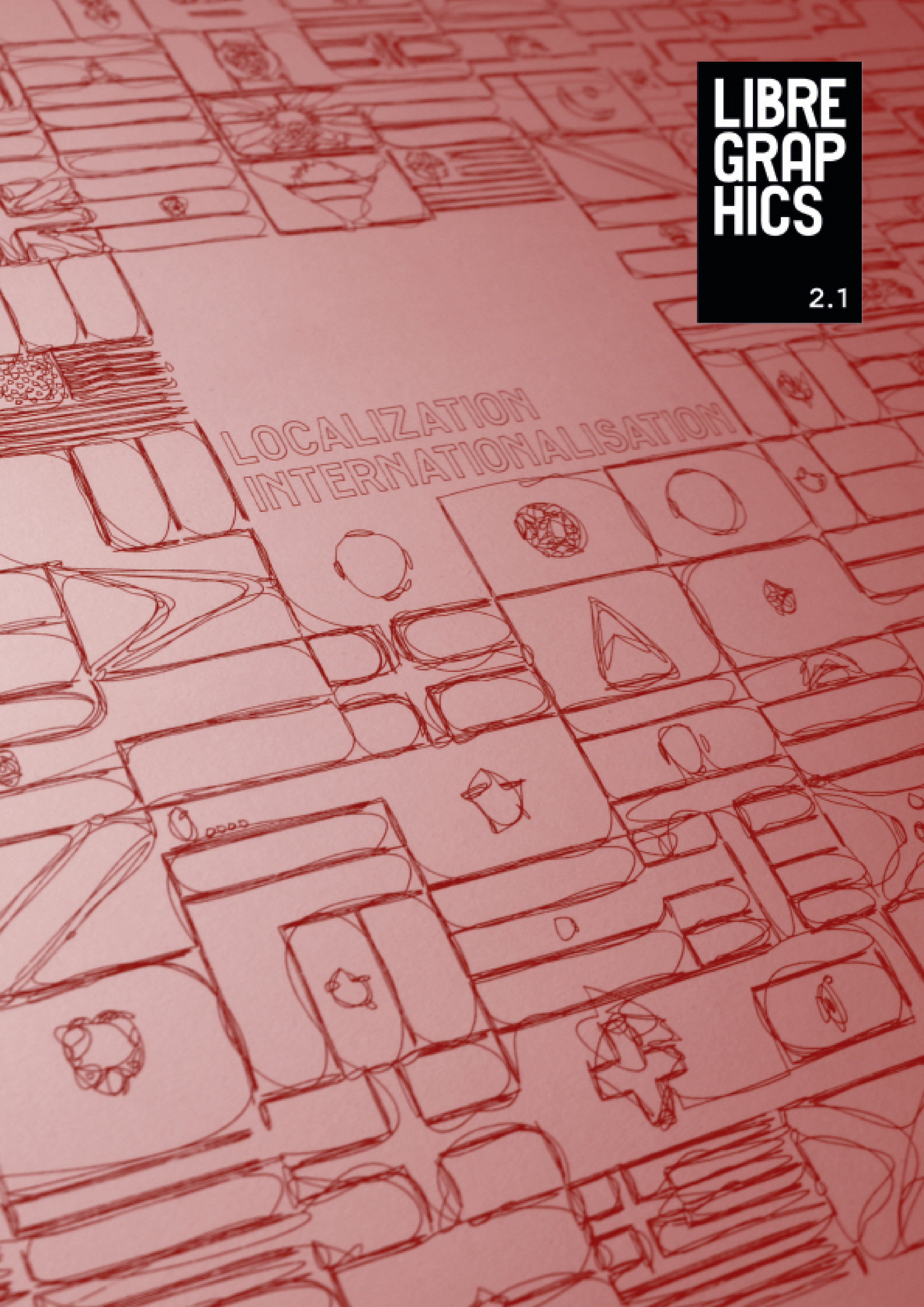
LIBRE GRAPHICS

2.1



LOCALIZATION INTERNATIONALISATION

Arabic
Shift JIS
Korean
n Europe
MIK
UTF-7
CP8
UTF-EBCDIC
BCODE
ISO 88
ANSEL
JIS X 0213
IATA c
e
Windows-1253 Greek
92
on System
KOI8-U
ISO-202
dows-1256 Arabic
Hong Ko
uthern European
Shift_JIS
59-10 Western Europe
UTF
Floriography
SPARQCode
Leetspeak
JIS X 0208

INDEX

Localisation and internationalization

MASTHEAD

EDITORIAL TEAM

Ana Isabel Carvalho
ANA@MANUFACTURAINDEPENDENTE.ORG

ginger coons
GINGER@ADAPTSTUDIO.CA

Ricardo Lafuente
RICARDO@MANUFACTURAINDEPENDENTE.ORG

COPY EDITOR

Margaret Burnett

PUBLISHER

ginger coons

COMMUNITY BOARD

Dave Crossland, Louis Desjardins, Aymeric Mansoux, Alexandre Prokoudine, Femke Snelting

CONTRIBUTORS

Davide della Casa, Constant, Dave Crossland, Denis Jacquerye, Julian Oliver, Nikki Pugh, Pierros Papadeas, Antonio Roberts, Eric Schrijver, Susan Spencer, Danja Vasiliev, Vinzenz Vietzke.

Printed in Porto by Cromotema (WWW.CROMOTEMA.PT) on recycled paper.

Licensed under a Creative Commons Attribution-Share Alike license (CC-BY-SA). All content should be attributed to its individual author. All content without a stated author can be credited to Libre Graphics Magazine.

CONTACT

Write to us at
ENQUIRIES@LIBREGRAPHICSMAG.COM

Our repositories with all source material for the magazine can be found at WWW.GITORIOUS.ORG/LIBREGRAPHICSMAG

Localisation/Internationalization

ISSUE 2.1, February 2013
ISSN: 1925-1416
WWW.LIBREGRAPHICSMAG.COM

IMAGES UNDER A CC ATTRIBUTION SHARE-ALIKE LICENSE

Photo of ginger coons by herself.
Photo of Ricardo Lafuente by Ana Carvalho.
Photo of Ana Carvalho by Luís Camanho.
Photo of Antonio Roberts by Emily Davies.
Photo of Eric Schrijver by himself.
Notebook Mozilla Festival: photos by Flickr user paul_clarke. Image names, from top left to bottom right: Mozfest_10Nov_011, Mozfest_11Nov_048, Mozfest_11Nov_052, Mozfest_11Nov_010, Mozfest_10Nov_134, Mozfest_11Nov_004, Mozfest_11Nov_069.
“Crafting type” photos by Rob & Lauren Lim (*http://robnlauren.com*). These images can be found at *http://www.flickr.com/photos/workturn/sets/72157631445394770*.
All the sketches from “Sketching Creative Code“ can be found at http://www.flickr.com/photos/11701517@N00/sets/72157631514318200. *Sketches names and authors, by order of appearance: Page 16: Suffragette by Sophie McDonald (source code: http://www.sketchpatch.net/view/9Zwh66llS8u). Page 17, from the top to bottom: Stretchy Points Play v2 by DARYL.Gamma and charles_m_dietrich (source code: http://www.sketchpatch.net/view/Lqr5gWHLlg0). Tapestry 4 by DARYL.Gamma (source code: http://www.sketchpatch.net/view/yEeWLKQDCrM). Peaks v3 by Kim Asendorf and DARYL.Gamma (source code: http://www.sketchpatch.net/view/ZC83eYHWq1r); Knots Weave v3 by DARYL.Gamma (source code:http://www.sketchpatch.net/view/oOLMrflXAK8). * Page 18, from the top left to the bottom right: Kind Of Pollock by DARYL_Gamma and eiichiishii1 (source code: *http://www.sketchpatch.net/view/GeIursTDPjj*). Cushion by DARYL.Gamma and conroy (source code: *http://www.sketchpatch.net/view/L8arQDcCGYZ*). 80's mania by Sophie McDonald (source code: *http://www.sketchpatch.net/view/naotkeywUPi*). Where Is She Going by Sophie McDonald and DARYL.Gamma (source code: *http://www.sketchpatch.net/view/auPpktqHG9o*). *Page 19: Magma Jumping Peaks v2 by Kim Asendorf, DARYL.Gamma (source code: http://www.sketchpatch.net/view/N6fOWgdKacJ).
“Crafting type” photos by Rob & Lauren Lim (*http://robnlauren.com*). These images can be found at *http://www.flickr.com/photos/workturn/sets/72157631445394770*.
Credits for the photos of the piece “La Langue Schaerbeekoise/De Schaarbeekse Taal.” Photos from pages 30, 31, 33 34 and 35 by Pablo Castilla (with the exception of the first top photo in page 34). All other photos are by Constant.
All images in the “Showcase” section can be attributed to the creators mentioned therein. All are licensed CC BY-SA with the exception of “La Langue Schaerbeekoise/De Schaarbeekse Taal” which is under the Free Art License.

IMAGES UNDER OTHER LICENSES

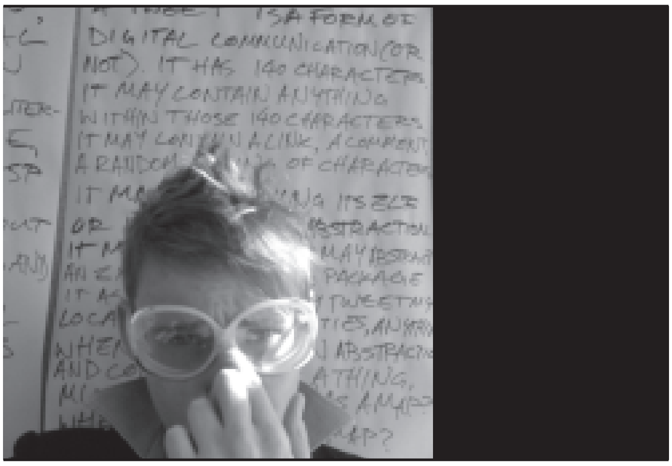
Illustrations from the “Picture Feature” are in the Public Domain and can be found on WikiMedia Commons.
Denis Jacquerye's photo by Michael Murtaugh (*http://gallery3.constantvzw.org/index.php/LGRU-Friday-Shared-Vocabularies/P2245659*) under the Free Art License.

GENERAL

Advertisements, with the exception of those representing Libre Graphics Magazine, are not necessarily covered by the blanket CC BY-SA license. It is best to check with the projects they represent before reusing them.

Localisation and internationalization

GINGER COONS


DIGITAL COMMUNICATION (OR
NOT). IT HAS 140 CHARACTERS
IT MAY CONTAIN ANYTHING
WITHIN THOSE 140 CHARACTERS
IT MAY CONTAIN A LINK, A COMMENT
A RANDOM
OF CHARACTERS

The simple difference between an “s” and a “z” is a small but vital representation of the theme for this, the first issue in the second volume of *Libre Graphics magazine.* While the two letters sound the same when used in words like “localisation” and “internationalization,” the cultural baggage attached to them differs. They indicate the way small regional differences are played out, the way choices are made on national and regional levels, for reasons of culture, heritage or simply backlash.

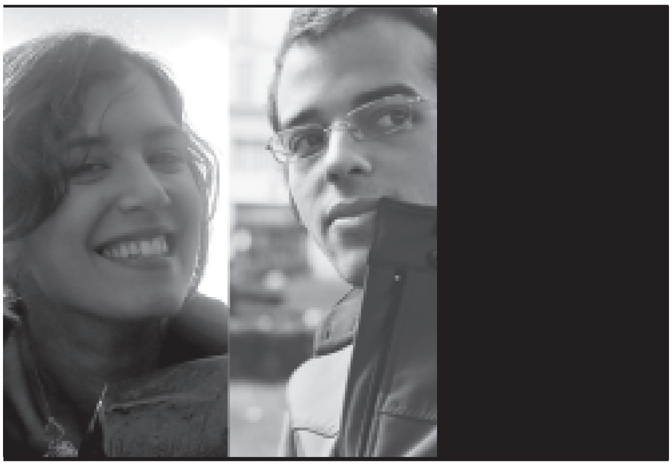
We react to regional differences, as well as efforts at internationalisation, in varied ways. In a world that has become increasingly globalised, we may hope for ways to communicate more effectively with others, or we may cherish our own regionally-specific terms and ways. We may create habits and classification systems which help us to trade knowledge and understanding with others, or we may take refuge in personal eccentricities.

In software, localisation and internationalization go hand in hand, with internationalisation forming the framework into which localization is slotted. Creating a piece of software representing a notional no-place allows customisation, serving very real some-places. In technology, art, design and everyday life, we see countless examples of artefacts walking the line between localization and internationalisation. From the no-place, wordless, pictorial instructions for assembling flat-pack furniture to the clothing hang tag written in six languages, we find different tactics for coping with our small world.

In this issue, we're exploring the unique problems of non-latin type, the hyper-localisation of custom clothing patterns and international visual languages, among others. We're mixing our zeds and esses, and we're leaving a little more regional variation in our writing than normal. We hope you'll join us in these explorations and for the explorations yet to come in this volume of *Libre Graphics magazine.*

Pruning branches

MANUFACTURA INDEPENDENTE



After the end of our first volume, we met in Brussels to carry out the Libre Graphics magazine camp—a week of reviewing and planning for volume 2.

One of our main tasks was housekeeping. Besides simplifying and updating our website, we spent quite some time taking care of our Git repositories. Our main problem was that all issues and other assets were together in a single, 8GB repository. A certain lack of discipline and knowledge about Git workflows ended up in a lot of cruft and leftover files that were mostly filling up space. On the other hand, we thought it would be useful to split our huge repo into smaller ones for better structure.

In the process, we created a few tools that helped us manage and prune the existing repositories. You can find them in the new 'productiontools' repo. Here are the other repositories that you'll find in our Gitorious page (see link below):

- **vol1issue1**
- **vol1issue2**
- **vol1issue3**
- **vol1issue4**
- **vol2issue1**
- **propcouriersans**: The official magazine typeface
- **documentation**: Photos, presentations, event coverage and other documentation
- **persistent**: General magazine assets (logo, columnists' photos, editing and style guides...)
- **productiontools**: Git and Scribus scripts written to help with a magazine workflow

In case you cloned our old repository before, you can safely trash it and clone the new ones—their file size is about half as large now.

Wiki

We had also created a wiki, running on the popular MediaWiki software. We used it as a great resource for syncing our notes and plans, but soon ran into problems. Automated spambots are eager to dive into MediaWiki installations, and that's what happened to our small wiki, where garbled usernames scribbled strange ads into our carefully-written notes. We spent a few hours cleaning up the mess and installing anti-spam measures, but even then we were getting seriously hit: even unable to create accounts or edit our wiki, the spambots kept fetching our pages repeatedly and non-stop. We decided to pull the plug on MediaWiki, and are now working to port the content to a smaller DokuWiki installation that will, hopefully, be easier to maintain.

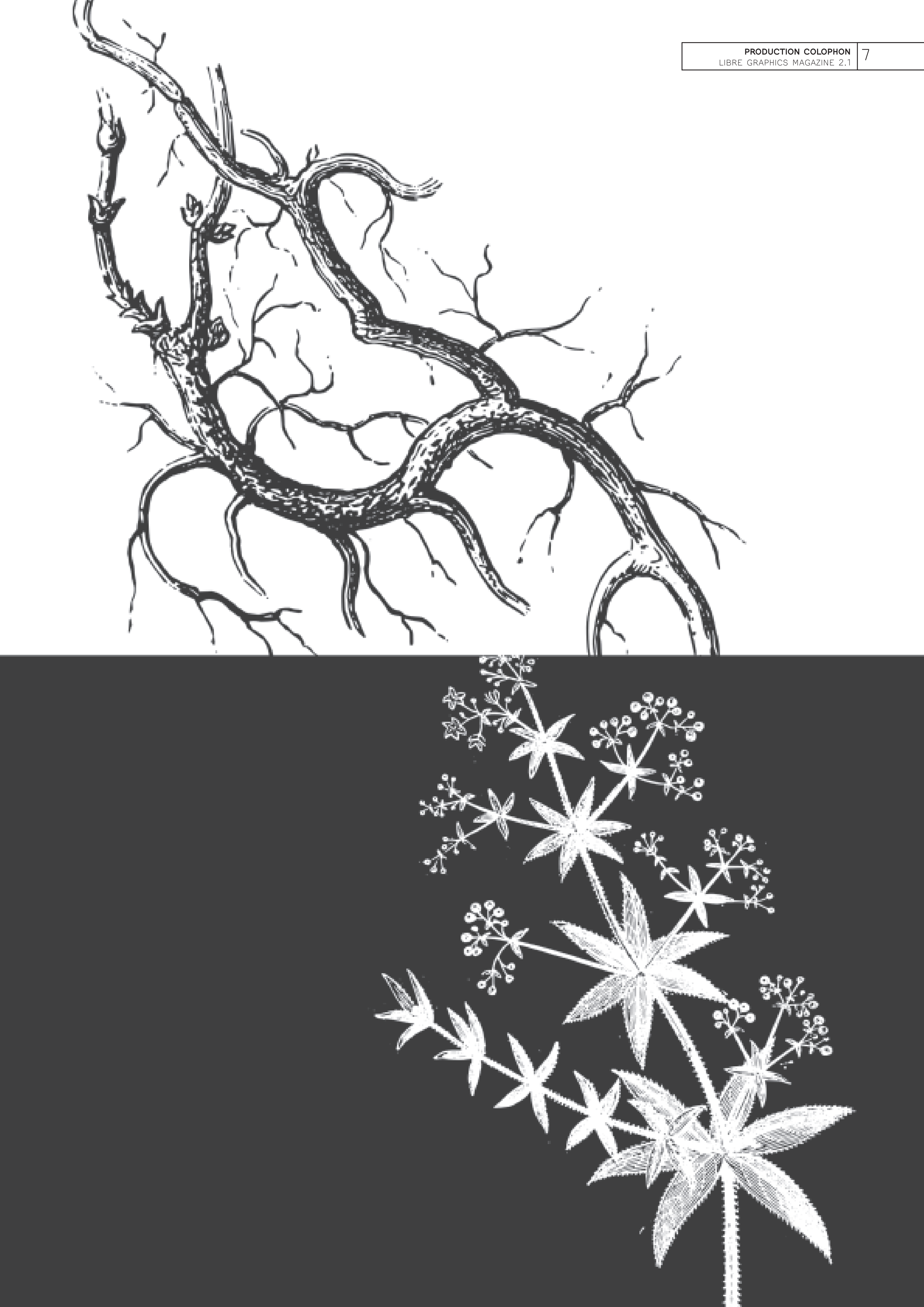
Online articles

Since the beginning of the Libre Graphics magazine project, we've discussed often how to combine print and web. Our priority always was to publish a physical version, but at the same time we wanted a good and clean way to provide the articles online in sensible formats. This would imply serious thought and work into selecting, installing and maintaining a CMS, as well as a set of scripts that can ensure that the online content is directly imported from the magazine repository. While an online article repository has always been on our collective TODO lists, we somehow kept it in a mid-to-low priority—print is our main focus.

In the meantime, we received a great tip from @TheRealPomax on Twitter: use the fantastic Pdf.js library from Mozilla to display the PDF versions of the magazine in your browser, without the need of a full download. This will be our next step after sending the issue you are reading to the printer.

Note that our production repositories, where all text, layouts and assets live, are publically available on Gitorious, so feel free to clone and look into what's inside!

http://gitorious.org/libregraphicsmag



NOTEBOOK

Where:

Mozilla Festival, London, UK.

What:

A three day, massively multi-track festival organized by Mozilla, promoting the Open Web and citizen involvement in networked media.

Best vehicle:

Over the weekend, a radio-controlled blimp roamed the building, taking advantage of the open spaces and connecting atria to pay visits to different corners of the festival.

Most wanted:

A pre-release version of Firefox OS was on display, promising to bring some new excitement to the smartphone operating system space. Operating under the tagline “the web is the platform,” Firefox OS is based on languages already standard in web development.

Best mash-up:

Developers and journalists hacking on open data, accessible public information and citizen political action spent their weekend showing off existing open journalism projects, and building the next wave.

Biggest release:

Building on the already popular Popcorn.js framework, Popcorn Maker was released to massive cheers. Promising easy development of contextually-enhanced video, Popcorn Maker is an easy, WYSIWYG interface for integrating text, web services and other assets into web video.

air mozilla

mozilla
Festival

numiko

NEW RELEASES

http://krita.org

Krita 2.5

The KDE-native painting and sketching application has undergone some useful changes in version 2.5. With themes designed by David Revoy, improved smearing capabilities, new shortcuts and better interoperability with GIMP and MyPaint, Krita is bringing its user experience to a higher level.

http://linuxlibertine.org

New version of Linux Libertine

One of our favourite font families, Linux Libertine has reached version 5.3.0. This release includes a brand new monospaced face as well as improvements to existing bold faces.

http://fr.flossmanuals.net/creer-un-epub

New F/LOSS Manuals book: *Créer un epub*

For French readers interested in the merits and requirements of the EPUB format, F/LOSS Manuals has created an indispensable guide. This book is a welcome addition to the growing collection of non-English F/LOSS Manuals titles.

http://mango.blender.org/download

New Blender movie: *Tears of Steel*

The hotly anticipated live action open movie project showcases Blender's capabilities as a special effects platform. As with previous projects from the Blender Foundation, the source files for Tears of Steel are released under a Creative Commons licence.

http://gimpmagazine.org

First issue of *GIMP Magazine*

Released in August 2012, GIMP Magazine is an online publication focused on practical aspects of GIMP usage and mastery. It offers tutorials, how-tos and review articles.

http://pinta-project.com

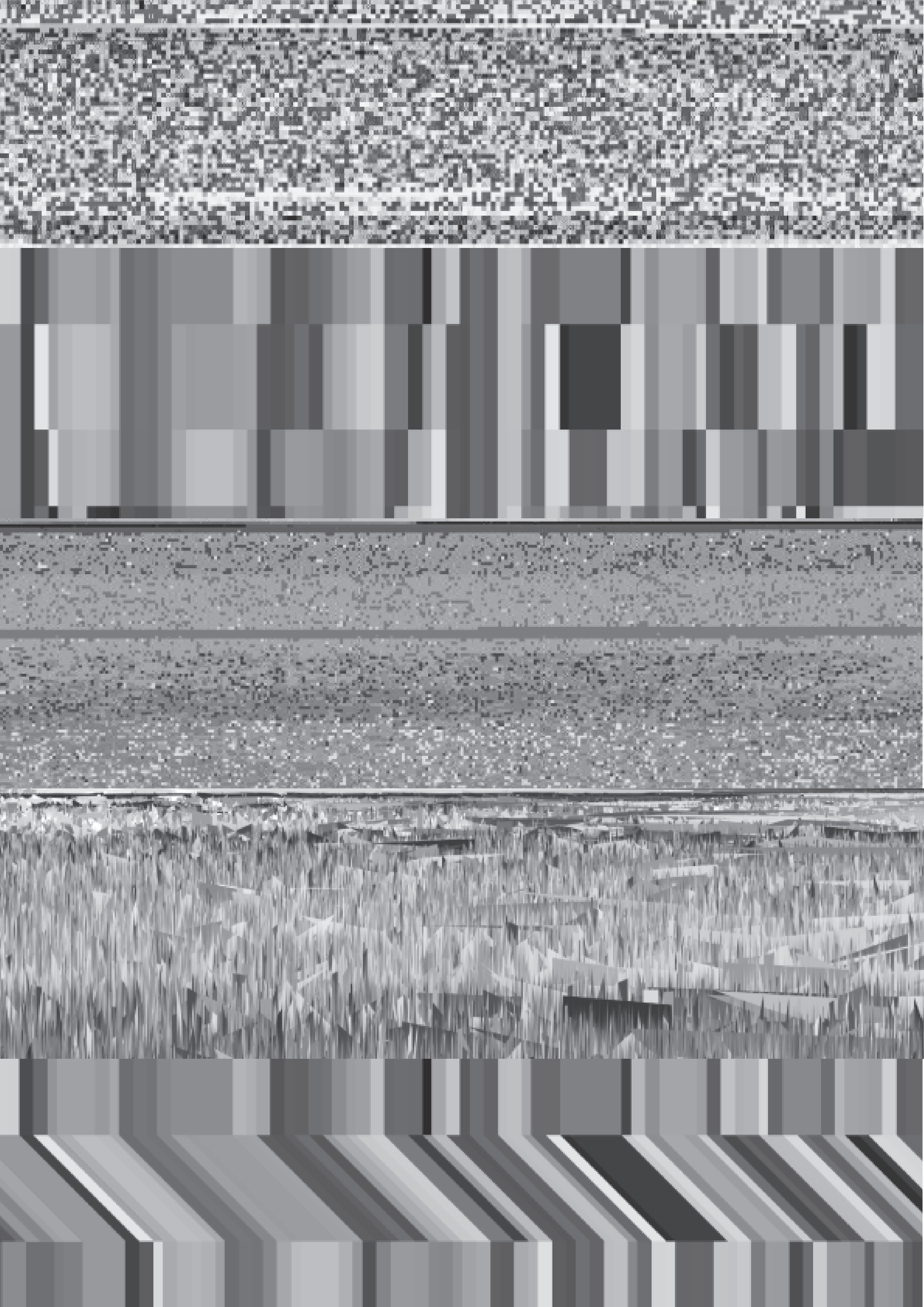
Pinta 1.4

A simple painting and editing tool, Pinta has undergone a collection of major updates prior to its latest release. Pinta 1.4 includes improved text support and expanded capabilities for its colour selection and magic wand tools.

BIRDFONT

a free font editor

HTTP://BIRDFONT.ORG



The transnational glitch

ANTONIO ROBERTS

American English is the common language of computing and the internet. That's quite unfortunate when there are so many talented non-English speakers building our websites and shaping our digital future. That potential aside, one only has to look at the programming languages themselves and even small things like web addresses to see a bias towards English. Functions in popular programming languages are derived from English and, while websites that are not in English exist, their URLs are always in English, with only the domain extension (.fr, .pt, .es, .cn, etc.) available to give the website a sense of cultural identity.

The English-language bias also extends itself to digital art. Creative programming languages like Pure Data and Processing still use English as their common language and present barriers to those who want to take part.
Is an English-only ecosystem really the way forward?

One area of digital art that I see transcending these barriers is glitch art. Glitch art is the aesthetisation of digital or analogue errors, such as artifacts and other “bugs,” by either corrupting digital code and data or by physically manipulating electronic devices. Glitching through physical manipulation of electronics has been popularized by the practice of circuit bending. If this sounds too vague, think of a television screen beginning to corrupt or a camera taking strange-looking pictures. Glitch artists try to capture and reproduce these types of ephemeral moments and display them as art.

The history of glitch art is very hard to trace. Glitch music (Aphex Twin, Autechre) has been around since the 1990s and with it, chaotic and noisy visuals like those of Gantz Graf by Autechre and Szamar Madar by Venetian Snares. In popular culture it has even broken out of the electronic music scene and can be seen everywhere from music videos by Kanye West, Xiu Xiu and Everything Everything to advertisements for MTV and The Biggest Loser. One only has to look at the “glitch art” tag on Tumblr or Flickr to see that it is an art form that has sparked the imaginations of people the world over.

Beyond the internet, digital arts and new media festivals serve as physical meeting places for those interested in digital art forms. Glitch art has found an audience at these festivals. Festival de Arte Digital in Brazil, AND (Abandon Normal Devices) in the UK and Transmediale in Germany are only a selection of venues which, at one time or another, have had sections devoted to glitch art. However, until 2002, there hadn't been a festival dedicated solely to glitch art.

In 2002, in Oslo, Norway, the Motherboard art group was the first to hold a large-scale glitch art event. Post-Oslo, glitch-specific events laid dormant for a time, until 2010, when the GLI.TC/H conference began in Chicago. Since then, it has taken place in 2011 in Amsterdam and Birmingham, UK. I attended the 2010 festival in Chicago and the 2011 festival in Birmingham. What quickly became apparent to me from this festival was the international appeal of glitch art. While the bulk of participants physically present at the 2010 festival in Chicago and the 2011 festival in Birmingham were English speaking, the contributions of art came from an international community of creators.

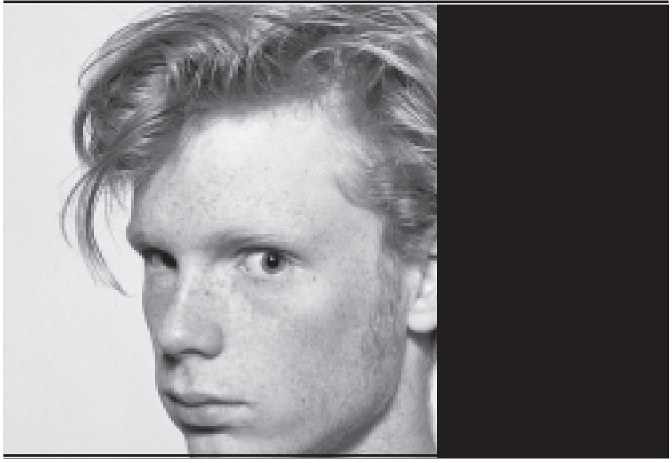
The popularisation of glitch art on the internet, the increasing number of festivals featuring sections devoted to glitch art and the overwhelming response to the GLI.TC/H festivals since 2010 only highlights its international appeal and suggests that the visual language of glitch art transcends spoken languages, cultural differences and location barriers. Glitch art needs no common speech. The process of throwing a camera into the air in order to produce glitches requires no proficiency in any language—programming or spoken—or professional qualification. The shattered screens, errors on computers and broken things will always evoke the same feelings of panic, frustration, annoyance, elation or glee no matter where in the world you are.

Antonio Roberts is a digital visual artist based in Birmingham, UK. He produces artwork that takes inspiration from glitches in software and hardware, errors and the unexpected.

http://hellocatfood.com

A journey through form fields

ERIC SCHRIJVER



The screen you are confronted with when you log into a WordPress admin and begin a new post is the base of the online writing experience. Little has changed from the very first content management systems, where a single un-styled text field would let you pour in your thoughts and send them off.
The only notable difference is that the interface now allows you to style your text and insert links. But this is not "what you see is what you get.' There is still little resemblance between the field you fill in here and the finished post that comes out at the other end.

ALL TOGETHER NOW

The traditional web form field is a lonely place. You are typing your information in the browser and only when you hit a submit button is the information sent to the server. This means that if you are editing a resource at the same time as someone else, the person who sends their information last will override the contributions of the predecessor. Content management systems put in place editing locks, allowing only one user to edit at a time. Wikipedia employs a sophisticated merging tool to merge various edits together.

When I started working with OSP and Constant in 2010, I met Etherpad. Etherpad presents you with an online document allowing you to start typing. As you do, you might see others connected to the pad start typing as well. There is no submit button. Everything is saved while you type so that it can be shown to your collaborators at the same time. You are no longer solitary with your text box.

Once you have used Etherpad to write, it becomes difficult to imagine writing collaboratively without it. In a book sprint in Rotterdam, we used Booki, which allows for sophisticated PDF and ebook creation. Yet like many content editing tools it imposes a single-user content lock on each chapter. At the end of the session it turned out everyone had used Etherpad to write their chapters together, before submitting to the Booki platform.

Will we see more Etherpad-style collaboration online?
In 2009, Etherpad was bought by Google and released under a permissive license. Parts of it have been re-used in Google Docs and the discontinued Google Wave. The technology behind it has remained notoriously difficult to implement, though. It is hard because it requires an intimate collaboration between the front-end and back-end of a website, a traditional division between the site displaying the data and the system processing and storing it. You need a kind of persistent connection not offered by traditional web servers.

This technology is getting more accessible. JavaScript, the language once used solely for creating scrolling ticker tape effects on web pages, is making its way into the back end, running in browsers and on servers alike. Frameworks like Meteor might make this technology accessible to a large number of developers in the same way that a framework like jQuery has enabled many designers to implement JavaScript effects and interactivity on their sites.

THE FUTURE IS BEING WRITTEN

Open source writing tools have been hugely successful. WordPress powers 16% of the web. Wikipedia is the go-to resource for factual knowledge. As the intimate link between Wikipedia and its editing software shows, we can’t talk about the writing technologies of the future without talking about the texts of the future. Writing tools rise in response to, and at the same time give shape to, what is being written.

We immediately recognize the tone of Wikipedia even if an article is written by thousands of people. The tool, and the process it entails, stimulate a certain kind of discourse. At the same time, these tools have responded to needs and desires that were there already — the reason they were invented in the first place. Before one invents a writing tool, one might need to ask: “What do I want to write?”

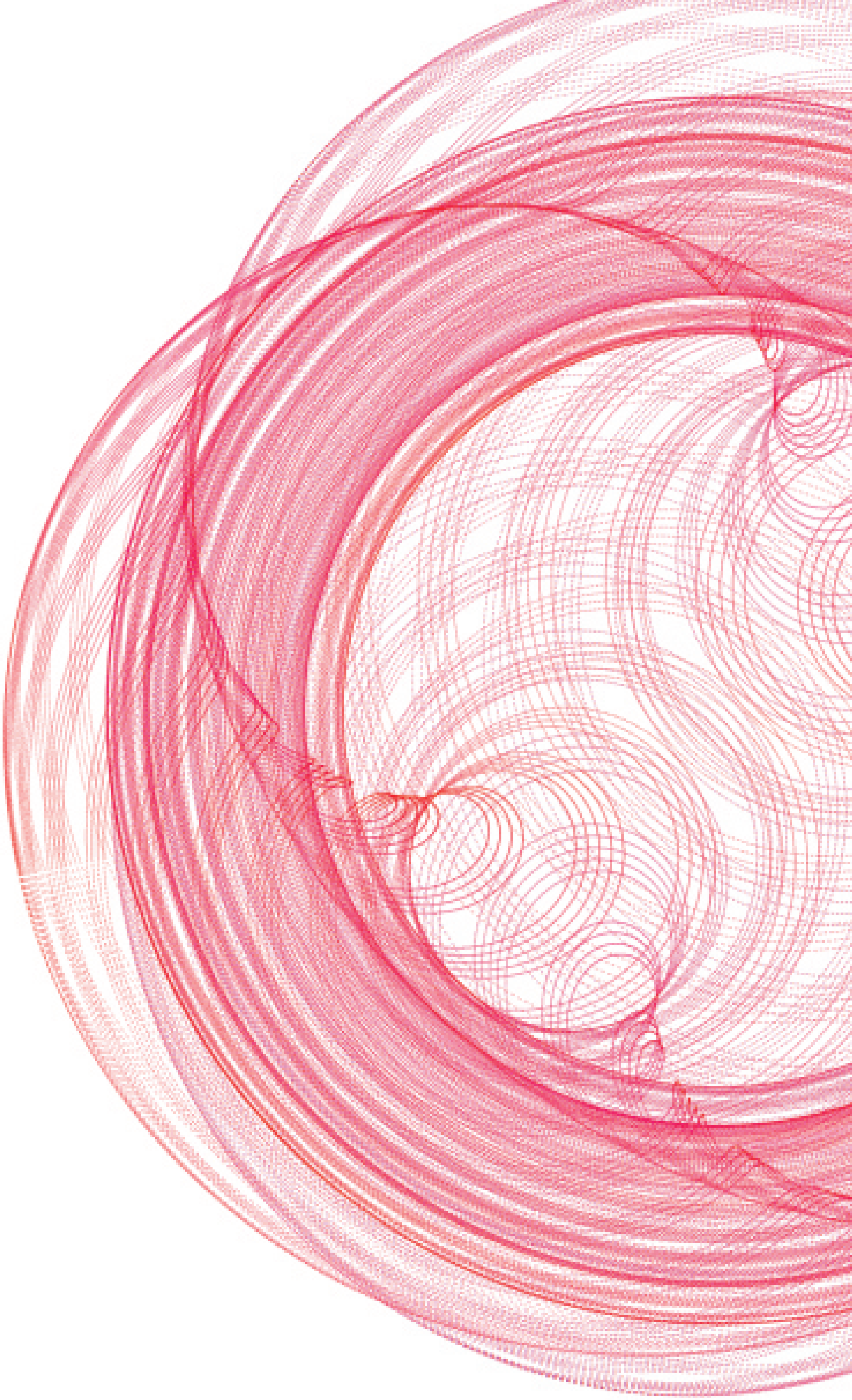
Sketching creative code

DAVIDE DELLA CASA

SketchPatch is a hobby project, a programming playground site that Sophie McDonald and I launched in the summer of 2009. The originating idea for the site was to build a place where anyone could create, show and collaborate on Processing sketches using only a common web browser.

The need for this actually requires some explanation: at the time of sketchPatch's inception, there were no sites catering for online creative coding. OpenProcessing was already popular as a way to show uploaded sketches, but the elements of online editing and collaboration weren't there. When we launched, only one other site (HasCanvas) was playing with similar concepts. As strange and remote as it might seem now, in 2009, online coding was new territory to play with. To put a figure like "2009" in a proper "online technology time" perspective, imagine that Google Chrome was only in version two (we are well into the double digits at time of release of this issue) and had just started to support mouse wheel scrolling and form auto fill. Internet Explorer 8 had just launched, with still negligible uptake, and Java was still a popular technology for interactive comment.

In that context, Sophie and I were thinking: What if people didn't need to download Processing? What if they just went on a page, scribbling stuff and playing with each other's code? It really was a cool idea, particularly at the time. A few months down the line, and uncountable decisions about look and feel and technical struggles later, we had sketchPatch ready for everyone to play with.

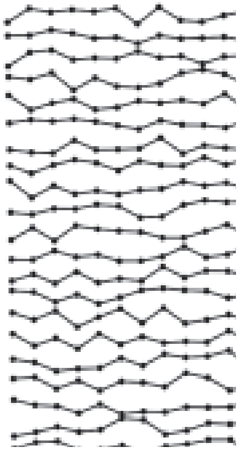


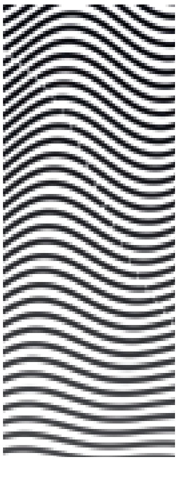
sketchPatch for education

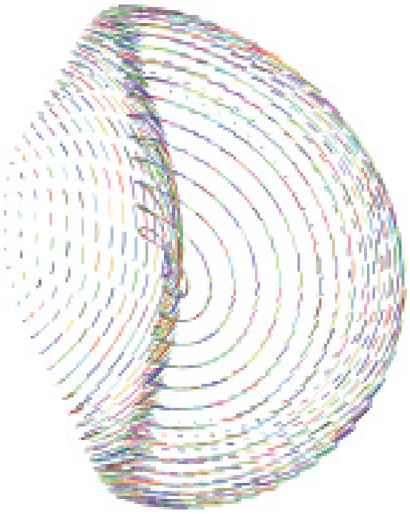
One of the ideas behind sketchPatch was to make it immediate and accessible enough that it could be used to learn about coding. That did drive our decision to go with a “light and fun” look and feel as opposed to a “dark and techie” theme. We also fleshed out a “hint” feature that would allow anyone to pick up the basics of Processing with a few examples.

With our friends at dorkbotlondon, we tried out sketchPatch as a learning and collaboration platform. Working in the round, everyone started to write a simple sketch. After five minutes, the sketch would be passed to the person on the right, who would do some changes and add bits to the original sketch. Since the sketches are all online, passing a sketch is simply a matter of handing over a link or a sketch title. We repeated the passes again and again, until everyone had dabbled with all of the sketches. We had really good feedback from the session. The lack of a deep sense of ownership of any particular sketch really unlocked some free range creativity. The passing method also scaled very well to the knowledge of each participant. A beginner could merely tweak a number, duplicate or delete a line, or add a simple statement. Experts would remix sketches in more complex ways to get more elaborate tweaks.

Following this positive and encouraging session, Sophie scaled things to a wider school context. She held almost a dozen sessions in colleges (you can see some of the activities at *http://blog.sketchpatch.net*). I think of myself as a very jaded creative coder, but every time I saw the sketches from those school sessions, which were often a first attempt at coding for the students, I’d always find a number of them to be laugh-out-loud funny or quirky and interesting.

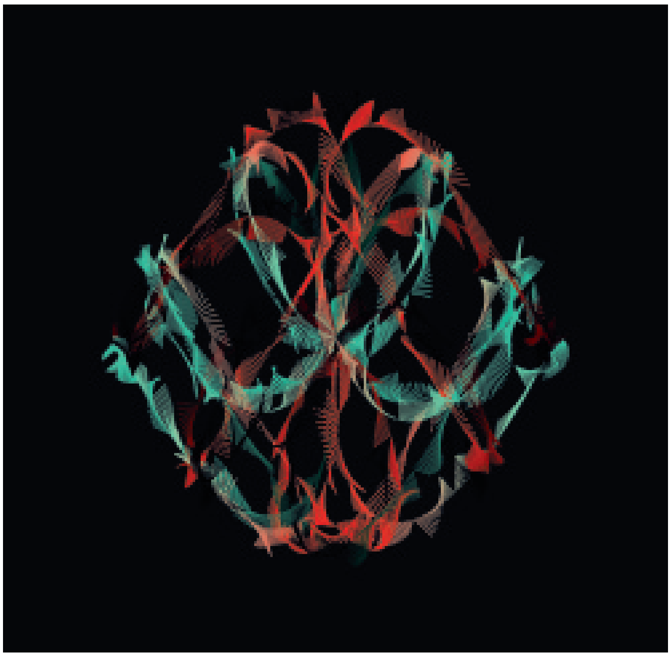


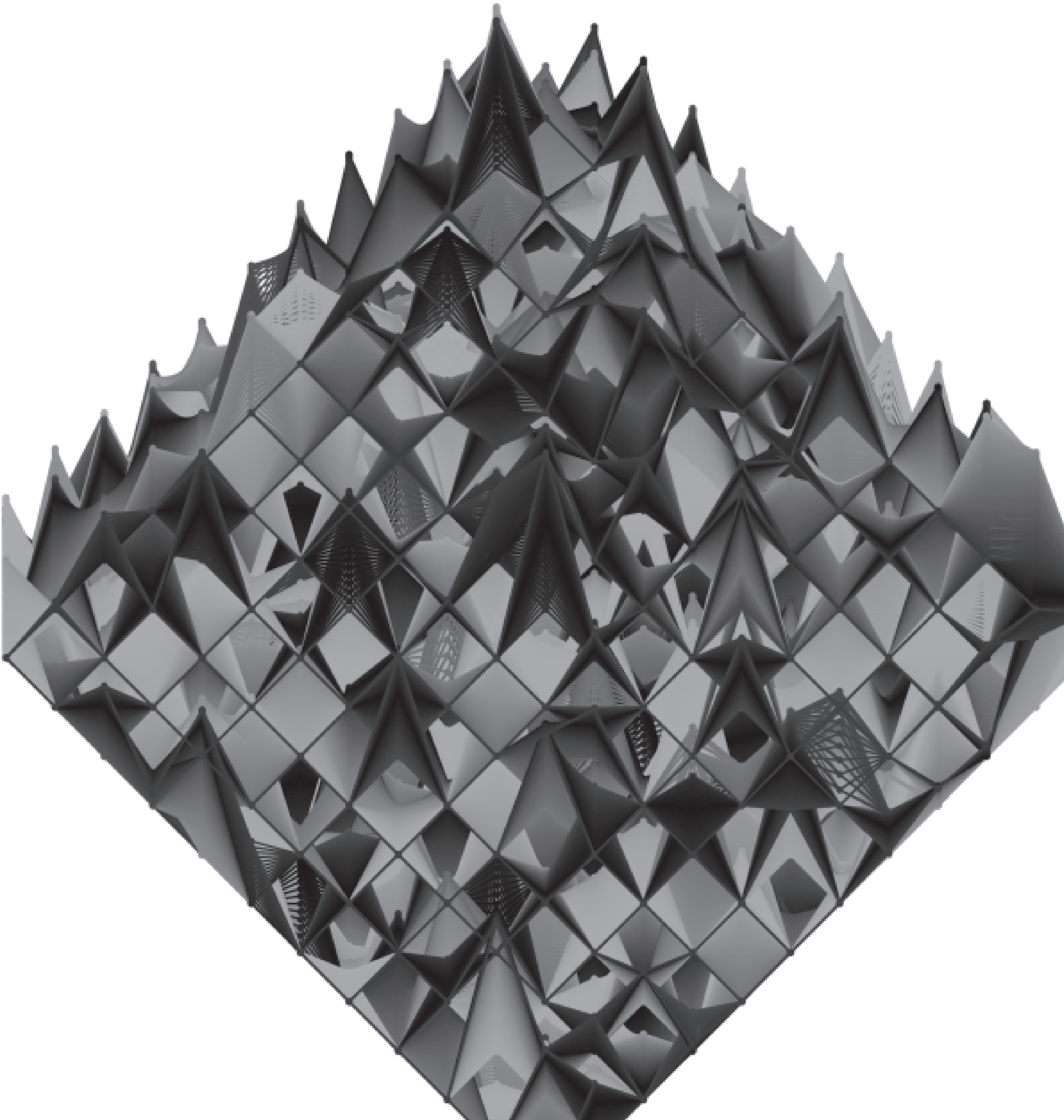


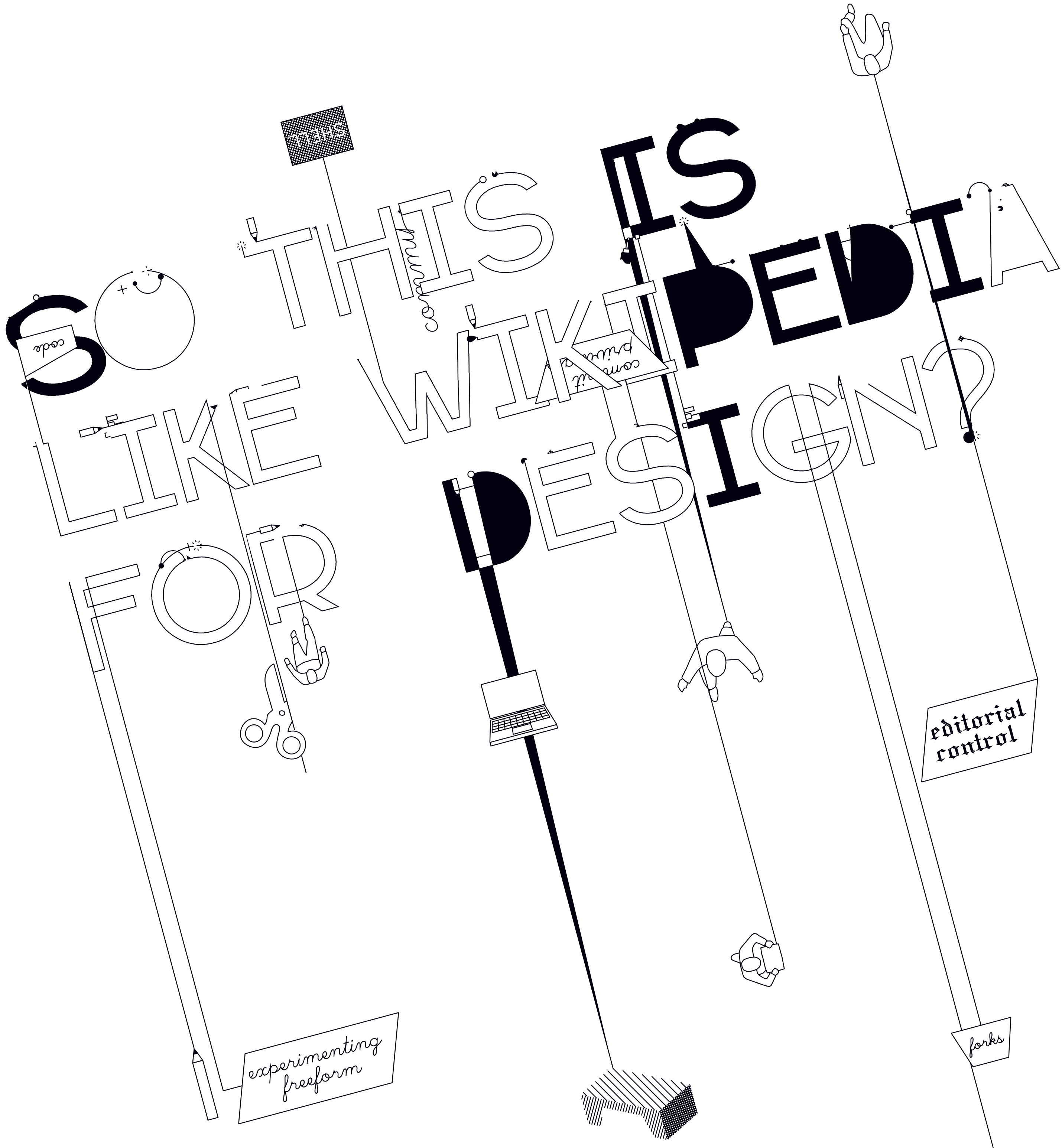


The impact of sketchPatch

Projects built on sketchPatch are fertile many times over, even before and after they exist. They are fertile ground for the people who play with them as users. They learn things, they have some fun. They don't necessarily come back, but a project like sketchPatch touches hundreds of people, even if for maybe only five minutes and to a limited degree. We do know that we've influenced a lot of people, even if only ever so slightly, and gave them a taste of first-hand creative coding. But that's just one level. Sophie and I learned a great deal from it, and met and dealt with dozens of great new people because of it. And the outcome is not just a website. It's an open source platform that keeps giving even in the form of entirely new incarnations, such as the LiveCodeLab project and workshops at MzTek. It is an absurd amount of work and stress at times,but I confess, I still look at this IE6-compatible hobby horse from 2009, which is beginning to show its age with subtle and not so subtle cracks, and I think: I don't know how long this will be able to stand up for, but isn't this just the best site ever.







FUTURE TOOLS
MADRID 10-13 APRIL
MEDIALAB PRADO

WWW.LIBREGRAPHICSMEETING.ORG

LIBRE GRAPHICS
MEETING 2013

¡MADRID! MEDIALAB PRADO

Culture Programme

Education and Culture DG

Libre Graphics Research unit

LAFKON.NET

SMALL & USEFUL

COMPILED BY **VINZENZ VIETZKE**

There's an adage in the software world: programs should do one thing very well. In that spirit, we offer you a round-up of small and useful programs and resources which do one thing particularly well.

http://www.cityinthesky.co.uk/opensource/pdf2svg

pdf2svg

A tool requiring some familiarity with the command line, pdf2svg directly converts PDFs to SVGs, exactly as it promises in its name.

http://www.knotdraw.org

Knotter

Knotter is a Celtic knot designer with rich customisation capabilities and robust export options.

http://www.blackfiveimaging.co.uk/index.php?article=02Software%2F05CMYKTool

CMYKTool

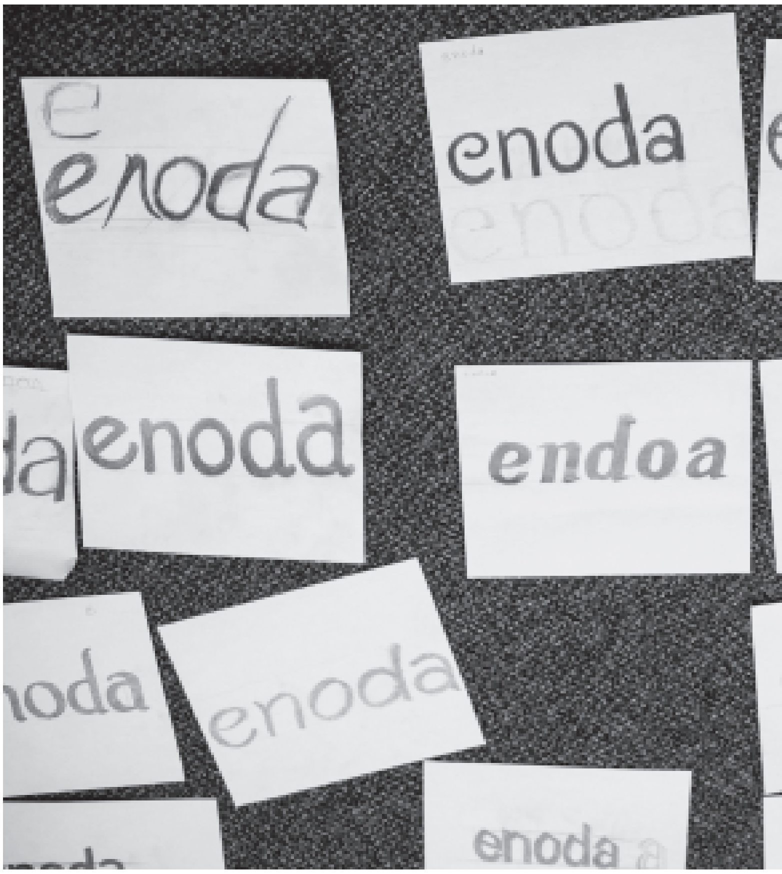
CMYKTool converts images from RGB to CMYK and back again, with profile embedding capabilities.

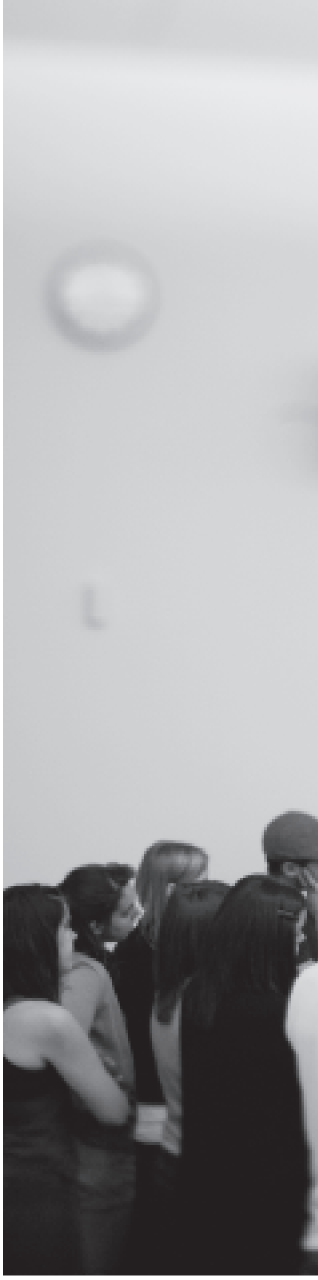
http://www.selapa.net/swatchbooker

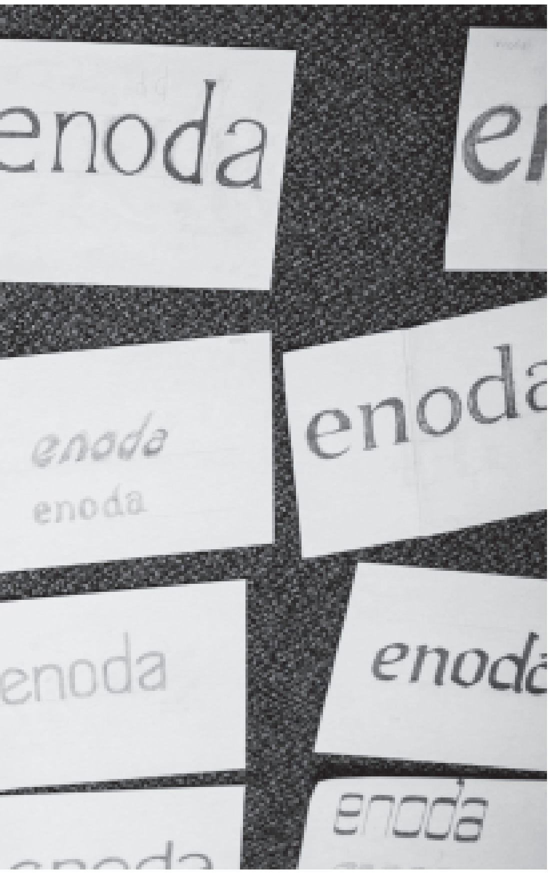
Swatchbooker

SwatchBooker reads and writes a variety of colour swatch formats, both F/LOSS and proprietary, as well as supporting colour profile assignment and name changing.

MOUNTA
Westfield
RUNNE
Darken
PUBLI
Vendu
CO
Te
H
M


enoda
enoda
enoda
endoa
noda
enoda
enoda a





Type design, around the world

DAVE CROSSLAND

Love letters? Many designers, artists and writers do. But the process of crafting typefaces has been shrouded in mystery. The Crafting Type project started with the motivation to change this, by holding workshops which push participants through the entire type design process: from the secrets of sketching to digital drawing techniques to OpenType features, using only Free/Libre software—mainly FontForge, plus some Inkscape and Scribus.

A number of professional type designers including myself, Alexei Vanyashin, Eben Sorkin, Octavio Pardo and Vernon Adams, have been running Crafting Type workshops in venues around the world. The workshops are held at design schools with attendance from both students and local professionals. Despite the variety of backgrounds in design, since all participants are beginners at type design, the mix of students and professionals adds a real buzz to the room.

In addition to teaching the workflow, each participant leaves the workshop with a functional OpenType font with key glyphs, and the knowledge about how to take the design all the way to a complete and useful typeface design.

The first workshop was held in Edmonton, Canada, in August 2012, as a 5 day boot camp for students and professional graphic designers. Kyle Fox and Jeff Archibald, a pair of Edmontonian designers, knew I was in town and proposed we run a workshop. Two months later they had organized one for 37 people. In the fall, Alexei Vanyashin set up a series of workshops in Ukraine, where there is also a lack of type design education available to local designers. We're hoping to do more, bringing type design know-how to designers everywhere. Get in touch if you'd like to see a Crafting Type workshop in your city.

http://craftingtype.com

MADRID
Malasaña
Plaza de España
Chueca
Salamanca
Calle de Goya
Calle de O'Donnell
Jardines del Campo del Moro
Paseo del Arte
Parque del Retiro
Barrio de la Latina
Cementerio de la Almudena
Moratalaz
Arganzuela

Styling maps like the web, for the web

PIERROS PAPADEAS

Providing up-to-date, beautiful geographic information has always been the challenge of cartography. Using proven web technology practices to build on top of editable open data from sources like OpenStreetMap is one way of rising to the challenge.

TileMill is a F/LOSS tool which allows cartographers to quickly and easily design maps for the web using custom data. Its editing and styling paradigm originates in web design principles. Each layer has an ID and a Class that can be selected to style through Carto .mss files, a style format specific to maps.

Deep down in TileMill we find Mapnik, a powerful, full -featured library supporting RGBA color, True Type fonts, rasters, patterns, and even SVG transforms.
The integration of web-friendly features like these is a boon to developers and designers. Using known technologies to create maps—which are in turn web-friendly—is a clever technique that has already produced a collection of beautiful maps.

Process

The process of designing a map in TileMill can be described in three steps. The first is sourcing data. OpenStreetMap (*http://www.openstreetmap.org*) is a great open data map source with a variety of different export sets (like *http://downloads.cloudmade.com*).

After import, each layer or data set can be assigned an ID (#) or Class. All classed or ided layers and data sets can be styled, either with regular css or using TileMill's pre-set custom properties.

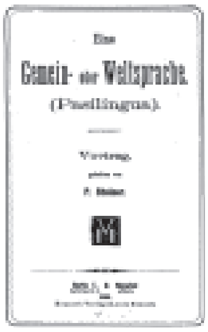
Finally, share the results. TileMill offers static image export, export of MBTiles, served as interactive, embeddable maps through MapBox (*http://mapbox.com*) or through your own server (*https://github.com/mapbox/tilestream*).

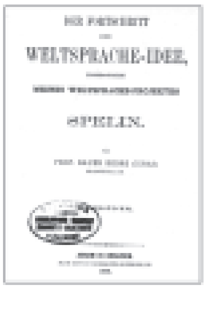
Happy mapping!

Chueca
Barrio de las Letras
Paseo del Arte
Calle de Barceló
Calle de Mejía Lequerica
Calle de la Beneficencia
Calle de San Mateo
Calle de Fuencarral
Calle de Campoamor
Calle de Argensola
Calle del General Castaños
Calle del Marqués
Calle de Doña Bárbara de Braganza
Calle de San Lucas
Calle de San Gregorio
Calle de Hernán Cortés
Calle de Hortaleza
Calle de Piamonte
Calle de Gravina
Calle del Conde de Xiquena
Calle del Almirante
Calle de Augusto Figueroa
Calle del Barquillo
Calle de Prim
Paseo de Recoletos
Calle de Recoletos
Calle de Villalar
Calle de Salustiano Olózaga
Calle de San Marcos
Calle de las Infantas
Calle de la Reina
Gran Vía
Calle de Alcalá
Calle de Valenzuela
Paseo del Prado
Calle de Alfonso XI
Calle de Montalbán
Calle Juan de Mena
Calle de los Madrazo
Calle de Zorrilla
Calle de Antonio Maura
Calle de Alfonso XII
Plaza de la Lealtad
Calle de Felipe IV
Calle Academia
Calle Alberto Bosch
Calle Espalter
Calle del Prado
Calle del Príncipe
Calle de San Jerónimo
Calle de Atocha
Calle Amor de Dios
Calle de Santa María
Calle de Moratín

Speaking across borders

GINGER COONS





In the late nineteenth century, Volapük gained traction. A constructed language, Volapük was meant to be international. For a time, it was, with speakers, publications and interest groups in a number of European and American cities. After not too long, it faded, replaced by other constructed languages like Esperanto and Interlingua. These languages—rising and falling over the years, gaining and losing speakers, occasionally evolving or fizzling out—reflect a desire to speak freely across borders and to bridge existing communication gaps.

The constructed languages of the nineteenth and twentieth centuries, from Volapük to Interlingua, were similar in their use of the Latin alphabet and their nearness to existing European languages. Those are traits we see today in a comparatively new way of communicating across borders—in Python, in C, in Java, each one a popular programming language. Even in simple markup languages like HTML, used in all the world's websites, some linguistic bias is built in.

Despite desires by the drafters of international languages to put all speakers on an equal footing, little things seep in. The borrowing of words and grammar from already-dominant European languages make constructed languages like Interlingua and Esperanto easier to learn for speakers of the source languages. Fundamental words such as “if,” “then,” “while” and “print,” common to swathes of programming languages, are hangers-on from English.

The programming and markup languages we use now, though less explicitly intended to promote international cooperation than the constructed languages of the previous century, carry out many of the tasks intended by those early international languages. And they come with the same artefacts: speakers the world over, conferences, books and newsletters, forums for discussion. Though intended to fill a different functional purpose, to ease communication between human and machine instead of between human and human, they offer a compelling side-effect for those who know them: a common second language.

AD BAL PÜK BAL

HAVE YOUR SAY.

SUBMIT YOUR WORK, ESSAY,
ARTICLE OR RANDOM PIECE.
YOU'RE THE COMMUNITY,
YOU'RE OUR FEATURE.

SUBMISSIONS@LIBREGRAPHICSMAG.COM

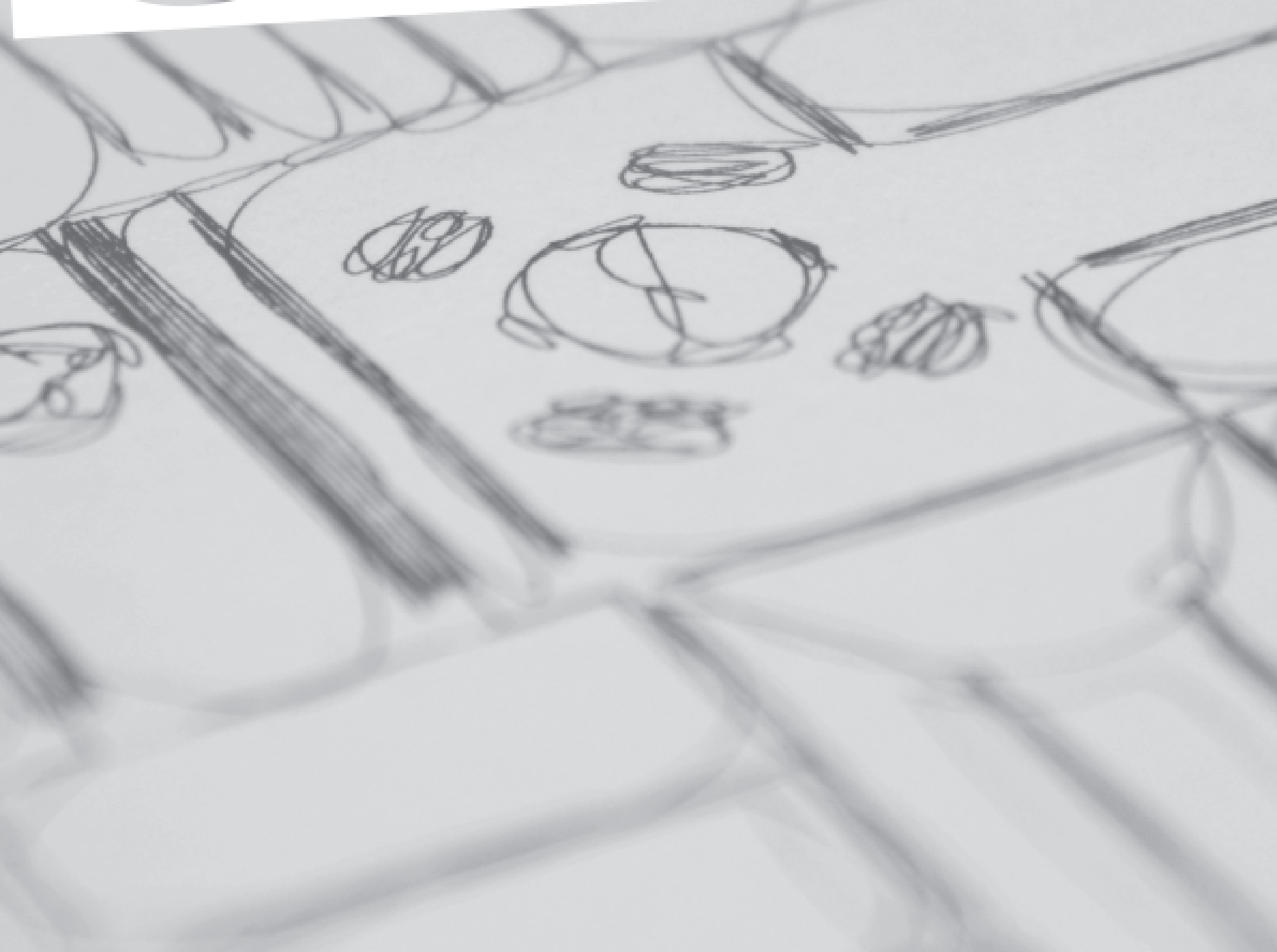
WE WANT YOUR FEEDBACK.
TELL US WHAT YOU THINK.
WHERE WE COULD DO BETTER.
WHERE YOU COULD HELP.

FEEDBACK@LIBREGRAPHICSMAG.COM

DISTRIBUTE THE MAGAZINE.
GET YOUR LIBRARY TO ORDER A SUBSCRIPTION.
SPREAD THE WORD.
HELP US REACH OUT.

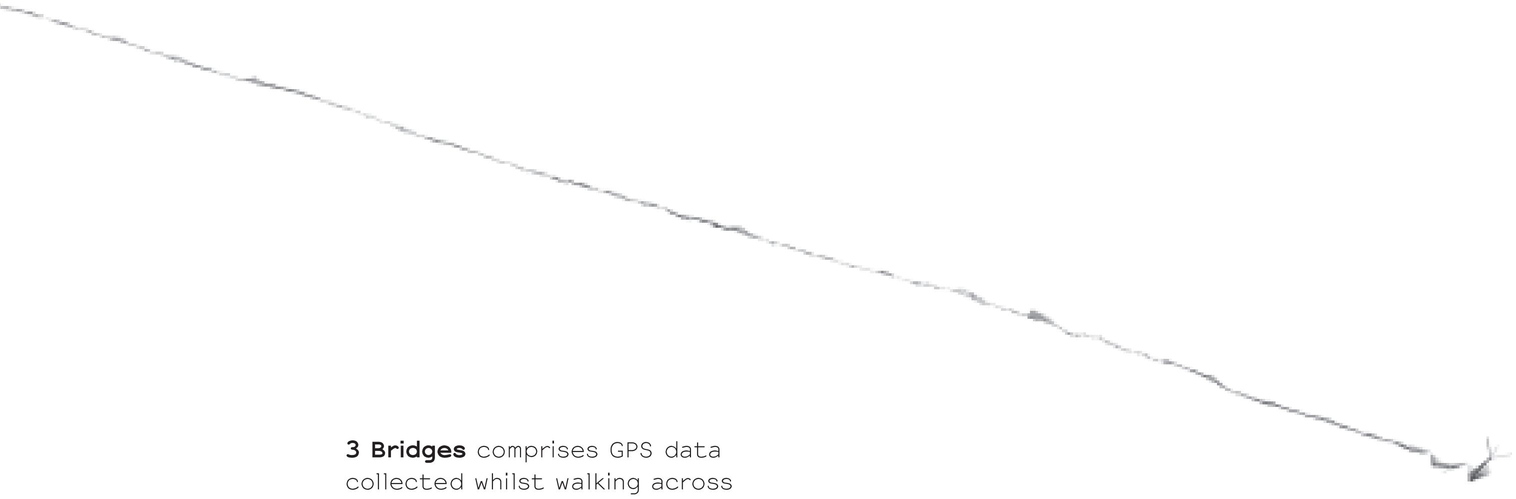
ENQUIRIES@LIBREGRAPHICSMAG.COM

Showcase



3 Bridges

NIKKI PUGH



3 Bridges comprises GPS data collected whilst walking across the Manhattan, Williamsburg and Brooklyn Bridges between Manhattan and Brooklyn, New York. The different signatures reflect different methods of bridge construction and the location of the pedestrian walkways within them. Collected data were visualised using Processing and Inkscape.

Nikki Pugh is a British artist working at the intersection of people, place, playfulness and technology. Her practice encompasses locative and digital media, walking, performative actions in public spaces (in turn including pervasive games), installation, physical computing and collaboration. Her work is currently being guided by the (evolving) Splacist manifesto (*http://npugh.co.uk/blog/splacist_manifesto_v2*).

La Langue Schaerbeekoise/ De Schaarbeekse Taal

CONSTANT

La Langue Schaerbeekoise/De Schaarbeekse Taal is a book and an online audio database presenting contemporary local words collected between 2009 and 2012 in the in the plurilingual neighbourhood around De Berenkuil/La Cage aux Ours, officially known as Verboeckhovenplein/Place Verboeckhoven, found in the multilingual municipality of Schaarbeek, in the bilingual Brussels Capital Region. The collection of words shows that amongst the two official languages of the neighbourhood, there is a great grey linguistic zone that relates to experiences, people and stories situated all over the globe.

175

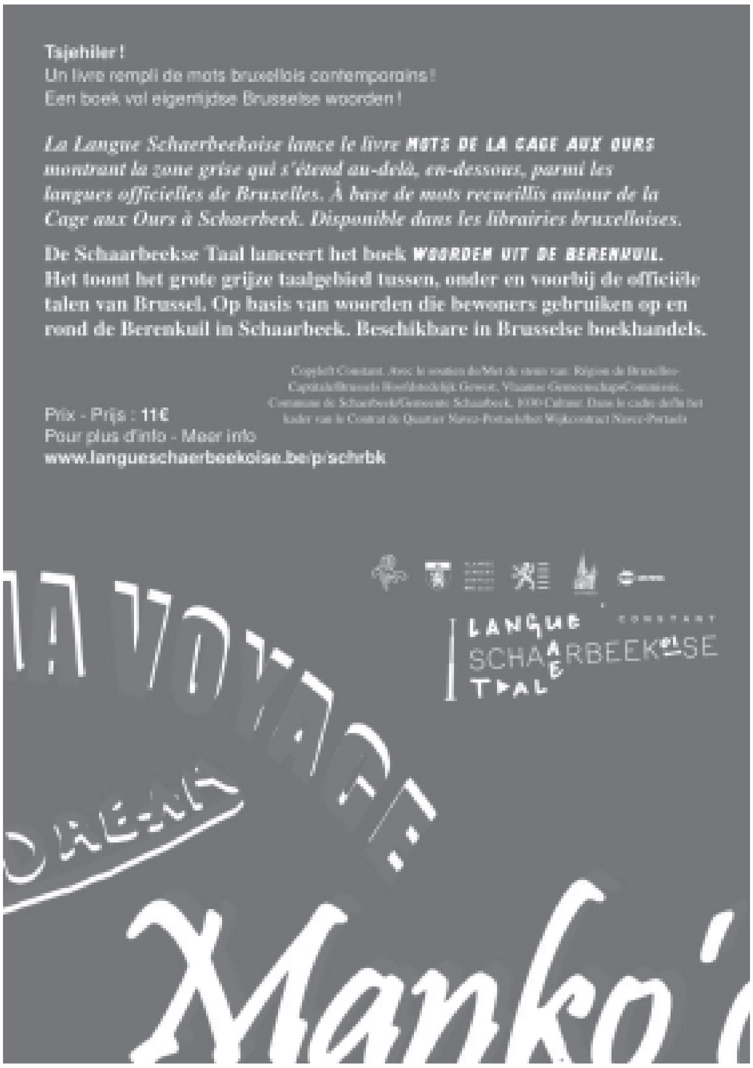
Snack la Cage
2

77

Arab, Turkish, Berber, Brussels, Swahili and Polish are some of the languages that colour the conversations in French, the language of the common ground. During discussions, meetings, film projections and walks, people offered us their words, with their definitions. These words have been organised in categories that show their effects: the Biographers, the Travel Agents, the Matchmakers, the Reporters, the Time Travelers.

An example is the Lingala word "Eza," a Travel Agent. Eza means "Being, a presence, a presence of possibilities." "It refers to here now or here there; it is her in this room, here in one hour, or a five minutes walk from here; and there is no doubt."

SORORITÉ QATRAN EZA PARTCHIS ? BONHEUR FRIOUL BAMOUN MANKO'O SHOP. MALANGUA ? MAMA NAPLES, VACANCIERS DE 28 JOURS.


Tsjehiler !
Un livre rempli de mots bruxellois contemporains !
Een boek vol eigentijdse Brusselse woorden !

La Langue Schaerbeekoise lance le livre **MOTS DE LA CAGE AUX OURS** *montrant la zone grise qui s'étend au-delà, en-dessous, parmi les langues officielles de Bruxelles. À base de mots recueillis autour de la Cage aux Ours à Schaerbeek. Disponible dans les librairies bruxelloises.*

De Schaarbeekse Taal lanceert het boek WOORDEN UIT DE BERENKUIL. Het toont het grote grijze taalgebied tussen, onder en voorbij de officiële talen van Brussel. Op basis van woorden die bewoners gebruiken op en rond de Berenkuil in Schaarbeek. Beschikbare in Brusselse boekhandels.

Copyleft Constant. Avec le soutien de/Met de steun van: Région de Bruxelles-Capitale/Brussels Hoofdstedelijk Gewest, Vlaamse GemeenschapsCommissie, Commune de Schaerbeek/Gemeente Schaarbeek, 1030 Culture. Dans le cadre de/In het kader van le Contrat de Quartier Navez-Portaels/het Wijkcontract Navez-Portaels

Prix - Prijs : **11€**
Pour plus d'info - Meer info
www.langueschaerbeekoise.be/p/schrbk

BONNE FETE MAMA

Materials for the La Langue Schaerbeekoise/De Schaarbeekse Taal project were prepared using OpenStreetMap, Scribus, Inkscape, LibreOffice and libgeos.

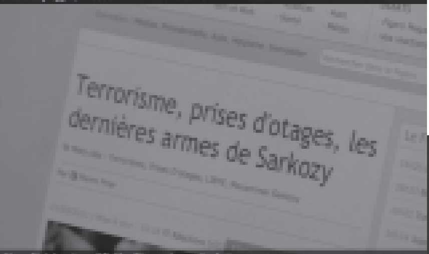
http://lalangueschaerbeekoise.be &
http://deschaarbeeksetaal.be

Starbucks Café, 75013 Paris

agent receives Newstweek device

ad by café visitors is modified by filters active on the device

ek device plugged, innocuous within its environment



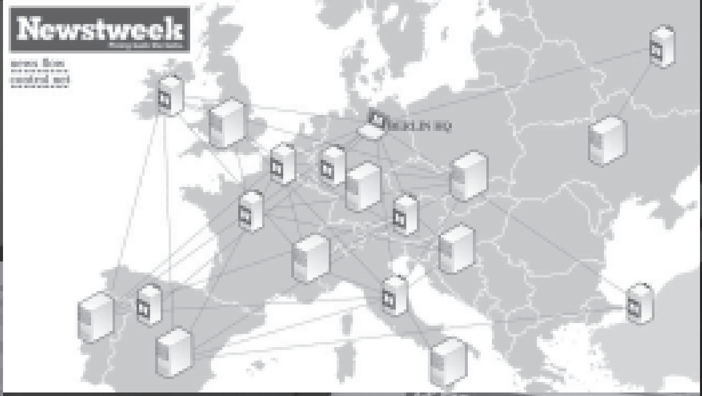
ad by café visitors is modified by filters active on the device

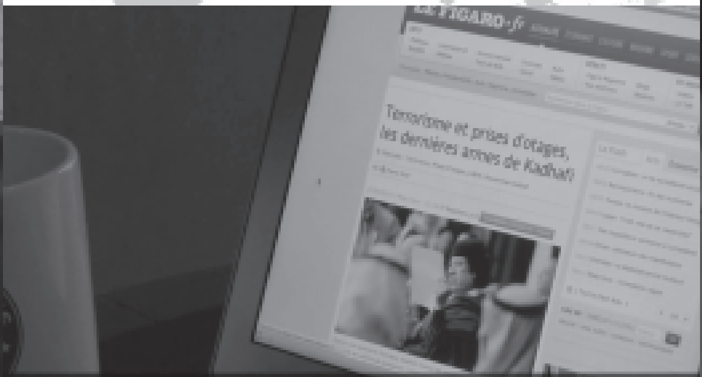
pares to Paris

ek devices ready for deployment across Europe

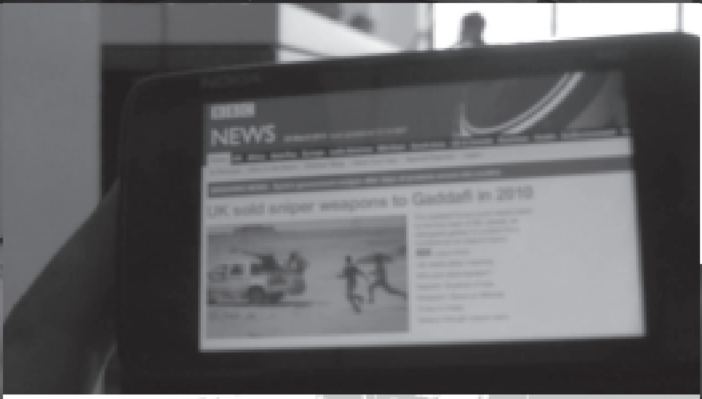
Istanbul agent receives Newstweek device

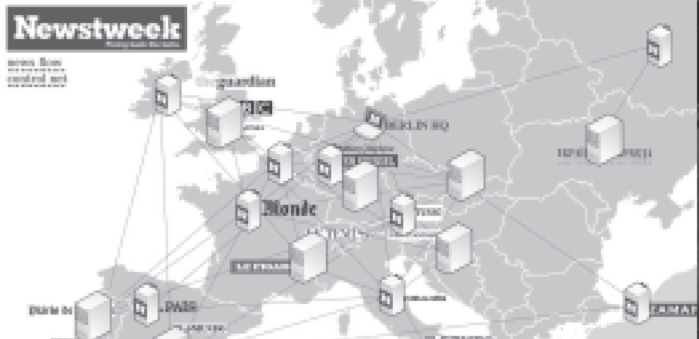
Paris agent receives Newstweek device

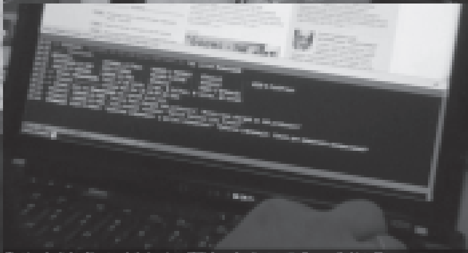



LE FIGARO·fr
Terrorisme et prises d'otages, les dernières armes de Kadhafi

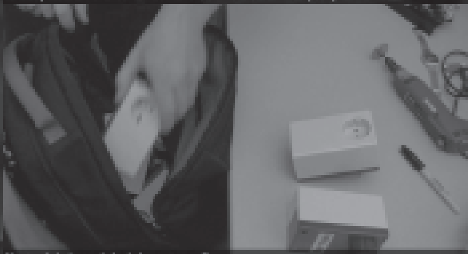
News read by café visitors is modified by filters active on the device



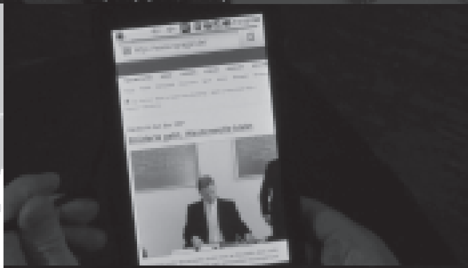


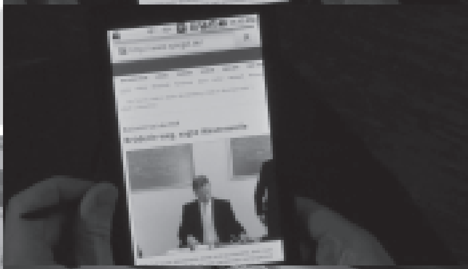


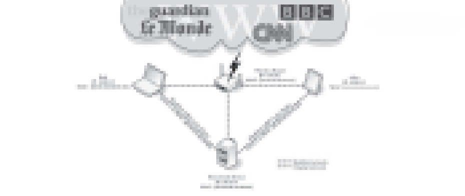
Text is uploaded to Newstweek devices in an IRC channel and automatically compiled into filters.



Newstweek devices ready for deployment across Europe





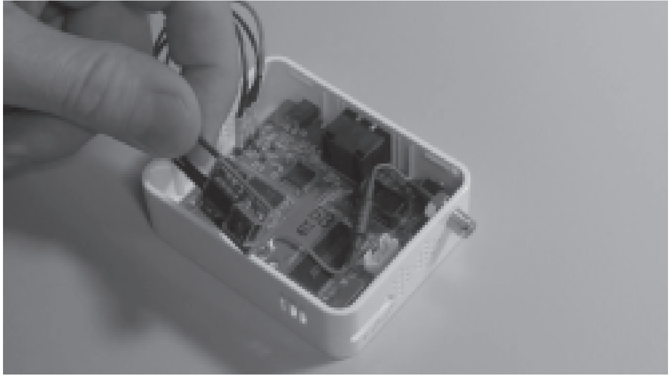


NEWSTWEEK: NETWORK MODIFICATION DIAGRAM

Newstweek

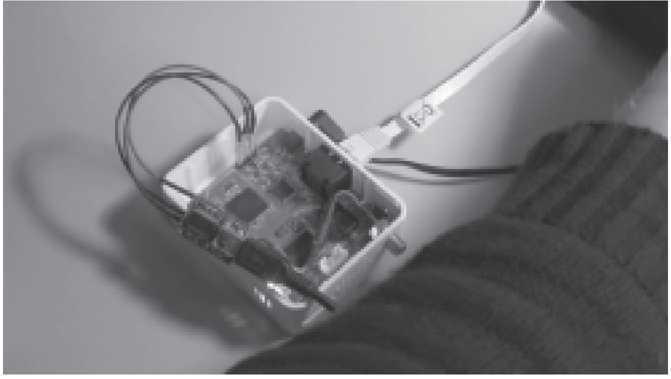
JULIAN OLIVER AND DANJA VASILIEV

In a world where hackers are portrayed going through opaque and complicated processes to sow chaos, Julian Oliver and Danja Vasiliev are happy to turn that picture around, proposing a simple and apparently harmless device that hijacks the general perception of current news events. This is thanks to the bottlenecking of current media distribution, which is now based on a select group of outlets, a strange symptom of modernity that the authors wish to highlight.



Through a serendipitous set of hacks and détournements of off-the-shelf wireless technology, the finished incarnation of **Newstweek** — a simple pluggable device, no more menacing than an electric air-freshener — won't give out clues as to its real role as a man-in-the-middle filtering the news you read.

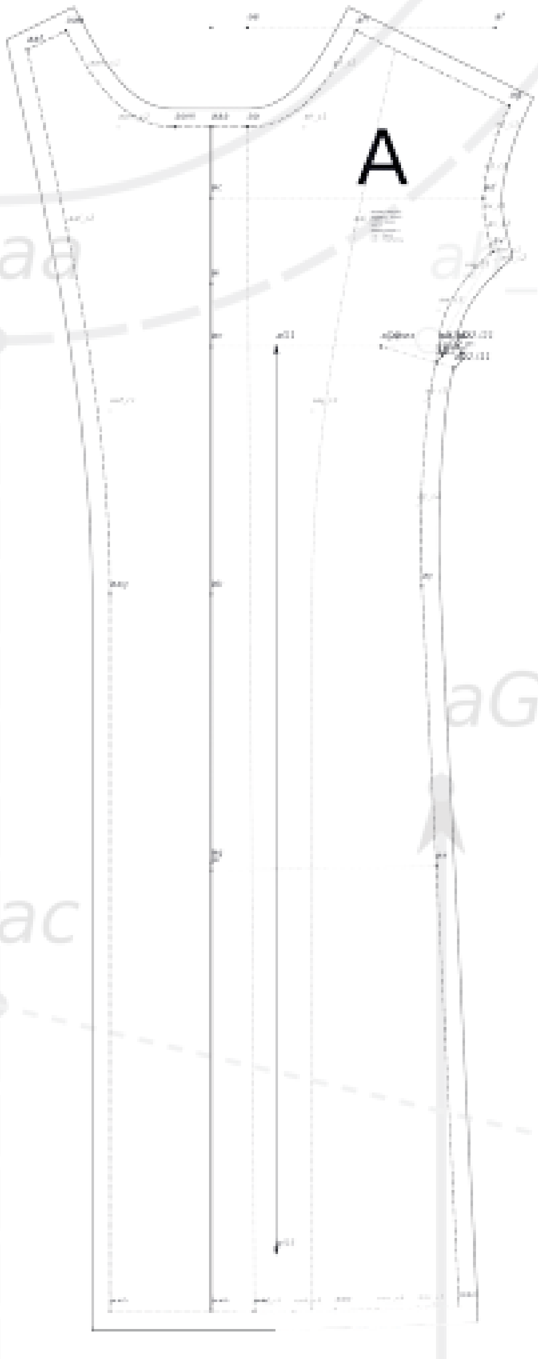
A thorough and clear guide is provided by the authors at http://newstweek.com/howto, with full instructions on how to build your own news-bending device, peppered with hints and clues on how to better understand wireless sniffing and networking in general.



Hacking clothing: An interview with Susan Spencer

NATALIE MACIW

Tau Meta Tau Physica offers a hyper-localization of clothing. The program helps patternmakers generatively make clothes for individuals, based on individual specifications. Natalie Maciw interviews its creator, Susan Spencer.



Natalie Maciw: *How would Tau Meta Tau Physica work for its users?*

Susan Spencer: The TauMeta program is still under development, it is not a finished tool. The user interface will allow patternmakers to select from libraries of patterns, create new, or modify existing patterns, and generate these patterns with a selected client's body measurements or other specifications. There are new tools being developed that focus on 3D methods, but I believe that this sacrifices control of the pattern. 3D methods are best used to initially develop an idea, and at the end of the process check the results of a pattern and present the final pattern in spectacular ways, but letting the software determine details during actual pattern development means that the designer or patternmaker is giving up control and the results may not be what was desired. TauMeta will utilize the best aspects of 2D and 3D methods to develop patterns.

What were your reasons for initiating the Tau Meta Tau Physica project?

There are several excellent patternmaking software packages available, but the ones I could afford did not appear to allow the user to go deep enough into patternmaking to meet my creative needs. There isn't any existing open source patternmaking software that I'm aware of, so I began to create my own. I decided to rethink how patterns were generated so that I could have complete control of the output. I wanted to create patterns in an open data format to enable the patterns to be usable by anyone. Craftsmen, artists and manufacturers should be able to interact without needing to match up the software packages they use.

I am a proponent of the open source and open data movements for programming, government, fashion and just about everything. In our technology-driven world, we need access to tools which are modifiable and customizable to leverage our talents in performing our work and creating our art. If all tools are produced, designed and patented by a few large corporations, and user customization is prohibited, then specific needs of the market may get overlooked. The marketplace of goods and services produced by those tools may become unrepresentative of the products which could be produced by the talents of the design community.

Regarding the fashion industry, it makes sense to have open source tools to design and create fashion-based products. The velocity of the evolution of fashion product is based upon the re-use of existing ideas, especially ideas taken from the street. Creative tools should allow a designer, patternmaker or artist to express their ideas, customize their product, store their work, and allow collaboration and communication on their terms. If the creative person's tools are defined by the needs of entities that are not similar to them, then the tool doesn't serve that person well. Open source software typically invites participation by the users and encourages the development of communities to define new requirements for the software. The users become the producers of the tools. Through this process the behavior and voice of open source communities can indicate the direction and determine the future of the industry that the software serves. Participation in an open source community lends authority to the individual and provides a channel to influence the marketplace. Fashion is a culture where the contribution of the individual is valued. Tools and communities that contribute to this culture increase the relevance of fashion and the vitality of the industry. It makes sense to have flexible and accessible open source tools for the fashion industry.

Is the program primarily intended to be a business or facilitate businesses, or is it designed for hobbyists?

The TauMeta software is intended to be free as in “free access to modify the code” and free as in “free beer.” zIt is intended to give patternmakers complete and accurate control of every aspect of the pattern. This granularity should enable working with extremely creative forms and also provide easy generation of patterns with excellent fit regardless of whether the pattern is unusual or conventional. The focus is to assist patternmakers with artistic creations.

Currently the patterns are home-sewist type patterns with seam allowances and grainlines. But there will be a set of options to choose which type of pattern to generate. Hobbyists will hopefully be an important user base. Someone who doesn't make patterns professionally should be able to understand it and use its full range of functions. It would be nice if TauMeta could be modified to be used by other communities outside of the fashion industry. Perhaps woodworking or sheet metal work. I've played with generating a pattern for the Trajan font's letter A as described in David Lance Goines' book A Constructed Roman Alphabet, *(http://goines.net/acra_book/acra_content/files)*.

Do you think it is important to introduce open source tools to the fashion industry? Could such tools create change in the industry? Who stands to benefit? Does anyone stand to lose out?

Tools define the workflow in any industry. And they can greatly affect profits. Proprietary patternmaking tools have been around for a long time and reflect common needs of the commercial garment industry. The business model, workflow and assumptions built into proprietary software may not be appropriate for all businesses or applications.

The garment industry is huge. The beneficiaries of using open source tools in the fashion industry will primarily be the independent designer or patternmaker, the small or regional businesses, the businesses for which proprietary software has been a bad or marginal fit.

Here are a few examples of how open source tools could benefit the fashion industry: Design tools can be written for improved collaboration between design teams, and further modified to meet the needs of specific projects between teams, schools, print or web magazines, etc. They could be written to easily input fabric characteristics, colour, and other building blocks of fashion, so that what someone sees on the street today they could add to their library of design bits and pieces tonight. The tools could be written to assist in determining where the strengths of a design or manufacturing team lies, and to help restructure to take advantage of these strengths. The point is that these tools should be created for designers or patternmakers to develop products their own way. The essence of open source software is that it can be customized and modified for the needs of the users. The open source community allows everyone to share their findings and developments, and get ideas from others.

I don't see anyone losing as a result of introducing open source tools into the fashion industry. The current propriet-

ary software manufacturers will always be around. I don't see open source fashion tools displacing the big players, I see them enabling new players into the marketplace and creating new communities that will put new ideas into the fashion world.

Pattern cutting seems to suit open source in the sense that it can be made digital—do you see any way of implementing open source principals into the more physical aspects of the fashion industry? How?

Open source software is hackable by definition. Hacking into it to create your own tools and make adjustments helps you do what you do best, leverages your talents, lets you develop your own workflow. Fashion creativity, which includes unique fit and ease, could be extended past the design, patternmaking and sample phases into the manufacturing and distribution phase. Once you embrace doing things for yourself, and understand how something works on your own terms, and then change it to fit your needs, you own the process and that field of endeavor. It's an empowering feeling, and it can make you see the world with different eyes. After that, you may want to change your entire approach to fashion. Anything could change.

There are physical and cultural characteristics of every societal group around the world. Having access to the most detailed level of pattern design enables changes to be made to patterns to accommodate these characteristics.

These differences can also affect workflow and manufacturing processes. If software works only one way, then it can't be adjusted to take advantage of the characteristics of the people who use it or are served by it. Open source software is customizable so that individual or group characteristics and local competencies can be reflected in the business. If a local community is known for specific fabrics or sewing techniques then these should be explicitly incorporated into the design tools.

Is open source a growing trend in the fashion industry, or is your project a one-off?

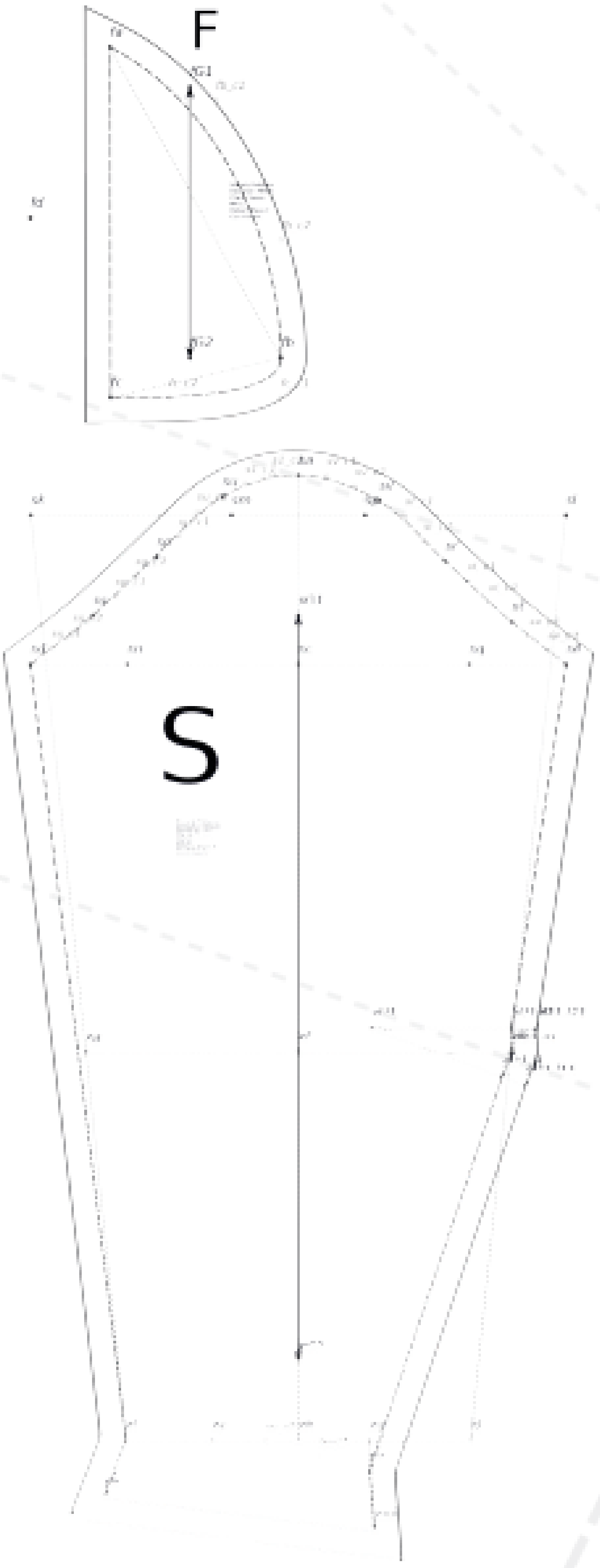
Open source has always been a silent partner in the fashion industry. There are some who work towards copyright restriction, but I must respectfully disagree with this approach. I believe that protection for a product is actually a restriction on the industry. For example, Diane von Furstenburg combined the previously existing wrap-around dress and the simplified dress silhouette of the late 1960s and early 1970s with the amazing new knit textiles of the mid 1970s to achieve her gloriously fashion-breaking wrap dress. She did not develop this dress in a bubble, nor did she have to deal with another designer owning the copyright for wrap-around dresses, or for usage of a particular color or pattern in combination with the dress, or for a particular slope on the neckline, nor for having a particular sleeve and cuff and collar, nor for hemming and lining and seam finishing techniques, nor for any other design feature that existed in other garments before they existed in her dress. She had open source access to the design elements which her artistic talent combined into a fashion hit and personal global career success. There is nothing in her garment that did not previously exist in other garments. Yet, the world knows the name Diane von Furstenberg. This is the lesson of open source in the fashion industry. Fashion is a constant recycling of existing design elements combined with traditional and emergent textiles to generate fresh new product and art.

In the United States, a copyright is automatically applied to garment patterns without the patternmaker having to apply for the copyright. The copyright does not extend to the design or design elements the pattern was based upon. Automatic copyrights on patterns don't appear to have affected the fashion industry very much!

The TauMeta project in the fashion industry is a reflection of the need for open source tools in every industry. As our dependence on technology grows, so does our need to own our own tools. As I see it, the TauMeta project is part of the greater trend and is not a stand-alone development.

What effect do you hope the software will have on the fashion industry once it is complete?

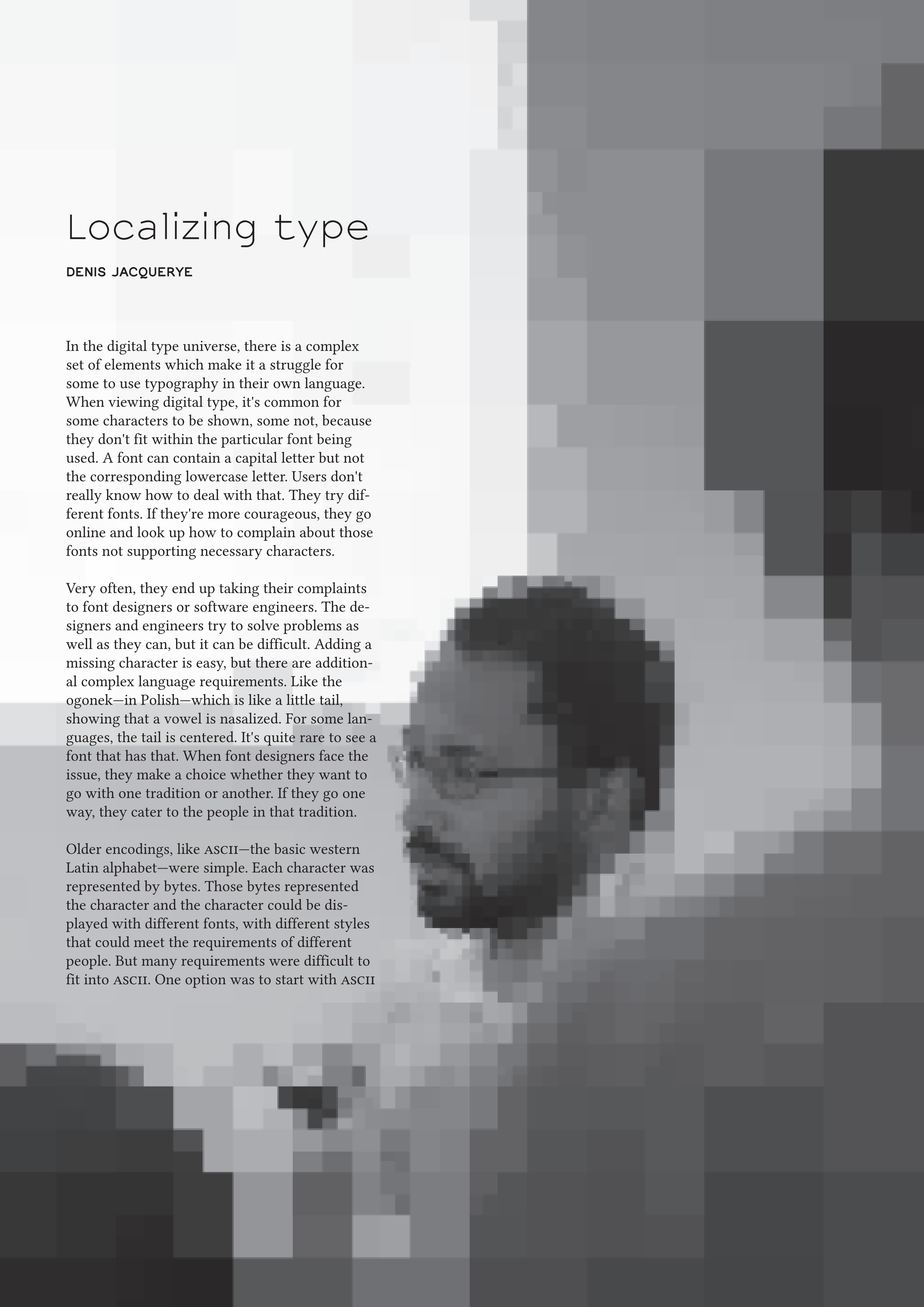
I hope that it enables the entry of new fashion designers into the industry, that it encourages fashion designers to enjoy patternmaking and patternmakers to enjoy designing fashion. I hope it helps local and regional fashion designers and garment manufacturing businesses to bloom as a result of designing to reflect regional identities. I hope it helps to provide garments to previously ignored or under-served markets and to help remove a source of depression for people who are frustrated by the garment choices available to them and who have a negative relationship with clothing and fashion.



On this point, I must express the opinion that most people don't wear fashion. They wear clothing. Many dislike the clothing they wear because it doesn't fit and it looks awful. They may even reject the fashion industry as being useless and self-serving and pointless. Yet everyone on the planet wears clothing!

I hope the TauMeta project can help in its small way to improve access to clothes that fit, clothes that don't engender self-loathing and that go the extra distance to assist in building a healthy outlook and lifestyle. The average person deserves to have a better relationship with what they put on their body every day. There may be some who say that what we wear is not that important. I would say to these people that our clothing shouldn't be the most important aspect of our lives and our societies, but neither should it be the most consistently humiliating. Access to hackable design patterns which can be generated with custom measurements could help make a necessary change in our culture.

http://taumeta.org



Localizing type

DENIS JACQUERYE

In the digital type universe, there is a complex set of elements which make it a struggle for some to use typography in their own language. When viewing digital type, it's common for some characters to be shown, some not, because they don't fit within the particular font being used. A font can contain a capital letter but not the corresponding lowercase letter. Users don't really know how to deal with that. They try different fonts. If they're more courageous, they go online and look up how to complain about those fonts not supporting necessary characters.

Very often, they end up taking their complaints to font designers or software engineers. The designers and engineers try to solve problems as well as they can, but it can be difficult. Adding a missing character is easy, but there are additional complex language requirements. Like the ogonek—in Polish—which is like a little tail, showing that a vowel is nasalized. For some languages, the tail is centered. It's quite rare to see a font that has that. When font designers face the issue, they make a choice whether they want to go with one tradition or another. If they go one way, they cater to the people in that tradition.

Older encodings, like ASCII—the basic western Latin alphabet—were simple. Each character was represented by bytes. Those bytes represented the character and the character could be displayed with different fonts, with different styles that could meet the requirements of different people. But many requirements were difficult to fit into ASCII. One option was to start with ASCII and add specific requirements. That choice resulted in a collection of different standards, responding to different needs. One byte representation could have different meanings and two meanings could be displayed differently in different fonts, often resulting in rendering which looked like gibberish.

Enter Unicode

In the late '80s, people started thinking about compatibility problems in type. In the '90s, with Unicode, they started really working on it. Companies got together and worked on one single unifying standard that would be compatible with all of the previous standards. In Unicode, there's a universal code point to identify a character. That character can be displayed with different glyphs depending on the font or style selected.

Inclusion criteria have changed a bit in the past. Initially, in Unicode, there was basic Latin. And then they started adding all the special characters that were used in, for example, the International Phonetic Alphabet. Initially, they added the characters already used in other encodings. Then they added all the other accented characters they new about, even the ones which weren't already present in other encodings. Then Latin letters with marks, used in transcription. At some point, they realized that this list of accented characters would continue growing and considered that there must be a smarter way to do things. They figured they could use parts of characters, broken apart. A base letter, with marks added. Breaking things apart would would save them from having thousands of accented characters. They could have pretty much any possible accented character, using parts to represent it.

Most keyboards are based on the old encodings, with accented characters as single characters. For a sequence of several characters, like those in the new Unicode style, either more typing is necessary or a special keyboard layout allowing one key to be mapped to several characters is needed. That's technically feasible, but it's a slow process. Developers might add very common combinations to the keyboard layout or to applications, but other people have different needs that are less common. It takes the same effort again to make those sequences available.

Most of the necessary documentation is actually available in a book published by Unicode with every new version. That book has a few chapters that describe how Unicode works and how characters should work together, what properties they have, all the differences between scripts that are relevant. They also have special cases, trying to cater to those needs that weren't met with the proposals that were rejected.

Extending Unicode

Unfortunately, sometimes there's just no code point for the needed character. That could be because the character wasn't in any existing standard, no one has ever needed it before, or the people who needed it simply used old printers and metal type, never switching to digital. In cases where a character doesn't exist in Unicode, it comes down to dealing with the Unicode organization itself. They have a few ways to communicate. There's a public mailing list and also a forum. In those venues, it's possible to ask questions about characters. It might be that they do exist and are difficult to find. Most of the time, finding characters can be a problem because Unicode is organized with a very restrictive set of rules. In most applications, characters are just ordered in the way they're ordered within Unicode, meaning the code point order. So the capital “A” is 41. “B” is 42, etc. Because Unicode is expanding organically, work is done on one script and then another, and then they come back to the previous script and add things, which may not be in a logical or practical order.

If a character actually isn't available in Unicode and you want it added, there has to be a formal proposal. Unicode provides a template with questions. Proof needs to be provided that the proposed character is actually used.
The final element in the proposal is an actual font with the character, so that they can use that in their documentation. Basically, the inclusion process is quite difficult to navigate.

Designing with Unicode

The documentation of Unicode itself is not prescriptive, meaning that the shapes of the glyphs are not set in stone. There's room for styling. Unicode just has one shape, and then it's the font designer's choice to have different ones. Unicode is also not about glyphs: It's really about how information is represented, not how it's displayed.

One of the ways to implement all of those features is with True Type/Open Type. It's very technical and can be slow to update, so if there's a mistake in the actual specification of Open Type, it takes a while before it's corrected and before that correction shows up in any applications. A further issue is that it has its own language code system. Some identified languages just can't be identified in Open Type. One of the features in Open Type is that it's possible to specify: “If I'm using Polish, I want this shape and if I'm using Navajo, I want this shape.” That's really cool, because it allows for a single font that's used by Polish speakers and Navajo speakers and they don't have to worry about changing fonts as long as they label what they're doing according to the language they're using. But it becomes a problem for languages that don't have language codes in Open Type. The option is closed to people using those languages.

Most font designers still work with the old encoding mindset where one character is equal to one letter. Some think that following the Unicode character charts is good enough. It's a hard change to make, because there are very few connections between the Unicode world and the people who work on Open Type libraries, how Open Type is handled, the desires of font designers and not least of all the actual needs of the users.

RESOURCES/GLOSSARY

Blender: A powerful 3D animation application for GNU/Linux, Mac OS X and Microsoft Windows.

Bug: A technical error in either software or hardware. Apocryphally believed to have been popularized by insects causing faults in early analog computers.

cms: Stands for Content Management System. Software installed on a server in order to provide a simple framework for editing web pages. WordPress and Drupal are examples of content management systems.

Command line: A text-only user interface, allowing users to input commands without the intervention of cursors and graphical user interfaces.

Creative coding: Programming not for explicitly functional purposes, but for artistic or creative expression.

Creative Commons: A suite of licenses designed to allow creators and users of works flexibility beyond that offered in traditional copyright.

Domain extension/top level domain: A component in every domain name. Common TLDs such as ".com" or ".net" are available to anyone, while regional TLDs such as ".ca" or ".pt" may only be available to residents of the countries they represent.

f/loss: Stands for Free/Libre Open Source Software. Software which has a viewable, modifiable source. It can be modified and redistributed.

Firefox os: An operating system intended for use on mobile phones. Developed by Mozilla.

FontForge: A F/LOSS font editor for GNU/Linux, Mac OS X and Microsoft Windows.

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

Free software: A term describing software which is made available under licenses permitting users to not only run it, but to examine its code, redistribute it and modify it.

Front end/back end: A distinction in web development between code which runs on the computer of the person viewing a given site (front end) and on the server which hosts the site (back end).

gimp: A raster based image editor for GNU/Linux, Mac OS X and Microsoft Windows.

Glitch art: A form of art which attempts to produce or leverage bugs or mistakes in software and hardware.

gnu/Linux: A group of operating systems which are built on the Linux kernel and components from the GNU project, among others, which are widely distributed and freely modifiable.

Inkscape: A vector graphics editor for GNU/Linux, Mac OS X and Microsoft Windows.

Java: A programming language and platform developed by Sun Microsystems, intended for nearly universal compatibility with a variety of devices.

JavaScript: A scripting language commonly used on websites.

jQuery: A popular library used to write and integrate JavaScript efficiently.

Krita: A drawing application supporting both vector and raster images. Available for GNU/Linux, FreeBSD and Microsoft Windows.

libgeos/Geometry Engine Open Source (geos): A library for rendering geometry. Available for GNU/Linux and Windows.

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

LibreOffice: A F/LOSS office suite incorporating standard productivity tools such as a word processor, spreadsheets and slideshow builder, among others. Available for GNU/Linux, Mac OS X and Windows.

Meteor: A JavaScript-based platform for the development of web applications.

Mozilla: A non-profit organization best known as the developer of the Firefox web browser.

Open data: A concept (and associated movement) promoting the public availability and sharing of data. Often associated with efforts to make government data public.

Open hardware: Hardware which follows the same principles as F/LOSS, including publicly available, freely licensed schematics.

Open source: See Free/Libre Open Source Software

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

OpenStreetMap: A permissively

licensed world map project developed by a community of contributors.

OpenType: A vector-based font format intended to offer more advanced features than its predecessor, TrueType.

Open web: A concept based around the combination of open standards and open licenses. Follows the ideal that development on the web should follow best practices of accessible and modifiable code and content.

Popcorn.js: An interactive web framework which allows increased interactivity with video content.

Processing: A programming language and development environment predominantly used for visually-oriented or media-rich projects. Available for GNU/Linux, Mac OS X and Microsoft Windows.

Programming language: An artificial language with a restricted syntax, used as an intermediary between computers and human programmers.

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.

Public domain: The legal status of a creative work for which the copyright (or other rights restriction) has expired. A work in the public domain can be used by anyone, for any purpose, without restriction. Licenses such as the Creative Commons CC0 license emulate public domain.

Pure Data: A visual programming environment designed for the production of interactive multimedia and audio works. Available for GNU/Linux, Max OS X, Microsoft Windows, iOS and Android.

rgba: A colour space commonly used on digital displays.

Scalable Vector Graphics (svg): A standard for vector graphics, developed by the W3C.

Scribus: A desktop publishing application for GNU/Linux, Mac OS X and Windows.

Server: A computer hosting data which is accessed remotely.

TrueType: A common vector-based font format.

Unicode: A standard used for the encoding of characters. The term is often used to refer to the set of characters defined by the standard.

Version control: A means of managing changes (and allowing reversion) to a commonly held body of work, most often a software project.

w3c: The organization responsible for setting web standards, such as HTML5 and SVG.

LIBRE GRAPHICS MAGAZINE ISSUE 2.2

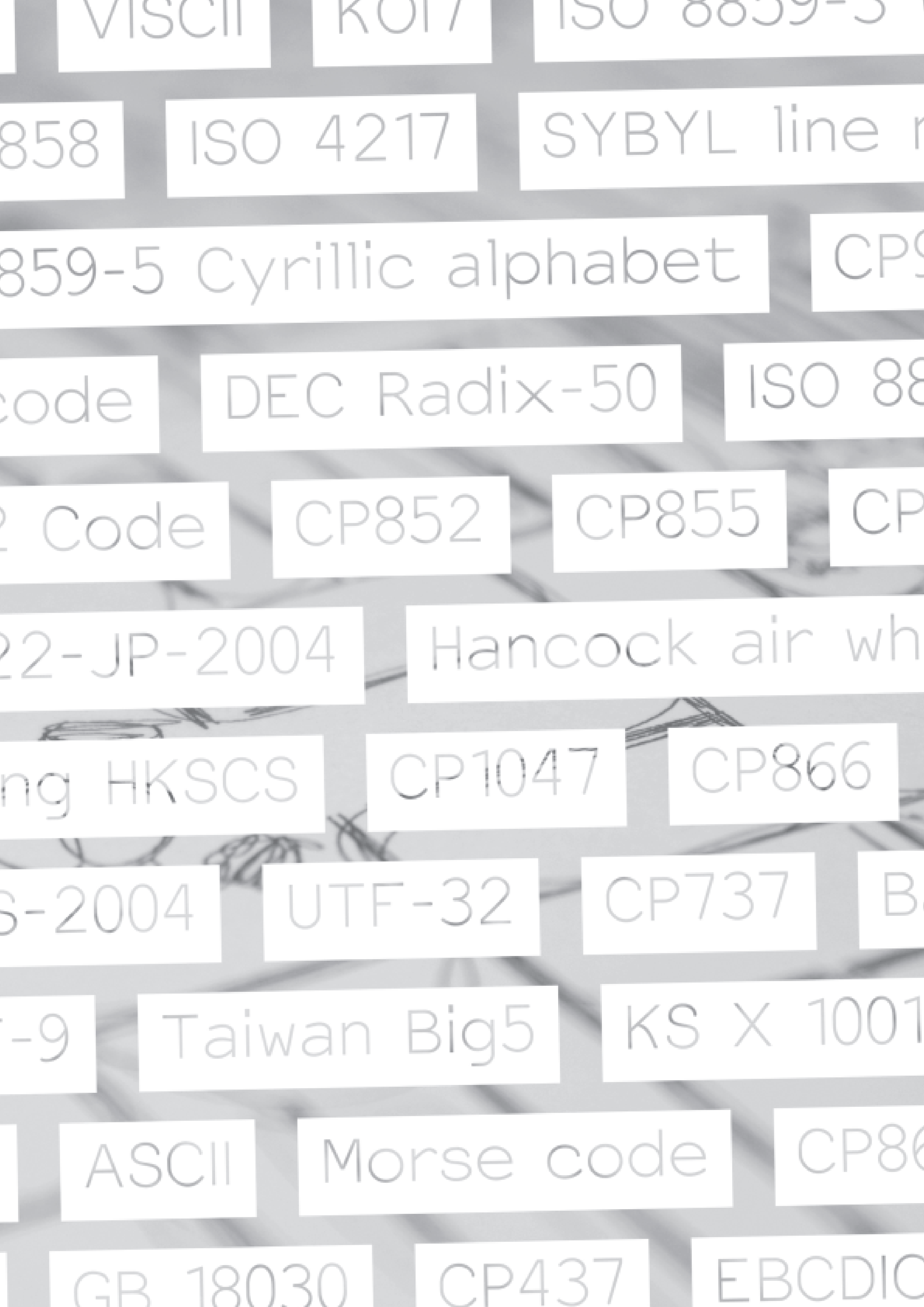
Gendering F/LOSS

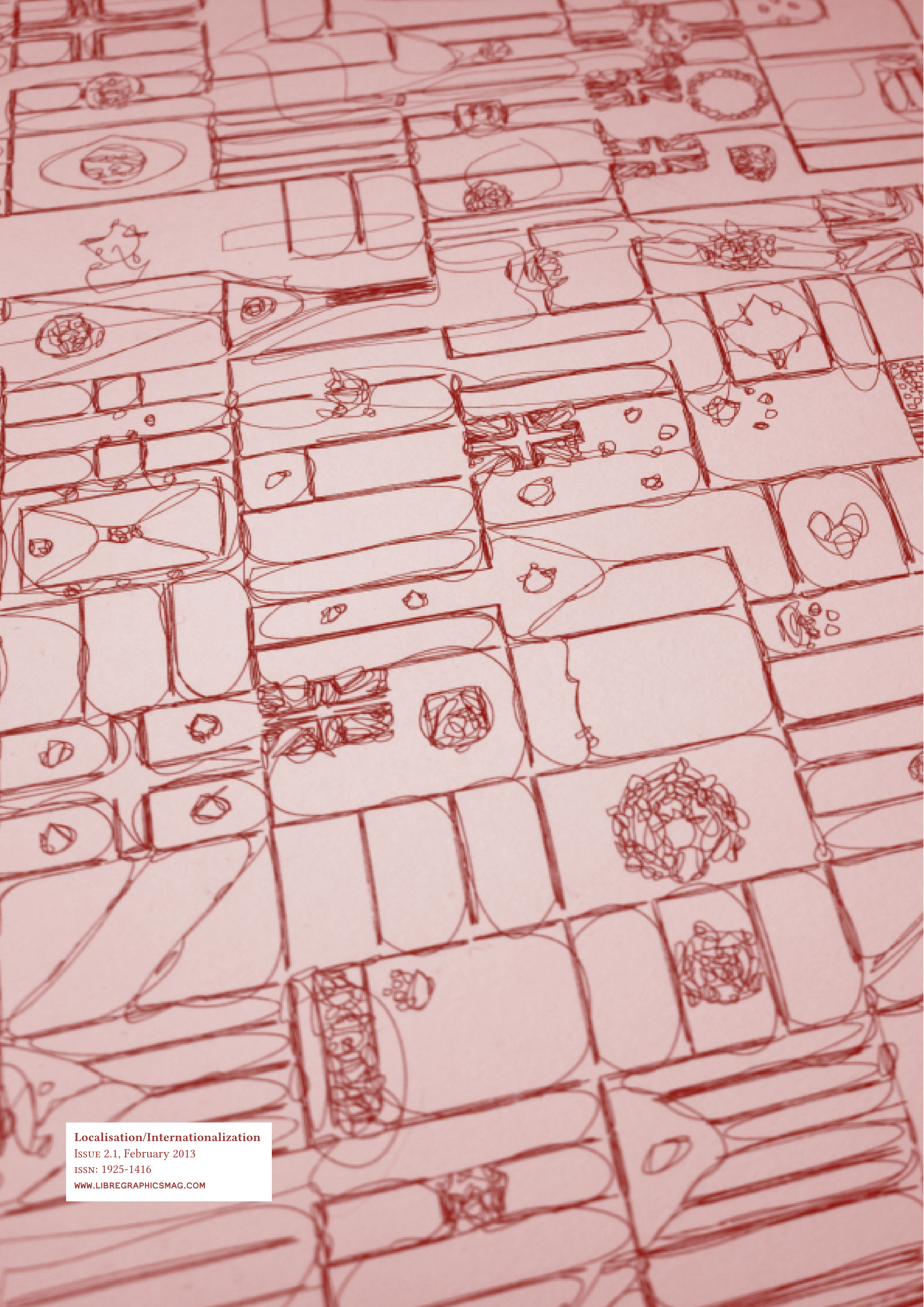
Art and gender have a long history together. From ancient Greek sculptures of ideal men to renaissance nudes, visions of what it is to be a man or woman have been with us for about as long as we've been representing our world. Design, often viewed as the commercial edge of art, has had a hand in shaping our attitudes towards gender. In the service of advertising, graphic design has given us visions of emancipated women smoking cigarettes, underwear-peddling men with spectacular abdominal muscles, and smiling families eating soup together. Design and art show us images of men, women and children: ourselves as we should and shouldn't be.

In the world of Free/Libre Open Source Software, and in the larger world of technology, debate rages over the underrepresentation of women and the frat house attitude occasionally adopted by developers. The conventional family lives of female tech executives are held up as positive examples of progress in the battle for gender equity. Conversely, pop-cultural representations of male developers are evolving, from socially awkward, pocket-protectored nerds to cosmopolitan geek chic. Both images mask the diversity of styles and gender presentations found in the world of F/LOSS and the larger tech ecology.

We're looking for work, both visual and textual, exploring issues of gender and its representation in F/LOSS art and design. Whether it's a tract on the physical appearance of sprites in video games, or a F/LOSS interpretation of a gendered art form, we want to hear about and see it. We invite submissions for articles, showcases, interviews and anything else you might suggest. Proposals for submissions (no need to send us the completed work right away) can be sent to **submissions@libregraphicsmag.com**. The deadline for submissions is March 22, 2013.

Gendering F/LOSS is the second issue in volume two of Libre Graphics magazine. Libre Graphics magazine is a print publication devoted to showcasing and promoting work created with Free/Libre Open Source Software. We accept work about or including artistic practices which integrate Free, Libre and Open software, standards, culture, methods and licenses.


VISCII
KOI7
ISO 8859-3
858
ISO 4217
SYBYL line
859-5 Cyrillic alphabet
CP
code
DEC Radix-50
ISO 88
2 Code
CP852
CP855
CP
22-JP-2004
Hancock air wh
ng HKSCS
CP1047
CP866
S-2004
UTF-32
CP737
B
-9
Taiwan Big5
KS X 1001
ASCII
Morse code
CP86
GB 18030
CP437
EBCDIC



Localisation/Internationalization
Issue 2.1, February 2013
ISSN: 1925-1416
WWW.LIBREGRAPHICSMAG.COM