How to make slow maps

Assume rendering is slow

(hint: io, encoding, reprojection, bad code)

Don’t profile

(hint: ab, gdb, dtrace, oprofile)

Use hardware
with < 2 cores
< 1 GB RAM

Use WMS

(unless you've done everything else right or use wms-c)

Allow on-the-fly reprojection (unless proj >= 4.8)

Use slow disks (hint: ssd)

Render multithreaded under high load (danger of contention)

Render large datasets multiprocess

Allow data * cores > memory

(danger of io boundness / swapping)

Place too many labels

Put too much on your map in general

Lack spatial indexes

Forget to vacuum your db

Query more than you render

(hint: use postgres with WHERE)

Use full resolution data at low zooms

Use round linejoins on detailed linework

Write your own server

* parse map for each request or share map objects
* allow map rendering in same process as application

Deploy a python server with threads

(hint: threads=1 processes=N cores)

Do things dynamically when you could cache

Use old versions of libs or operating systems

(watch out: mapnik, sqlite, RHEL, gdal, geos, proj)

Render big things serially, inability to parallelize

Forget to disable the OL scroll wheel