Web map printing solutions

Marc Monnerat

2018-03-06

Abstracts

This document explores various alternatives for generating suitable document for printing to be used in a web mapping application.

1 Current situation

The swiss federal geoportal map.geo.admin.ch uses a customed developped server-side printing application (service-print based on a forked of MapFish Print v2 and a Python Flask application to allow for multiserver use and multipage PDF document generation. Currently, it runs in a Docker auto-scalling group, with one server at night and two during the day.

1.1 Capabilities

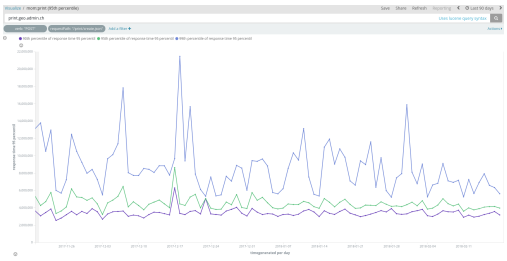
- Printing all 2D layers from map.geo.admin.ch, including import GPX/KML and most WMTS/WMS
- Generating an A4/A3 PDF 1.3 at 150 DPI (not a technical limitation)
- Generating a multipage PDF for Zeitreise (one year per page, about 20 pages)
- Synchronous print for single page, asynchronous for multipage
- 247'198 PDF page generated the last 90 days (500-5'000 per day) and 1'383 multipage print (0 to 50 per day)
- Dockerized application running on a auto-scaling cluster (time-based)
- Using an extend version of the standar print procotol used by GeoMapFish/GeoExt/GeoServer
- Merging legends of complex layers (geology) to the end of the PDF document.

1.2 Shortcomings

- Print only 2D map
- Imported WMS and WMTS layers won't be printed if remote server do not support LV95 projection (EPSG:2056). These layers may be still displayed in the application
- WMTS layers are added as indivual tiles to PDF, for every layers (way too much information if some layer are fully opaque) 232 MB for Zeitreise only, 376MB (25 pages)
- Very limited support for Vector. Luckily, imported vector do not support many styles.
- Symbols size for Raster layer (Vector symbol are adapted to print resolution)
- MapFish v2 is not developped anymore. swisstopo is patching a fork.
- Support for retry and multiserver is sub-optimal.

1.3 Performances

The 95th percentile for all print jobs are between 3.7 and 5.0 s (only time for generation, without download)



https://kibana.bgdi.ch/goto/3566fae450682eea244826f2ec49e477

Printing a standard A4 landscape page at 1:25'000

Generation time (POST to response): 1.14 ± 0.19 s (n=256, every 5 minutes)

Error rate: last month 1'082 errors for 97'988 success (about 1.09%)

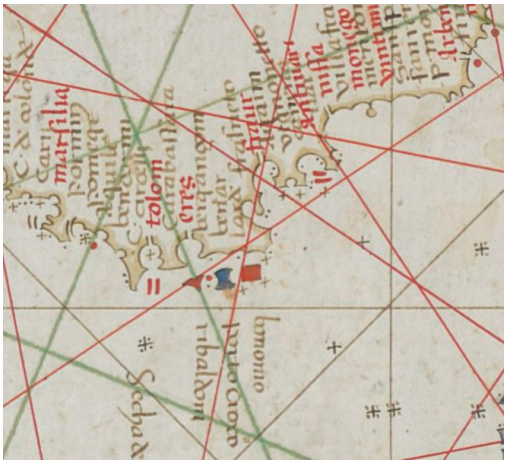
2 What is printing?

In the context on web mapping application, printing is generally seen as as a way to generate a file, like an image or a PDF, suitable to be sent to an office printer. But even in this context, some people may primarly interested in the PDF file, not a paper impression. This is a huge difference.

This is a bit restrictive, as PDF file may be also saved for offline use. The advent of 3D data and application, also brings new possiblities and challenges.

3 What should be printed?

Since its inception, maps have been decorated with more or less useful features.



3.1 Map data

3.1.1 2D

3.1.1.1 Raster

Source are generally satelite/aerial imagery, and scan of legacy maps (historical maps) Served usually as WMS and WMTS

3.1.1.2 Vector

Vector are now a few layer as GeoJSON, and imported GPX and KML

Question: rasterize data to print seems obvious, but it is definitely a loss of information.

- Sync between server tools and

3.1.2 3D

- To convert to 2D or not? If so, rasterize or not?
- Render as 3D (export ?) for use with 3D printer or anything else (virtual world, KML, etc.)

3.1.3 Time (4th dimension)

- Data changing over time (Zeitreise)
- Position changing over time (fly path)

How to render: multipage, movie?

Another challenge: mixing data with different rate of change and/or lacking data.

3.2 Non map data

Non-map data provide useful informations

3.2.1 Logo and corporate identity

Important or not, it gives a professinal look

3.2.2 Disclaimer/copyright

Providing a clear delimitation between data which are from the geoportal and which data are 3rd party. Some remainder of copyright use.

3.2.3 Title, note by user

Provide the user to give its work a title and notes

3.2.4 Legend to the map

Some data are more complexe, and need an explanation. It could be as *simple* as displaying the classification to a full grown geological explanation of the map.

3.2.5 Scale, scalebar, norh arrow

Useful information for the orientation, and when hiking.

3.2.6 QRcode, shortlink

Useful to recreate the map in the application online

3.2.7 Table data (reporting)

Some infomarmation be easier to display as table. Not used in map.geo.admin.ch, but proposed by some printing application

3.2.8 File metadata (if applicable)

Metadata are something useful for search engine if the main goal is to store the PDF.

3.2.9 Grid

Various geographical grids, for orientation and use with GPS.

3.2.10 Information on elements displayed

Could be the location of a search term in the simpliest form or additional information on a highlighted object (tooltips,etc.)

4 Tools, building bricks

4.1 Browser

4.1.1 @media print

4.1.2 Rasterizing

HTML5, *toBlob()*, *toDataURL()*

4.1.3 Export canvas to a vector format

As SVG or PDF (what libraries?)

4.2 Server-side rendering of GIS data

Either vector or raster.

4.2.1 Rasterizing of canvas

Challenges: * Pretty slow * How do you know the map is fully loaded and rendered?

4.2.1.1 SlimerJS (Firefox based)

- WebGL
- Not truely headless (?)

4.2.1.2 PhantomJS (WebKit based)

- No WebGL support

4.2.2 Rasterizing/exporting data

4.2.2.1 Mapserver

Cartographical Symbol Construction with MapServer

Mapserver is hower able to render to vector format like SVG and PDF (when using PDFlib)

4.2.2.2 Mapnik

Map Markup Language file) is a YAML or JSON Mapnik XML, Cascadenik MML, Carto MML Mapnik was the original tool to generate OSM tiles for the so-called slippy map. Mapnik is also able to genrate PDF when compiled with Cairo.

It uses Mapnik XML as configuration, also for styles. Cascading Sheets Of Style for Mapnik aka Cascadenik is a preprocessor for Mapnik, using cascading style sheet for map definition.

CartoCSS is a language for map design. It is similar in syntax to CSS, but builds upon it with specific abilities to filter map data and by providing things like variables. It targets the Mapnik renderer and is able to generate Mapnik XML or a JSON variant of Mapnik XML.

It is now deprecated (The end of CartoCSS) by its parent company, Mapbox.

4.2.2.3 Tileserver GL

A Mapbox Style Specification is a document that defines the visual appearance of a map: what data to draw, the order to draw it in, and how to style the data when drawing it. A style document is a JSON object with specific root level and nested properties. This specification defines and describes these properties.

Vector and raster maps with GL styles. Server side rendering by Mapbox GL Native. Map tile server for Mapbox GL JS, Android, iOS, Leaflet, OpenLayers, GIS via WMTS, Tileserver GL

- GDAL/Rasterio
- OWSlib

4.3 Print server

4.3.1 Mapfish print

The purpose of Mapfish Print 3 is to create reports that contain maps (and map related components) within them. The project is a Java based servlet/library/application based on the mature Jasper Reports Library.

4.3.2 Geoserver print

The Geoserver Printing Module allows easy hosting of the Mapfish printing service within a GeoServer instance. The Mapfish printing module provides an HTTP API for printing that is useful within JavaScript mapping applications. User interface components for interacting with the print service are available from the Mapfish and GeoExt projects.

4.4 Integrated print tools

Need more testing...

4.4.1 QGIS print

4.4.2 ArcGis print

ArcGIS Enterprise includes a geoprocessing service called PrintingTools. Web applications invoke the Printing-Tools service and get a printable document in return (see Printing in Web application):

5 Other considerations

5.1 Scale and resolution

Are scale and print quality of the result important?

5.2 WebGL

5.3 Vectors in PDF

Must vector layers rendered as vector or must be rasterized?

5.4 Need for User defined style

With vector data, it is easy to apply user defined style, though defining styles for complexes dataset is a daring undertaking.

5.5 PDF standard

Support of PDF/2 or PDF/A? No, open source library available. See PDFlib. Not possible in the client

5.6 Legacy client

For some legacy client, we must eventually also provide raster tiles (from vector layer)?

5.7 Vector style definition

How are the styles for vector layer defined? Where? And how it is applied?

Currently, all style are defined in Mapserver's Mapfile definition.

5.8 Complexe symbols

Especially complexe labels placement is hard (what to render, at what scale, collision avoidance)

5.9 Mashup

Maps should be mashed up with other sorts of infos (diagram, plot, data tables)

5.10 Movie

Zeitreise, Fly along path, etc.

5.11 Compute power

Externalize compute power on client or not?

5.12 Performance

Synchronous or asynchronous printing

5.13 WYSIWIG

Symbol scaling, which LK to use, grid display, etc.

5.14 Grid

Grid for various projection system

5.15 Projection issue

One selling point of vector tiles and 3D is that 2D in only a special case when pitch is 0 (or 90°). But the 3D world is a WGS1984 only world, which translate in the infamous Equirectangular (or plate carrée)projection. Using the same projection for printing as in the browser (projection of the data, target projection). When using a Webmercator, deformation, scale, with LK

5.16 Rasterize as a data protection tool

Rasterizing vector data was also a way to *protect* the more valuable original vector datasets (_e.g._WMS).

5.17 Export to other format

Only PDF, os as image

5.18 Use on smartphone and tablet

5.19 Printing API

Provinding an API for 3rd party

5.20 Multiserver use, autoscaling, etc.

5.21 KMZ with local file

5.22 GeoPDF

6 Approaches

6.1 CSS: @media print

@media in MDN web docs

> The @media CSS at-rule associates a set of nested statements, in a CSS block that is delimited by curly braces, with a condition defined by a media query. The @media at-rule may be use

6.1.1 Browser print function

Easiest solution, used by Google Map, Bing Map. WYSIWIG

All it takes is to define a CSS stylesheed for print (for inspiration Paper CSS).

A CSS at-rule @page to define page-specific rules when printing web pages, such as margin per page and page dimensions. Not supported by Safari (all versions).

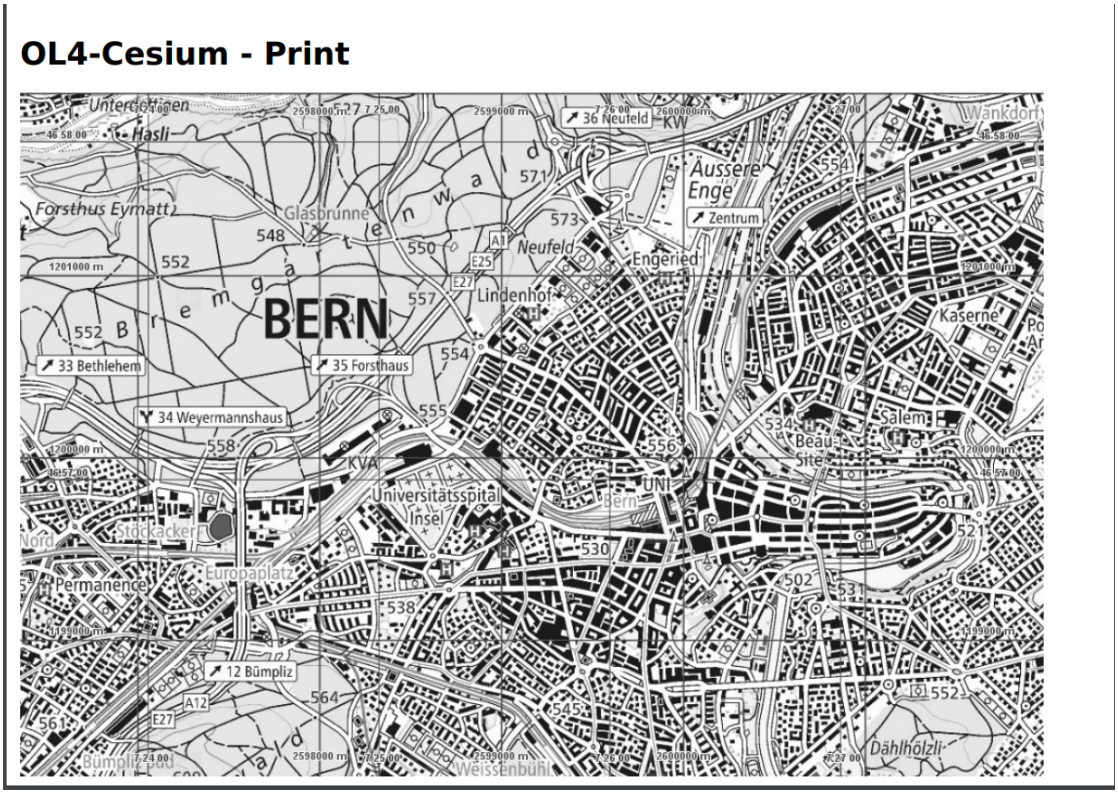
```
page[size="A4"][layout="landscape"]  {
    width: 21cm;
    height: 29.7cm;
}
```

But, as we do not have any influence on the size of the image generated in the browser, we have to make trade off in term of scale, print resolution and image aspect ratio. To print at 150dpi on A4, we need an image of 1750x1250 pixel approximatly, which is OK on a deskop computer, but probably not on a laptop or tablet.

Canvas of 650x450px not fitting an A4 page

OL4-Cesium - Print

Canvas of 1050x750px fitting an A4 page



A bigger display will be cropped

With a map covering the whole screen (`width: 100%; height: 100%`), the challenge is to get an absolute width and height, to be fitted in the print page.

Challenges

- Print to many different paper sizes and orientations
- Make browser recognize size and orientation
- Print quality, on smaller display *e.g.* Laptop with 14” screen

Works well on Chrome, Firefox and Opera (only 2D for the latter), when preview is activated.

Live demo

6.1.2 PDF generation in client

PDF.js (Open Source) and jsPDF (commercial), PSPDFKit (comercial)

6.1.3 Rendering PDF on a server

Generating PDF from XML/HTML and CSS - A tutorial and showcase for CSS Paged Media

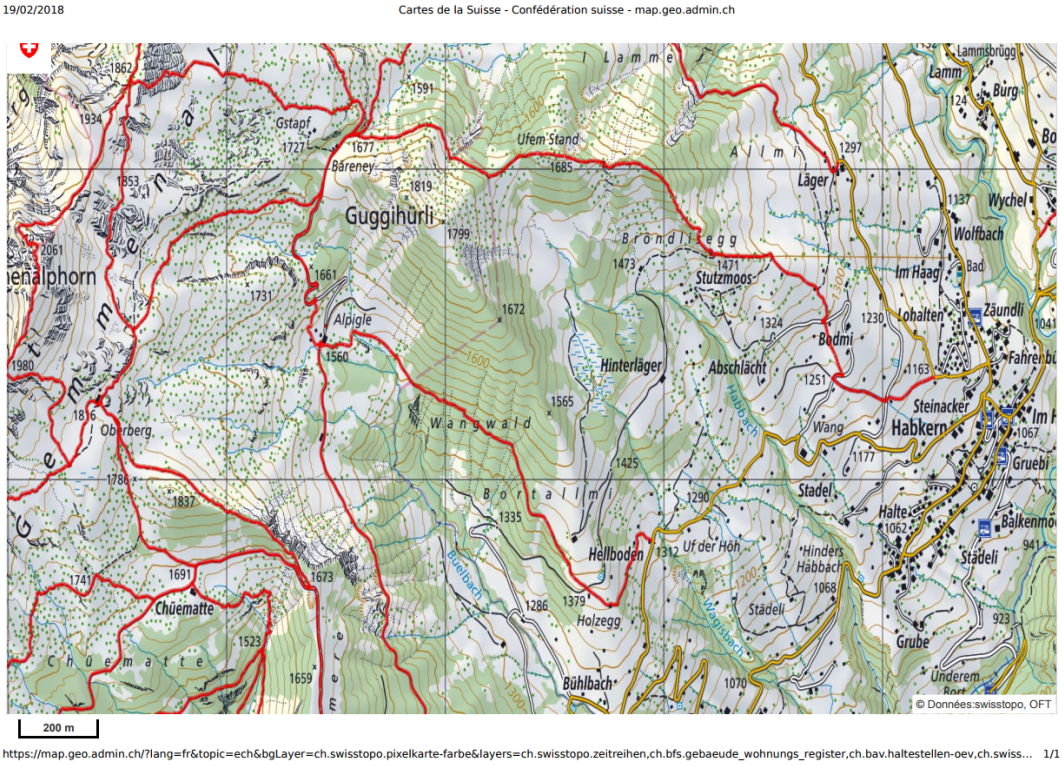
Print CSS rocks

Rendering HTML page to PDF using the full CSS Page media standard with commercial tool like PDFreactor, PrinceXML, Antennahouse CSS Formatter or DocRaport. Some are based on XSL-FO other on Webkit.

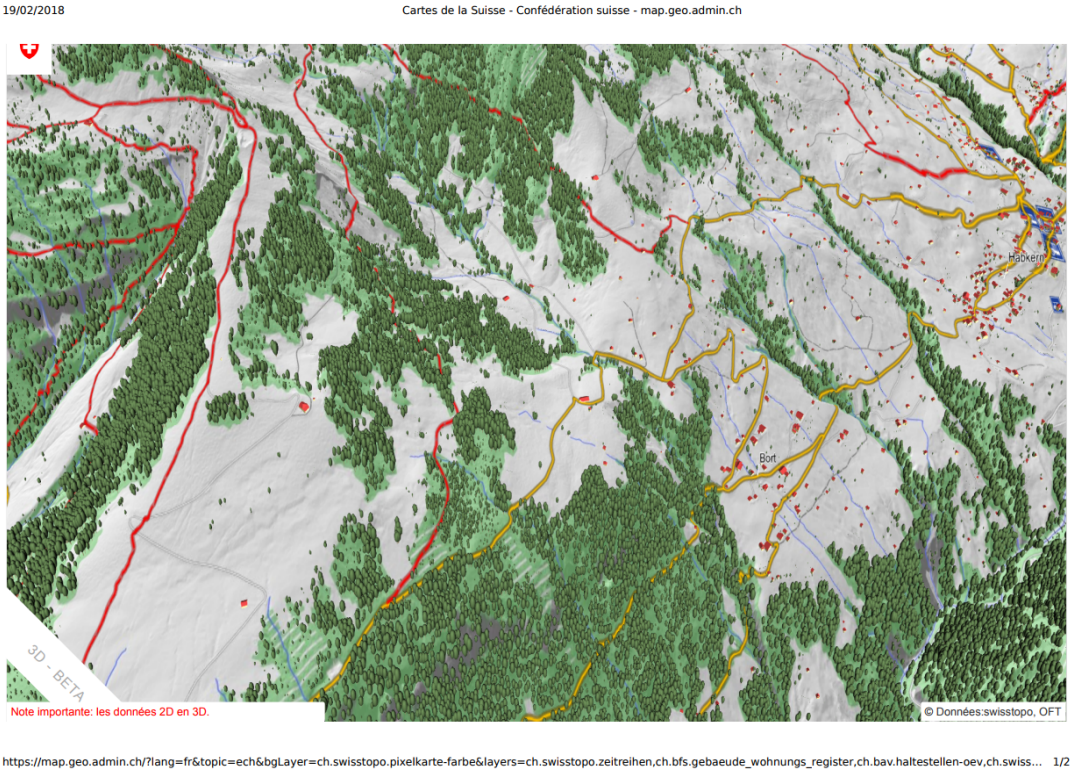
6.2 Export canvas as image

6.2.1 Export canvas as image

Printing in 2D



Printing in 3D



Live demo. This is OL-Cesium, you may toggle 2D/3D.

6.2.2 Export image + popup

Create a simple page in a new popup window, with additional elements, and copy the map as an image HTML template for printing

6.2.3 Better resolution

An approach to gain control over the generated image is to recreate a hidden canvas to generate an image that suit the need. This is the approach of the “High DPI print” for Mapbox (print-maps).

Maybe OK, for simple 2D applications, more difficult for complexe 3D applications with user defined content. What about performance ?

6.2.4 Examples

Mapbox GL using print-maps by Matthew Petroff.

Cesium and swisstopo terrain

OL-Cesium in 2D and 3D mode

OL4-Cesium - Print

Leaflet Easy Print Demo

6.2.5 Discussions

6.2.5.1 Shortlinks

Mabe useful to recreate the application elsewhere, when needed.

6.2.5.2 Browser support

But 3D and WebGL is forcing to use modern browsers.

6.2.5.3 Workload is offloaded to clients

No more server

6.2.5.4 Raster only

Raster only, but smaller images

6.2.5.5 WYSIWIG

Especially in Chrome with the print preview function, the user sees exactly what is going to be printed. A print preview may also be impletemented for other browser.

6.2.5.6 No special code for rendering

If we considere CSS is no code, yes...

6.2.5.7 Resolution, aspect ratio, screen size, pixel density

Basically, your get the canvas at dispostion and try to fit in an A4 page. A small screen means a poors print quality, while a large screen means a decent or good print quality.

Some are trying to get a lager image by recreating a large hidden map canvas. It may work in 2D, but will consume much resource in 2D. Remember, A4 at 150 dpi is 1750x 1250px, and A3 at 150 DPI is 1750x2480px.

Pixel density

A paper map is read at 25cm, screen at 50-70cm 14''and 75 - 105 cm for 20/21" The maximal resolution is about 300 dpi at 25cm, 152 dpi at 50cm and 76 dpi at 100cm. So while a resolution of about 100dpi is acceptable on a desktop, it makes no sense to print above 300dpi for instance.

Table 1: Relation between paper size (mm) and image size (pixels). *dpi* (dot per inch)

Paper	96 dpi	150 dpi	300 dpi
A5 (210x148mm)	793x563	1240x880	2408x1760
A4 (297x210mm)	1112x793	1753x1240	3507x2408
A3 (420x297mm)	1587x1112	2408x1753	4360x3507

6.2.5.7.1 Conserving the scale

The next three images have the same spatial extent (2500 meters wide), but have different sizes. If we want to keep the scale, here 1:25'000, we have to print an image of 100mm, regardless of the size of the image. The result is a low quality for samller image.

Image of 377x188 pixels, printed @96 dpi (100x50mm)

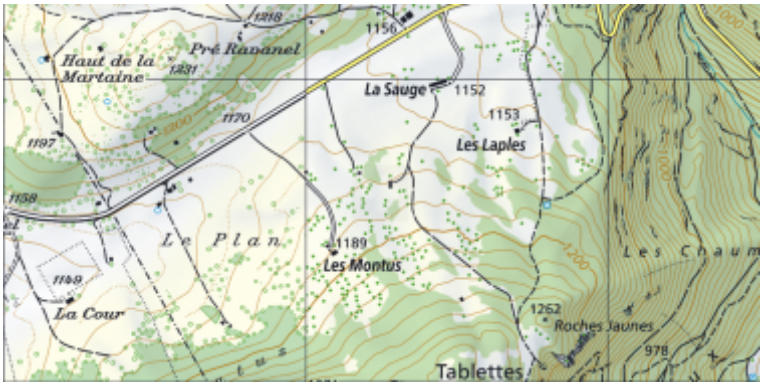


Image of 590x295 pixels, printed @150 dpi (100x50mm)

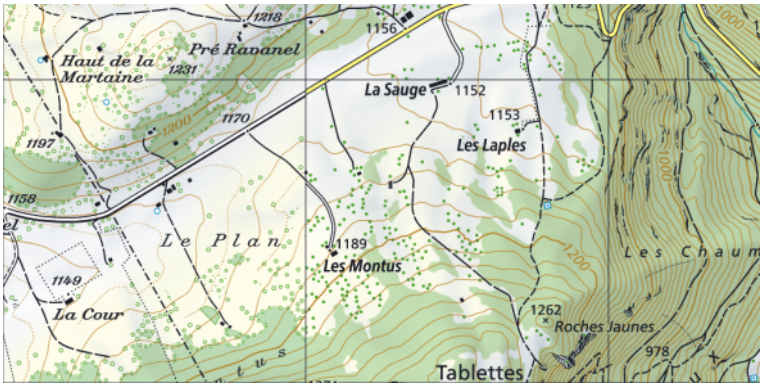
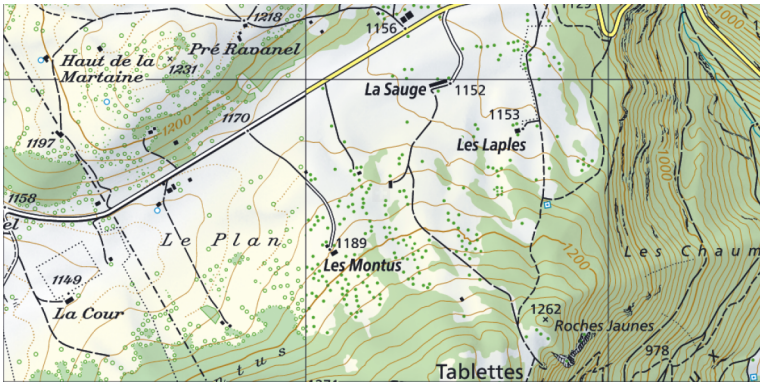


Image of 1181x590 pixels, printed @300 dpi (100x50mm)



6.2.5.7.2 Conserving the print resolution

The next three images have the same spatial extent (2500 meters wide), but have different sizes. In this example, we kee a good print quality (300 dpi), so we have to print . The result is loosing the original map scale of 1:25'000.

Image of 377x188 pixels, printed @300 dpi (32x16 mm), print scale is about 1:78'125

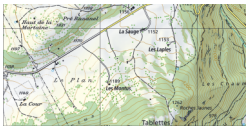


Image of 590x295 pixels, printed @300 dpi dpi (50x25 mm), print scale is 1:50'000

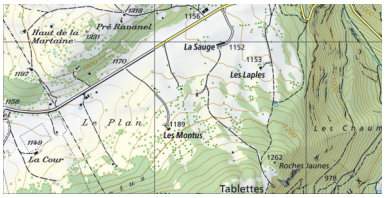
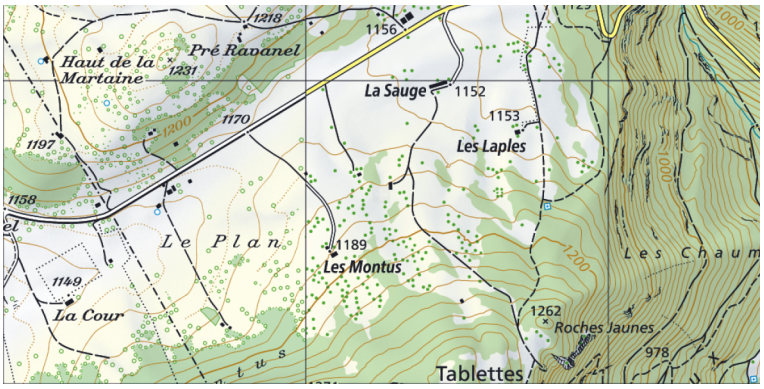


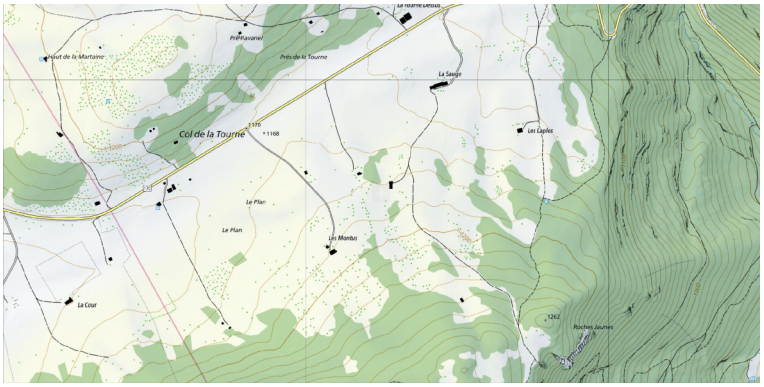
Image of 1181x590 pixels, printed @300 dpi (100x50mm), print scale is 1:25’000



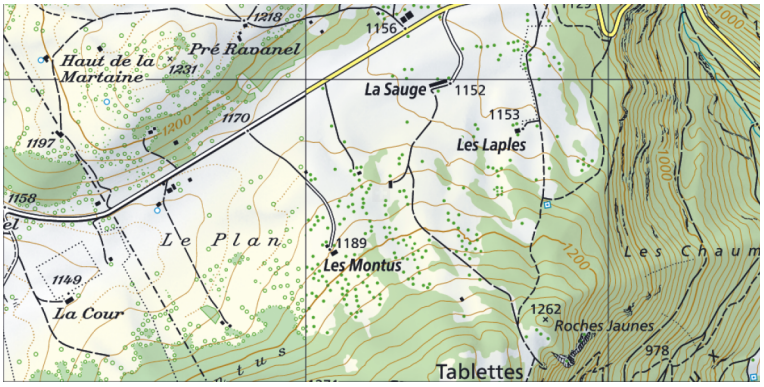
6.2.5.7.3 Effect of style

In this example we use the *Landeskarte/Pixelkarte* because their style is targeted for a very specific scale: 1:10’000, 1:25’000 and 1:50’000. The three following picture are printed at 300dpi for a scale of 1:25’000.

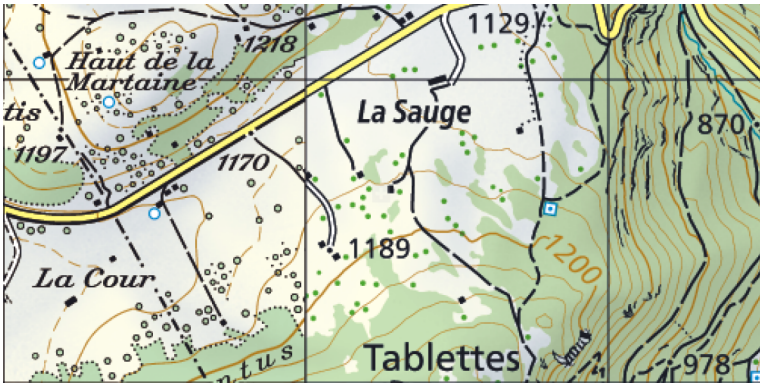
LK10, designed for 1:10’000 Labels and features are very small, hard to read.



PK25, designed for 1:25’000 Labels and features are easily readable, making a joyful impression.



PK50, designed for 1:50'000 Features and labels are big. You may clearly have more information.



6.2.5.8 Templating

6.2.5.9 Performance

Hard to find when the page is fully rendered…

6.2.5.10 Security

Cross-Origin Resource Sharing (CORS) and Content-Security-Policy (CSP)

6.2.6 Conclusion

If the added value is not much more than what a screenshot offers, there not point to provide this functionality

6.3 Server side

6.4 Mixed approach

Some elements are rendered client-side and then sent to the server, or only some operation are done server-side.

6.5 Integrated approach

Tools like QGis and ArcGIS used to configure the layers. Configuration files, including styles, are used in the web application. As both client and server are using the same configuration, printing may be done server-side.

QGis Print Composer

7 Discussion

8 Conclusion

Colophon

This document was created with pandoc, with the following commands:

```
$ pandoc -f markdown --latex-engine=xelatex   --number-sections \
  --variable mainfont="Gentium"   -V fontsize=10pt   "README.md" \
  --toc -o "web-map-printing.pdf"
```