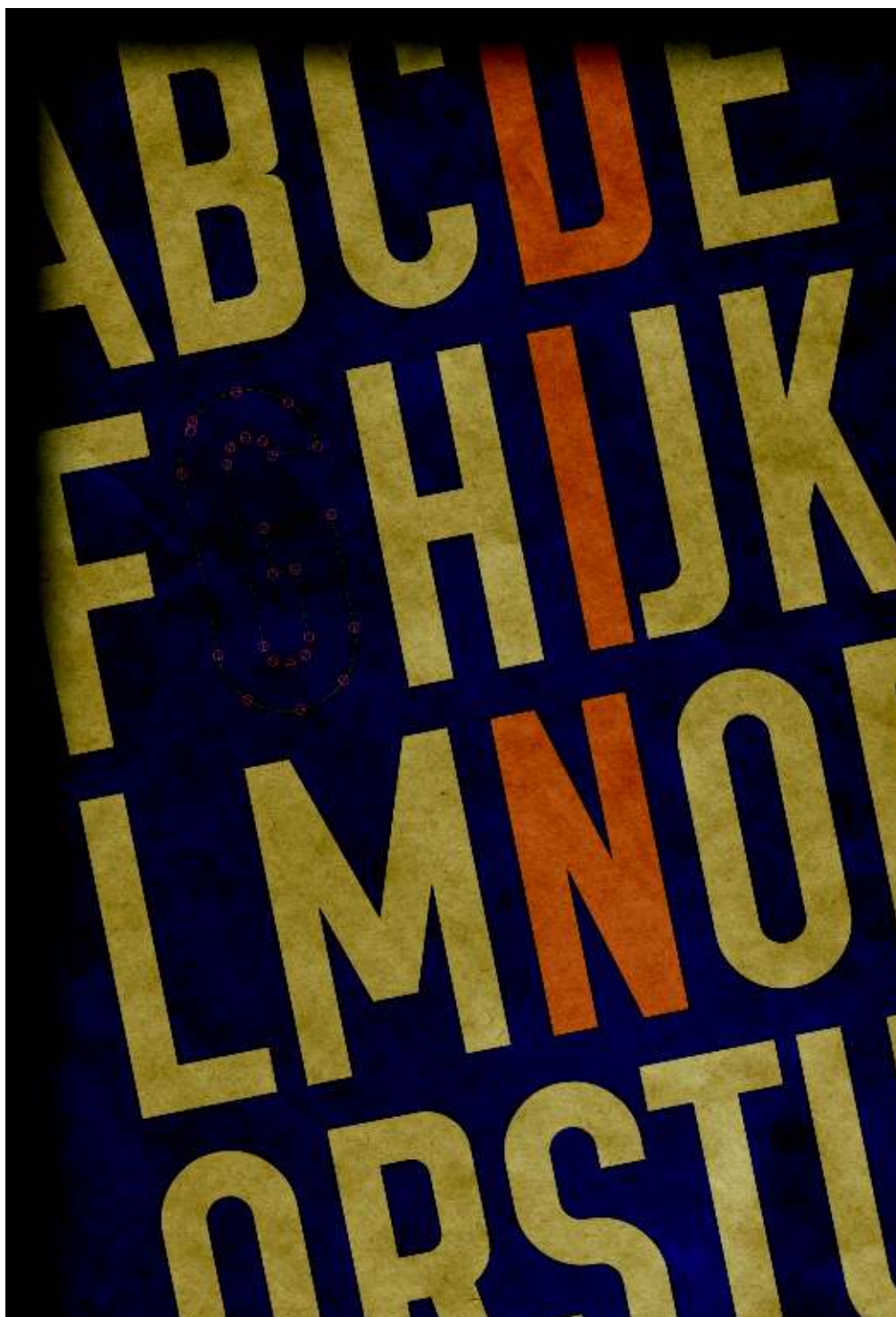
Another example is work by Krita developers on various digital painting technologies. Krita is still the only application to implement Kubelka-Munk color space for natural mixing of colours. The team is also working at implementation of a technology called Impasto – a really mind-blowing way to do realistic digital painting.

When it comes to usability studies, the first research project to name here is ingimp, or Instrumented GIMP. It's an academic study of adaptable user interfaces based on solid statistical research. The team discovered that on average, only 11 commands out of hundreds available are typically employed by GIMP users. It means that typically, a user of a such a complex tool has to go through a lot of functions to reach the ones he really needs. The proposed solution is to build editable clusters of commands used for particular tasks. The project was first presented at LGM in 2007 and subsequently updated in 2008 and 2009 as it progressed.

The mentioning of usability at LGM could never be justified without mentioning Peter Sikking's talks on making GIMP more usable. Thanks to his continuous work over the last few years, GIMP developers have managed to provide a much better user experience.

LGM is also big for standards. Apart from talks by representatives from W3C working groups, LGM has its own achievements. The OpenRaster initiative that was started in 2006 is now bearing fruits. With the help of OpenRaster, major open source players in the arena – Krita, GIMP and MyPaint – can open and save ORA files, which makes exchanging multilayered project files between them possible. Even more, MyPaint uses subset of ORA as native file format.

The Libre Graphics Meeting, in all of its varied interests and groups, has been central to many exciting developments in past. It only remains to be imagined what will come out of future meetings.



OSP DIN

Open Source Publishing (<http://ospublish.constantvzw.org>)

To get our hands into many interesting, but difficult questions, the Open Source Publishing team created a new digital rendering of the classic DIN 1451 font, released into the public domain. Our version is called OSP DIN.

The history of DIN 1451 brings up many questions about standards, their political implications and relations to use. In 1936 the German standards committee, the Deutsches Institut für Normung, decided DIN 1451 should be employed in technology, traffic, administration and business, with the idea to facilitate the development of German engineering and industry. Our point of departure is therefore far from neutral ground.

The starting point of the project was that we wanted to design an open format DIN 1451 font, based on the original documents stored in the DIN archives.

Our exploration took us to Berlin. While encountering books, people, and wandering in our minds away from the core problematic of how to design such a font, more general (and richer) questions arose: the idea of implementing a “standard” as well as public purpose fonts.

From the different versions we saw in the archives, it seems that the main DIN letter models are based on solo strokes drawn on a grid at small size. The thickness of the tool used (pencil, drawing pen, ball nose mill) defined the boldness of the strokes and the round or less round shape of their extremities, like flesh on bones.

Later this was also applied to larger lettering, so strokes became surfaces and the drawing began to be defined by the contour, by its skin. Simple geometric extrapolation from strokes were operated, using the unit of the grid as unit for the thickness of the stroke, to normalize sizes. In the oblique letters, the angle of the shape at the end of strokes became angled and went farther than the regular width, defined by the grid. So, as these letter parts could be less open in their fill version than in their stroke one, the core was moved a little towards the inside of the glyph to fit in the grid.

That shift from calligraphy to typography is traditionally hidden in the progressive adaptations by generations of letterers. But in the case of the DIN lettering, as a norm, the movement must be described in detail. And that effort produced the beautiful figure that appeared before us two years ago on a screen at the library of the DIN Institute.

Before long, our version of OSP DIN saw more use. For the BPI/Cinéma du Réel festival, OSP was asked to both design and “perform” the program map in full scale in the Centre Georges Pompidou main hall, where the festival occurred. The performed map is a derivative from the printed one, folded into the festival brochure. It networked the featured movies and the subjective links between them.

As the Centre Pompidou graphic chart is very restrictive, we had to use the DIN font. But it was impossible to use the FF DIN font from fontshop, as it carries a restrictive license. So we encoded the first cut of the OSP DIN, based on the drawings we got in Berlin in our previous adventures.

The re-drawing of our file “en dur” was a very strong and singular experience. We had the feeling of being drawn into the file, and re-interpreting SVG code

It was like crossing the screen. Performing the choreography of the interface is a nice journey! The installation is the translation of the file into an object, the transformation of vectors into movements.

It is a new interpretation, a new version. It's the gap between the mouseclick and the gesture by editing on another support. We felt the distance and the similarities of the interface choreography and the gesture choreography.

A new articulation...

It becomes something human again.

Collaborators on this piece were convinced by the idea and came to help. Among them, FIX, a renowned Paris graffiti artist. Graffiti crosses our problematics in many fields.

First he tried to follow the original computer file to the maximum, but the result was poor: all sensibility and spontaneity in the marking, the piece was disappearing. A paradoxical situation for a tagger. It was important that he could take the work for himself, and not be just a technician. So we simply redefined space of movement for him to translate the vectors with his feelings.

He could then confront his drawing and calligraphy experience to a new tool for him: 3M 471 model tape. And adapt his habits and talent. He found new ways to use it and we could all benefit from it.

The success of this job was to share a space of translation. And the interpretation gives a lively vibration to the general aspect of the installation.

It's an articulation.

With FLOSS, the resistance of the tool is now for us such a daily meal, that it has become a work field, an investigation space, and a playground.

OFL

THE GOALS OF THE OPEN FONT LICENSE (OFL) ARE TO STIMULATE WORLDWIDE DEVELOPMENT OF COLLABORATIVE FONT PROJECTS, TO SUPPORT THE FONT CREATION EFFORTS OF ACADEMIC AND LINGUISTIC COMMUNITIES, AND TO PROVIDE A FREE AND OPEN FRAMEWORK IN WHICH FONTS MAY BE SHARED AND IMPROVED IN PARTNERSHIP WITH OTHERS.

THE OFL ALLOWS THE LICENSED FONTS TO BE USED, STUDIED, MODIFIED AND REDISTRIBUTED FREELY AS LONG AS THEY ARE NOT SOLD BY THEMSELVES. THE FONTS, INCLUDING ANY DERIVATIVE WORKS, CAN BE BUNDLED, EMBEDDED, REDISTRIBUTED AND/OR SOLD WITH ANY SOFTWARE PROVIDED THAT ANY RESERVED NAMES ARE NOT USED BY DERIVATIVE WORKS. THE FONTS AND DERIVATIVES, HOWEVER, CANNOT BE RELEASED UNDER ANY OTHER TYPE OF LICENSE. THE REQUIREMENT FOR FONTS TO REMAIN UNDER THIS LICENSE DOES NOT APPLY TO ANY DOCUMENT CREATED USING THE FONTS OR THEIR DERIVATIVES.

OSP Talks Shop

Open Source Publishing, the F/LOSS publishing and design project based in Brussels, takes every possible opportunity to interview F/LOSS creators. Below are some choice pieces collected over the last few years. For full interviews, visit their website, ospublish.constantvzw.org.

Chris Lilley, SVG Working Group (W3C)

<http://ospublish.constantvzw.org/conversation/even-when-you-are-done-you-are-not-done>

"Engineers are much more forthcoming, because they are more interested in sharing stuff, because engineers like to share what they're doing and talk on a technical level. The worst thing is to get the managers involved and even worse is to get lawyers involved."

GRRRR, media artist

<http://ospublish.constantvzw.org/conversation/grrr-objectivity-of-the-unperfect>

"As [an] artist I try to do things differently, some disadvantages can turn out to be inputs for new ideas."

Michael Terry, HCI Professor

<http://ospublish.constantvzw.org/conversation/data-analysis-as-a-discourse>

”So, our goal is to be as unobtrusive as possible to make it really easy to get going with it, and then to just forget about it. We want to get it into the hands of as many people as possible, so that we can understand how the software is actually used in practice. There are plenty of forums where people can express their opinions about how GIMP should be designed, or what’s wrong with it. There are plenty of bug reports that have been filed, there are plenty of usability issues that have been identified, but what we really lack is some information about how people actually apply this tool on a day to day basis.”

”Instrumentation is not new. Commercial software companies and researchers have been doing instrumentation for at least ten years, probably ten to twenty years. So, the idea is not new but what is new, in terms of the research aspects of this, is how do we do this in a way where we can make all the data open? The fact that you make the data open, really impacts your decision about the type of data you collect and how you are representing it. And you need to really inform people about what the software does.”

Andreas Vox, Scribus developer

<http://ospublish.constantvzw.org/conversation/a-user-should-not-be-able-to-shoot-himself-in-the-foot>

”OSP: it is interesting how the 2.500 lines of code are really tangible when you use Scribus old-style, even without actually seeing them. When Peter Linnel was explaining how to make the application comply to the conservative standards of the printing business, he used this term ‘self-defensive code.’

A: At Scribus we have a value that a file should never break in a print shop. Any bug report we receive in this area, is treated with first priority.

OSP: We can speak from experience, that this is really true! But this robustness shifts out of sight when you use the inbuilt script function; then it is as if you come in to the software through the back-door. From self-defence to the heart of the application?

A: It is not really self-defence. Programmers and software developers sometimes use the expression: ”a user should not shoot himself in the foot.” Scribus will not protect you from ugly layout, if that would be possible at all! Although I do sometimes take deliberate decisions to try and do it. For example, for as long as I am around, I will not make an option to do automatic letter spacing because I think it is just ugly. If you do it manually, that is your responsibility; I just do not feel like making anything like that work automatically. What we have no problems with is to prevent you from making invalid output. If Scribus thinks a certain font is not OK, and it might break on one or two types of printers, this is reason enough for us to make sure this font is not used. The font is not even used partially, it is gone. That is the kind of self-defence Peter was talking about.”

George Williams, Fontforge principal developer

<http://ospublish.constantvzw.org/type/i-think-the-ideas-behind-it-are-beautiful-in-my-mind>

”OSP: And the pleasure of handling a material when you know it well. Maybe make reliable bread – meaning that it comes out always the same way, but by your connection to the material you somehow – well – it’s a pleasure to do that. So, since you’ve said that, and we then went on talking about pottery – how clay might be of the same – give the same kind of pleasure. I’ve been trying to think – how does FontForge have that? Does it have that and where would you find it or how is the...

G: I like to make things. I like to make things that, in some strange definition, are beautiful. I’m not sure how that applies to making bread, but my pots... I think I make beautiful pots. And I really like the glazing I put onto them. It’s harder to say that a font editor is beautiful. But I think the ideas behind it are beautiful in my mind and in some sense I find the user interface beautiful. I’m not sure that anyone else in the world does, because it’s what I want, but I think it’s beautiful.

And there’s a satisfaction in making something – in making something that’s beautiful. And there’s a satisfaction too (as far as the bread goes) in making something I need. I eat my own bread. That’s all the bread I eat, except for those few days when I get lazy and don’t get to make bread that day and have to put it off until the next day and have to eat something that day – but that doesn’t happen very often. So it’s just... I like making beautiful things.”



Bending free software, making a photo book

Marcus Holland-Moritz (<http://zrox.org/nzbook/>)

This is the book about my trip to New Zealand in 2009.

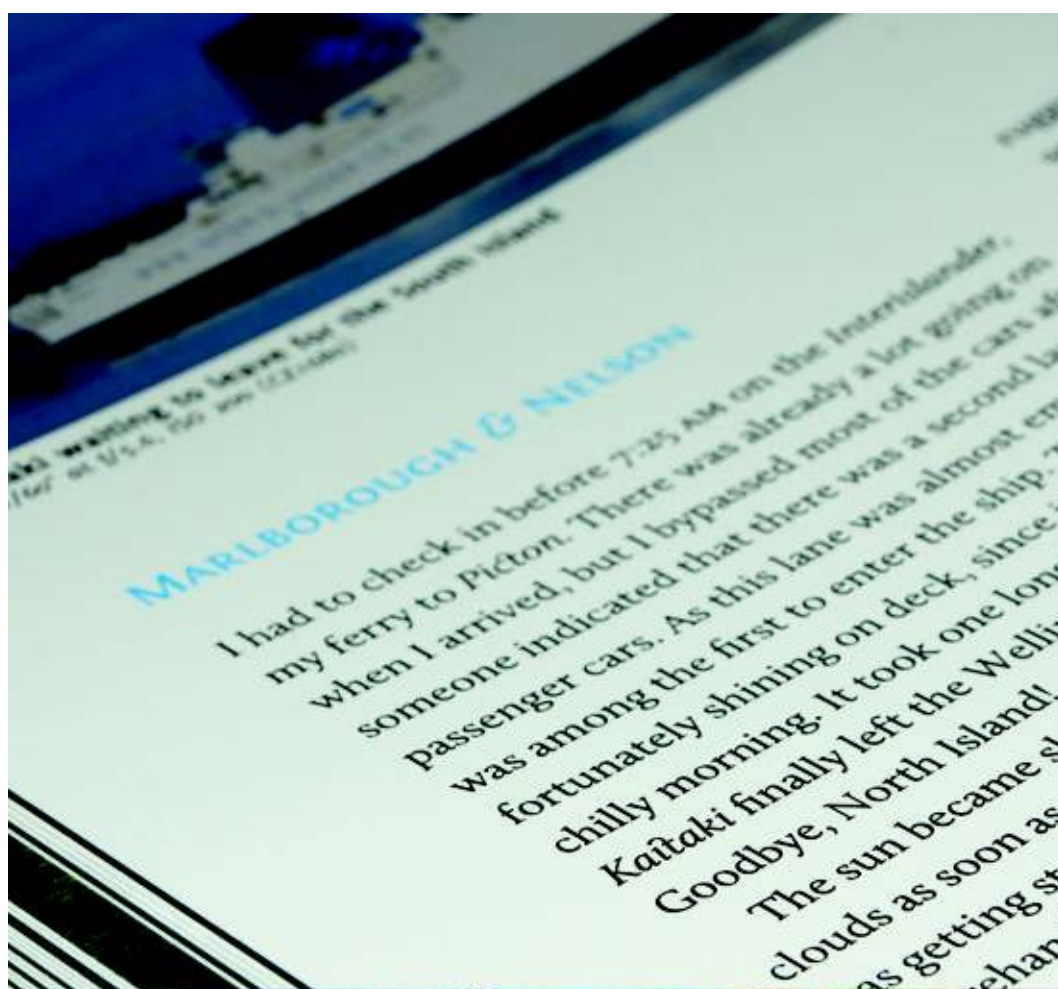
I've put several months of work into this project. It features about 200 out of the 15,000 photos I took during my five weeks "down under" along with some anecdotes about my journey across the two islands of New Zealand.

The book was created exclusively using free and open source software. It is itself available under a Creative Commons CC-BY-NC-ND license. The electronic version can be freely downloaded below in PDF format. A printed copy with high resolution photos can be ordered through blurb.



View towards York Park, Motiueka
24mm, 1/320" at 1/8, ISO 200 (CZ1588)

uku Ngutupapa – the Royal Spoonbill
m, 1/640" at 1/8, ISO 800 (M5008)



...waiting to leave for the South Island
 ...at 11.45 AM (NZ time)

MARLBOROUGH & NIHOA

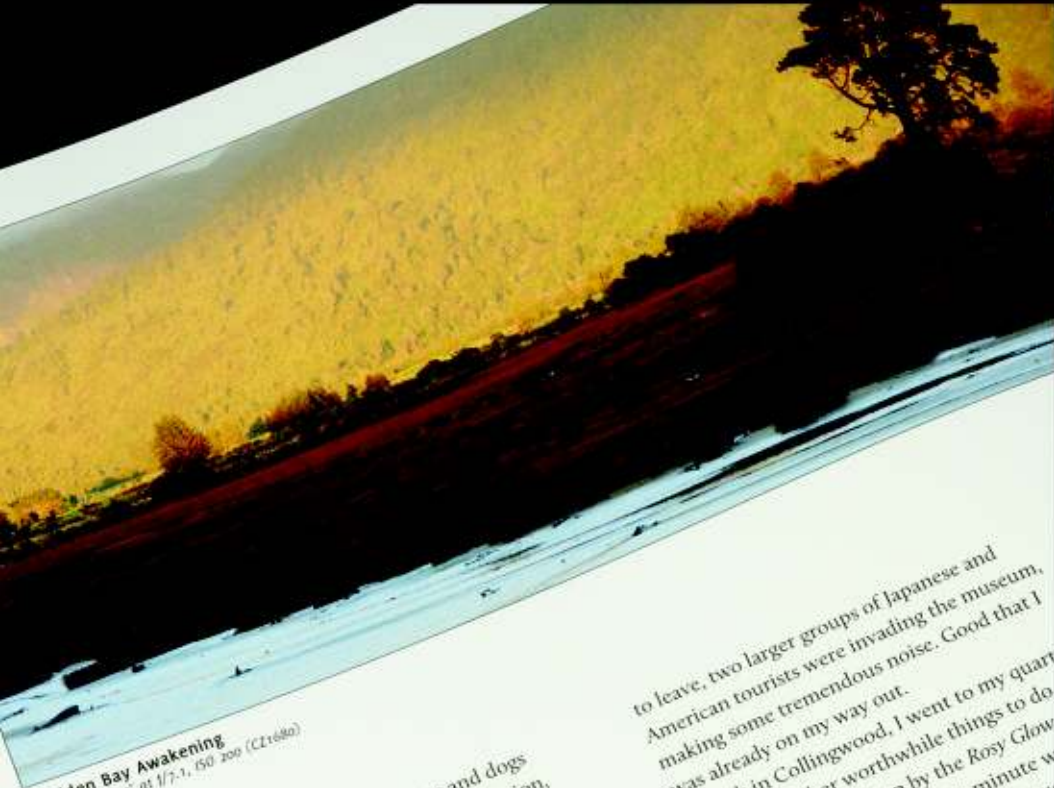
I had to check in before 7:35 AM on the Interislander, my ferry to Picton. There was already a lot going on when I arrived, but I bypassed most of the cars as someone indicated that there was a second lane for passenger cars. As this lane was almost empty, I was among the first to enter the ship. It was a chilly morning. It took one long hour to get to Kaikōi finally left the Wellington. Goodbye, North Island! The sun became visible through the clouds as soon as we were getting started. The beach and crystal clear waters of Waewaetorea Island



The beach and crystal clear waters of Waewaetorea Island
 ...at 11.45 AM (NZ time)

NORTHLAND & THE BAY OF ISLANDS

After finally saying goodbye to the big city of Auckland, I was heading north for Whangarei. To my surprise, New Zealand's first fully automated toll collection system had just been put into operation a few days ago on the SH1. Even though I was confused about how it worked, the toll collector, an artist and architect, had been living in this town for more than twenty years and the building design carries a lot of his trademark. The toll for these toilets or for the... Railway, the...

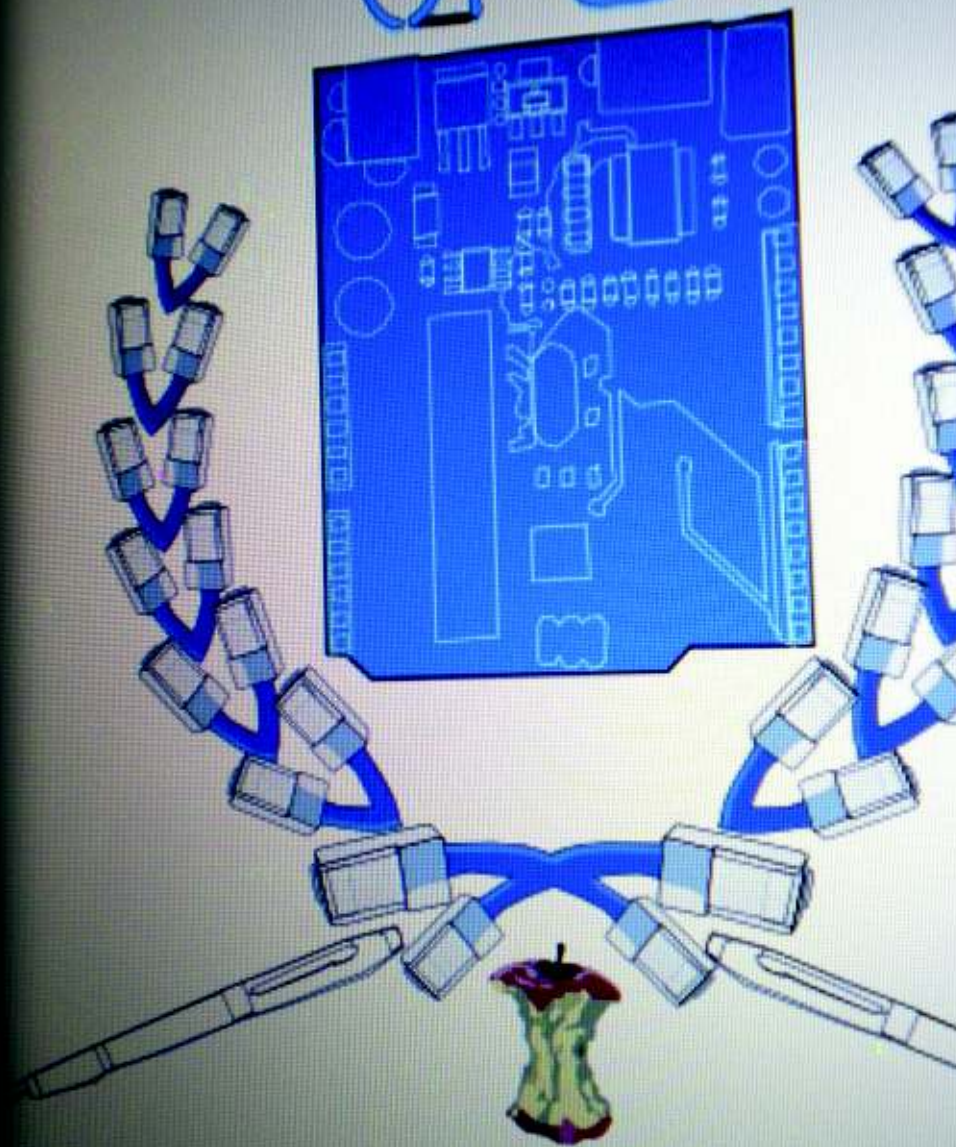


Golden Bay Awakening
120mm, 1/50 at 1/7.1, ISO 200 (C1168a)

Island. Sadly, it was still raining cats and dogs and arriving at the office of the tour operation, I was told that today's tour had been cancelled. I had plenty of time in the middle of the way to Rockville, an even smaller town where I visited the "Bartlers" Museum. As the

to leave, two larger groups of Japanese and American tourists were invading the museum, making some tremendous noise. Good that I was already on my way out.
Back in Collingwood, I went to my quarters. Asking for other worthwhile things to do, I recommended to drop by the Rosy Glow House, which was only a five minute walk where I was staying — although I guess place everything is just a five minute walk. Fortunately, I did walk there and was going sideways than a when I arrive

OPEN



Why F/LOSS, why not F/LOSS

ginger coons (<http://adaptstudio.ca>)

As the world of F/LOSS graphics grows, one notable absence remains. We're missing a critical mass of graphics professionals. Users who might benefit hugely from a F/LOSS graphics workflow and who might, in turn, benefit the community, aren't using. That's a problem for everyone.

These potential users are powerful and motivated. They spend their days parked in front of Adobe Creative Suite (among other closed source programs), making graphics happen. They are graphic designers, artists, illustrators or 3D professionals. Among their differences, they have an important common trait: they are demanding users, intimately acquainted with their software and dependent on it for their livelihoods. They live with their graphics software every day. They could be using F/LOSS, but they aren't. The whys and why-nots of their usage patterns are worth considering, whether you're a developer, a user, or one of the subjects of this discussion, a graphics professional.

What exactly prevents graphics professionals from using F/LOSS? If there were a simple answer, there would be no need for this discussion. Consider these issues: familiarity, industry standards, education, employability, limitations, optics, the loop and awareness.

The first issue is familiarity. Closed source software is often already ingrained in the heads of graphics professionals. They are willing to upgrade their skills when new versions of familiar software are released, but they do not wish to learn something entirely new. This practice is fed by industry practices.

Generally, closed source software and processes are the industry standards. Unlike in many industries, these standards are not administered by

the International Organization for Standardization (ISO), but are instead de facto standards based on their ubiquity. Designers are reluctant to use software and processes that have not been adopted by their peers and the other parties in their production chain. This connects directly to the education issue. Because closed source software and processes have become the industry standards, they are what is taught in design schools and what designers take into their professional lives. These then become market forces promoting particular applications and practices.

Of course, it goes deeper. Much eventually comes down to employability and being able to earn a living. Because graphics industries use certain programs and workflows, savvy employers look to hire employees whose skill sets match those standards. This can result in graphics professionals optimizing their skill sets for employability.

F/LOSS graphics has traditionally been plagued by limitations. The constant litany of wary professionals has been that the software doesn't do what they want and expect. Happily, this situation is changing. A common complaint in print design, for example, has been the lack of CMYK support. This problem, at least, is slowly disappearing, with many programs supporting CMYK. Some of these methods are simpler than others. And this is only one of the many issues raised about usability and necessary functions. If graphics professionals believe that F/LOSS options lack basic functionality, they will not use them and thus not become the catalyst for professional viability.

Beyond functionality, there are also optics problems. Users who are set in their ways can be short-sighted. Someone trained on Creative Suite might look at F/LOSS alternatives as clones and copycats. The GIMP may be seen as a cheap alternative to Photoshop, Inkscape as a knock off of Illustrator. Because these F/LOSS options are often seen as poor imitations for amateurs, they face low adoption.

Given all of the above, it's not surprising that a feedback loop exists. Graphics professionals learn proprietary software in school because the industry runs on it. Employers demand knowledge of specific software because it is the norm. Connected industries, like printing, run on those de facto standards because of their clients. This feedback loop cements the place of proprietary software in the graphics industry.

Aside from everything else, there's also the simple problem of awareness. Some graphics professionals may simply not know that F/LOSS alternatives exist. Without knowledge that an alternative exists, they have no hope of trying and liking those alternatives.

Now we have a hypothesis about why graphics professionals aren't already using F/LOSS. But why should they change their habits? How might they benefit from such change? Consider:

- F/LOSS means open standards. Open standards mean wider compatibility across programs and platforms.
- F/LOSS means community, active user groups eager to share and comment.
- F/LOSS gives flexibility
- F/LOSS allows access to developers and a chance to have an impact on tools
- F/LOSS gives security and backwards compatibility

We need to also consider how developers and projects benefit from having professional users. Consider the following:

- professionals mean professional problems
- professional users raise the bar
- professional users are by necessity a dedicated group

How do we encourage the professionals to use F/LOSS?

- get them while they're young: break the loop by teaching F/LOSS graphics software first
- ease compatibility: if it exports and prints nicely, switching is less of a problem
- dumb it down a little: most users don't want a complicated installation process
- get them know it exists
- improve distribution channels, aim for friendliness and accessibility.



Sintel

<http://durian.blender.org>

Every year, the Blender Foundation produces a major project. The following stills and process documents are from the upcoming Blender film Sintel.







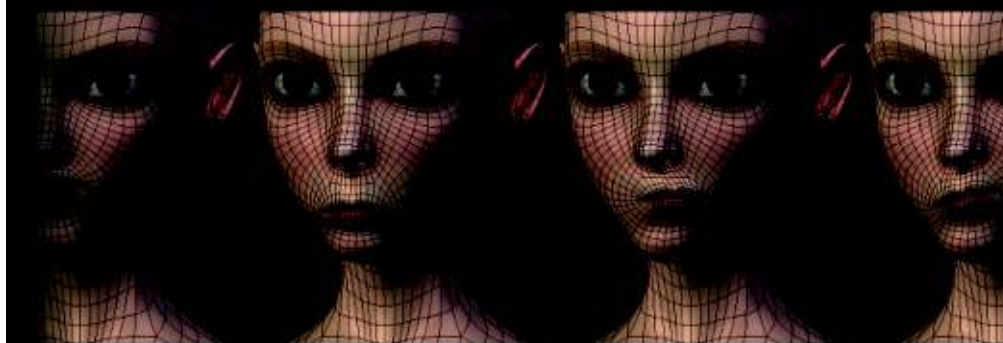


default

cornerUp.L

narrow

cornerUp.R

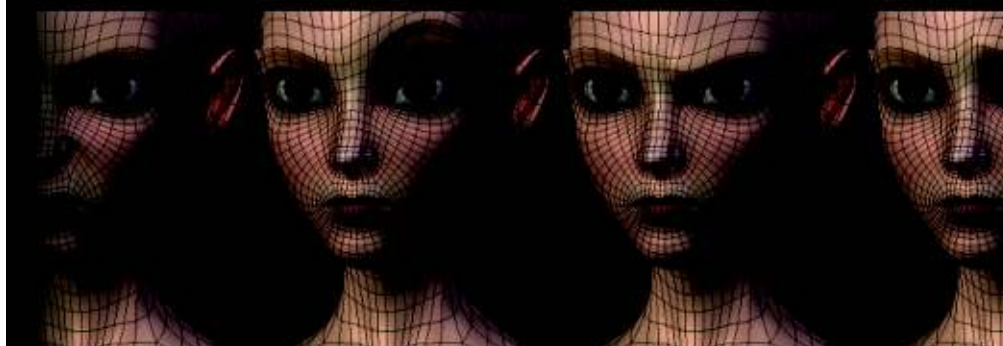


mouthUp

mouthDn

mouth.L

mouth.R



browSneer.L

browSurp.L

browMad.L

browSneer.R

The tank illustration in the cover was inspired by an analogy from Neal Stephenson's 1999 essay, "In The Beginning There Was The Command Line".

You can find it online, but sadly we cannot reproduce the relevant portion of the essay since it is not covered by a permissive license such as Creative Commons.

