Lighttrends application documentation

Developed by Deneb, Geoinformation solutions, Jurij Stare s.p., Dec 2018, updated Sep 2019

Contents

[Overview 2](#_Toc532900125)

[Back end 2](#_Toc532900126)

[Environment 2](#_Toc532900127)

[Lighttrends services 2](#_Toc532900128)

[Parameters in Web.config 2](#_Toc532900129)

[Keys 2](#_Toc532900130)

[Example of Web.config file 2](#_Toc532900131)

[Services 3](#_Toc532900132)

[Export chart 3](#_Toc532900133)

[Get counters 3](#_Toc532900134)

[Get layer list 4](#_Toc532900135)

[Get raster polygon 4](#_Toc532900136)

[Get statistics 5](#_Toc532900137)

[Update layers 6](#_Toc532900138)

[Database 7](#_Toc532900139)

[Front end 8](#_Toc532900140)

[Dependent JavaScript libraries used 8](#_Toc532900141)

[Development environment 8](#_Toc532900142)

[Application location 9](#_Toc532900143)

[Language settings 9](#_Toc532900144)

[How to use 9](#_Toc532900145)

[License 9](#_Toc532900146)

# Overview

Lighttrends application is a [GIS](https://en.wikipedia.org/wiki/Geographic_information_system) web application that is designed to quickly display information about radiance trends at a specific location. It uses data from two satellite system [DMSP](https://en.wikipedia.org/wiki/Defense_Meteorological_Satellite_Program) and [VIIRS](https://en.wikipedia.org/wiki/Visible_Infrared_Imaging_Radiometer_Suite) operated by [NOAA](https://www.noaa.gov/). New VIIRS layers are added automatically as soon as NOAA makes them available to public.

# Back end

Back end consists of number of services which are queried by the front end client application.

## Environment

Operating system: Linux Ubuntu 18.04 LTS distribution

Raster serving application: Geoserver 2.13.1

Database: PostgreSQL 10.5 with PostGIS 2.4 Extension

Image format converter: Inkscape 0.92.3

ASP.NET handler: Mod\_mono, Mono JIT compiler version 5.12.0.301

## Lighttrends services

Developed in Visual Studio 2015 Community with Framework 4.6.1 in VB. Documentation and comments are embedded in the source code.

## Parameters in Web.config

### Keys

*»passphrase«*

Used for updating layers from NOAA website. More about update procedure on page in the »Update layers« section.

*»noaa«*

Web address of the NOAA website where new satellite data is being added. As of December 2018 it is: “https://www.ngdc.noaa.gov/eog/viirs/download\_dnb\_composites\_iframe.html”

*»pguser«*

Username connecting to postgreSQL database

*»pgpass«*

Password connecting to postgreSQL database

*»tmpdir«*

A directory for writing data during layer update procedure and export chart/data ie. “/var/www/lighttrends/query/tmp/”

### Example of Web.config file

<appSettings>

<!-- for updatelayer procedure. Start. -->

<add key="passphrase" value="secret123" />

<!-- NOAA website for updatelayer procedure. -->

<add key="noaa" value="https://ngdc.noaa.gov/eog/viirs/download\_dnb\_composites\_iframe.html" />

<!-- for updatelayer procedure psql connect credentials -->

<add key="pguser" value="user" />

<add key="pgpass" value="password" />

<!-- for tmp directory for writing data. www-data needs read/write permission to this directory -->

<add key="tmpdir" value="/var/www/lighttrends/query/tmp/" />

</appSettings>

## Services

### Export chart

Endpoint address**:** https://lighttrends.lightpollutionmap.info/query/exportchart.ashx

Exports client generated SVG chart into different formats.

The conversion is done by Inkscape application. It will output a binary file in the specified format.

|  |  |
| --- | --- |
| HTML method | POST |
| Parameters | svg, format |
| Parameter »svg« (required) | File in svg format |
| Parameter »format« (required) | [png|pdf|eps] |

### Get counters

Endpoint address**:** <https://lighttrends.lightpollutionmap.info/query/getcounters.ashx>

Gets statistical counter for data export, point and area analysis. It will aggregate counters depending on the aggregation parameter. It will output results in JSON format.

|  |  |
| --- | --- |
| HTML method | GET |
| Parameters | aggregation |
| Parameter »aggregation« (required) | [day|week|month|year|all] |

Example request: <https://lighttrends.lightpollutionmap.info/query/getcounters.ashx?aggregation=month>

Example response:

{

"counters" : [{

"month" : "2018-09-01",

"point" : 483,

"area" : 280,

"csv" : 45

}, {

"month" : "2018-10-01",

"point" : 357,

"area" : 494,

"csv" : 31

}

]

}

### Get layer list

Endpoint address**:** https://lighttrends.lightpollutionmap.info/query/getlayerlist.ashx

Gets the latest list of radiance layers with the following fields (layername, datestart, dateend, period, satellite, sattypes). It will output results in JSON format.

|  |  |
| --- | --- |
| HTML method | GET |

Example request: <https://lighttrends.lightpollutionmap.info/query/getlayerlist.ashx>

Example response:

{

"dmsp\_c\_f121996" : {

"layername" : "F12\_19960316-19970212\_rad\_v4.avg\_vis",

"datestart" : "1996-03-16",

"dateend" : "1997-02-12",

"period" : "annual",

"satellite" : "DMSP",

"sattypes" : "F12"

},

"viirs\_npp\_201808" : {

"layername" : "SVDNB\_npp\_20180801-20180831",

"datestart" : "2018-08-01",

"dateend" : "2018-08-31",

"period" : "monthly",

"satellite" : "VIIRS",

"sattypes" : "npp"

}

}

### Get raster polygon

Endpoint address**:** https://lighttrends.lightpollutionmap.info/query/getrasterpolygon.ashx

Gets a polygon that undergoes statistical analysis. It will output results in JSON format. Mask parameter will use a VIIRS annual vcm-orm-ntl layer as a mask.

|  |  |
| --- | --- |
| HTML method | POST |
| Parameters | geometry, satellite |
| Parameter »geometry« (required) | WKT closed linestring geometry |
| Parameter »satellite« (required) | [dmsp|viirs] |
| Parameter »mask« (optional) | [none|mask\_{year}00] |

Example request: <https://lighttrends.lightpollutionmap.info/query/getlayerlist.ashx>

POST parameters:

geometry: »LINESTRING(14.6521568277944 45.9921614894469,14.6758460978106 46.0183335059416,14.7007369974654 45.9932944754442,14.6521568277944 45.9921614894469)«

satellite: »viirs«

mask: »mask\_201600«

Example response:

### {

### "geom" : "MULTIPOLYGON(((14.69375059755 45.99791643465,14.69375059755 45.99374976795,14.69791726425 45.99374976795,14.69791726425 45.99791643465,14.69375059755 45.99791643465)),((14.66458393065 46.00208310135,14.66458393065 45.99791643465,14.66041726395 45.99791643465,14.66041726395 45.99374976795,14.66875059735 45.99374976795,14.67291726405 45.99374976795,14.67291726405 46.00208310135,14.66458393065 46.00208310135)),((14.67291726405 46.01874976815,14.67291726405 46.01041643475,14.67708393075 46.01041643475,14.67708393075 46.01874976815,14.67291726405 46.01874976815)))"

### }

### Get statistics

Endpoint address**:** <https://lighttrends.lightpollutionmap.info/query/getstatistics.ashx>

Gets radiance values for requested rasters in either point or area. It will output results in JSON or csv format. If querytype is »area« then response for each raster contains an array with pixel count, sum, average and weighted sum (sum weighted by area) whereas »point« only contains a radiance value. This service is also used for data export. Mask parameter will use a VIIRS annual vcm-orm-ntl layer as a mask. Geometry parameter supports LINESTRING, POLYGON and MULTIPOLYGON geometries in WKT format. LINESTRING gets converted to POLYGON, so it must be a closed LINESTRING i.e. last coordinate must be the same as first one.

|  |  |
| --- | --- |
| HTML method | POST |
| Parameters | Format,querytype,rastercolumns,geometry |
| Parameter »format« (required) | [json|csv|excel] |
| Parameter »querytype« (required) | [point|area] |
| Parameter »rastercolumns« (required) | [rastercolumn; rastercolumn;…] |
| Parameter »geometry« (required) | [lon,lat|LINESTRING(lon,lat ... lon,lat)] |
| Parameter »mask« (optional) | [none|mask\_{year}00] |

Example request: https://lighttrends.lightpollutionmap.info/query/getstatistics.ashx

POST parameters:

format: »json«

querytype: »area«

rastercolumns: »viirs\_npp\_201204;viirs\_npp\_201205;viirs\_npp\_201206;viirs\_npp\_201207;viirs\_npp\_201208;viirs\_npp\_201209;viirs\_npp\_201210;viirs\_npp\_201211;viirs\_npp\_201212;viirs\_npp\_201301«

geometry: »LINESTRING(14.4354128817491 46.0562281773221,14.4218516329717 46.0464292955018,14.4396615008281 46.0411866267192,14.4541668871786 46.0507481658807,14.4354128817491 46.0562281773221)«

Example response:

{

"statistics" : {

"viirs\_npp\_201204" : [null, null, null],

"viirs\_npp\_201205" : [null, null, null],

"viirs\_npp\_201206" : [null, null, null],

"viirs\_npp\_201207" : [null, null, null],

"viirs\_npp\_201208" : [15, 27.01, 1.80067, 18.16],

"viirs\_npp\_201209" : [15, 57.98, 3.86533, 38.98],

"viirs\_npp\_201210" : [15, 50.99, 3.39933, 34.28],

"viirs\_npp\_201211" : [15, 40.51, 2.70067, 27.24],

"viirs\_npp\_201212" : [15, 164.4, 10.96, 110.53],

"viirs\_npp\_201301" : [15, 56.21, 3.74733, 37.79]

},

"result time" : "0.203 seconds"

}

### Update layers

Endpoint address: <https://lighttrends.lightpollutionmap.info/query/updatelayers.ashx>

Used for updating layer database with new layers when they are added by NOAA to their website at this time located at https://ngdc.noaa.gov/eog/viirs/download\_dnb\_composites\_iframe.html . If no parameters are used, the service returns the current state of the update process. If »passphrase« parameter is used then it will perform a manual update (otherwise it is performed each day at 01:00 GMT +1 via cron), if »passphrase« is correct. Parameter »reset« resets the process to »idle« if it gets stuck for some reason.

The update process performs these steps:

1. NOAA website is downloaded and a list is made of all eligable VIIRS layers
2. The table public.lighttrends is checked if its already up to date
3. If step two is false then it proceeds with updating process
4. Downloads .tgz (containing radiance and observation files) files from NOAA website and extracts contents
5. Sets pixels in radiance tiff to nodata (null) where no observations were conducted
6. creates a mosaic from 6 tiles
7. inserts new record into table public.lighttrends containing layer metadata
8. Projects the mosaic into EPSG:3857 projection
9. Performs color reduction to save space
10. Create overview pyramids
11. Publishes the layer in Geoserver

|  |  |
| --- | --- |
| HTML method | GET |
| Parameters | passphrase, reset |
| Parameter »passphrase« (optional) | passphrase (set in Web.config) |
| Parameter »reset« (optional) | [true|false] |

Example request: <https://lighttrends.lightpollutionmap.info/query/updatelayers.ashx>

Example response:

{

"activity" : "idle",

"timestamp" : "2018-10-25 01:01:02"

}

## Database

Lighttrends services use five tables described below.

Tables that contain rasters are public.dmsp and public.viirs. DMSP raster column nomenclature: dmsp\_[u|c]\_[sat. type][year]. u – uncalibrated, c - calibrated DMSP. VIIRS raster column nomenclature: viirs\_[sat. type]\_[year][month].

Table public.lighttrends contain metadata about loaded rasters.

Table public.lighttrends\_counters contains lighttrends usage statistics.

Table public.lighttrends\_update\_status contains the status of the update process.

public.dmsp

CREATE TABLE public.dmsp

(

rid integer NOT NULL DEFAULT nextval('dmsp\_add\_rid\_seq'::regclass),

dmsp\_u\_f101992 raster,

dmsp\_c\_f121996 raster

CONSTRAINT dmsp\_add\_pkey PRIMARY KEY (rid)

);

CREATE INDEX dmsp\_rid\_idx

ON public.dmsp USING btree

(rid)

TABLESPACE pg\_default;

CREATE INDEX dmsp\_st\_convexhull\_idx

ON public.dmsp USING gist

(st\_convexhull(dmsp\_u\_f101992))

TABLESPACE pg\_default;

public.viirs

CREATE TABLE public.viirs

(

rid integer NOT NULL DEFAULT nextval('viirs\_rid\_seq'::regclass),

viirs\_npp\_201204 raster

CONSTRAINT viirs\_pkey PRIMARY KEY (rid),

);

CREATE INDEX viirs\_rid\_idx

ON public.viirs USING btree

(rid)

TABLESPACE pg\_default;

CREATE INDEX viirs\_st\_convexhull\_idx

ON public.viirs USING gist

(st\_convexhull(viirs\_npp\_201204))

TABLESPACE pg\_default;

public.lighttrends

CREATE TABLE public.lighttrends (

"ID" integer NOT NULL DEFAULT nextval('lighttrends\_id\_seq'::regclass),

layername character varying(255),

rastercolumn character varying(255),

datestart date,

dateend date,

period character varying(255),

satellite character varying(255),

sattypes character varying(255),

CONSTRAINT lighttrends\_pkey PRIMARY KEY ("ID")

);

public.lighttrends\_counters

CREATE TABLE public.lighttrends\_counters

(

id integer NOT NULL DEFAULT nextval('lighttrends\_counters\_id\_seq'::regclass),

date date,

point bigint,

csv bigint,

area bigint,

CONSTRAINT lighttrends\_counters\_id\_pkey PRIMARY KEY (id)

);

public.lighttrends\_update\_status

CREATE TABLE public.lighttrends\_update\_status

(

id integer,

status text COLLATE pg\_catalog."default",

lastactive timestamp without time zone

);

# Front end

The application was developed to run on all modern browsers (Chrome, Firefox, Safari…) optimized for desktop display.

## Dependent JavaScript libraries used

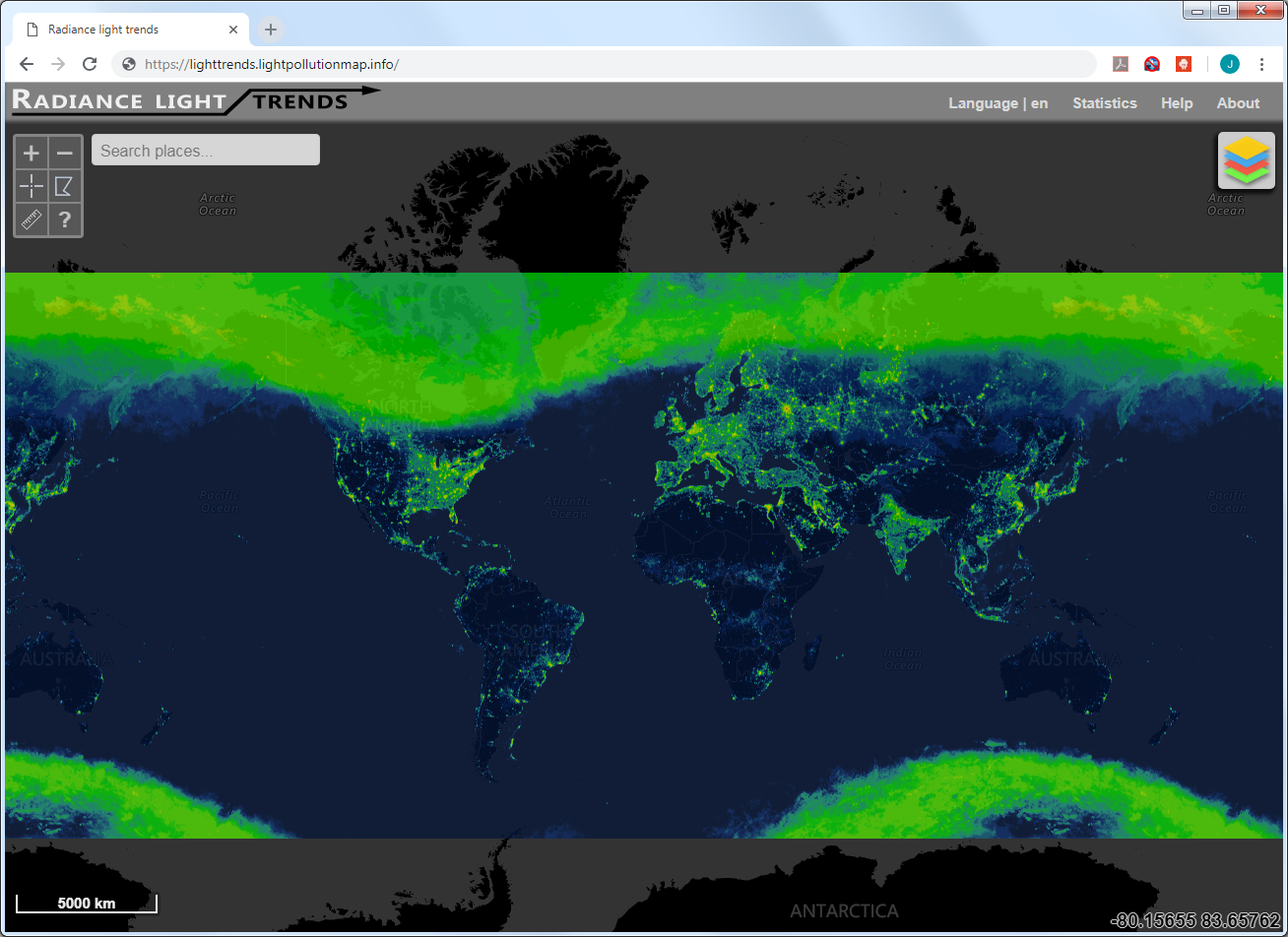
jQuery 3.3.1, jQuery UI 1.12.1, chart.js 2.7.2, chosen.js 1.8.7, jstree 3.3.5, canvas2svg 1.0.19, Open Layers 5.3.0, file-saver 2.0.0-rc.4, regression 2.0.1, hopscotch 0.3.1, datatables 1.10.18

## Development environment

NPM package manager for JavaScript (6.4.1) with webpack (4.8.1) module bundler. Documentation and code comments are embedded in the source code.

## Application location

<https://lighttrends.lightpollutionmap.info>



## Language settings

Application can be translated by adding a new language into the labels.json file. Currently two languages are available English and Slovenian. Setting a default language can be done by changing the value on language global variable at line 40 index.js.

## How to use

Use the »Help« button which will guide you through all functions of the application.

# License

Application is released under the European Union Public Licence (EUPL v1.2)