

Current state of geoprocessing development on the web applications:

NODE.js, R in the server , ArcGIS server and Geoserver



cooperación
española



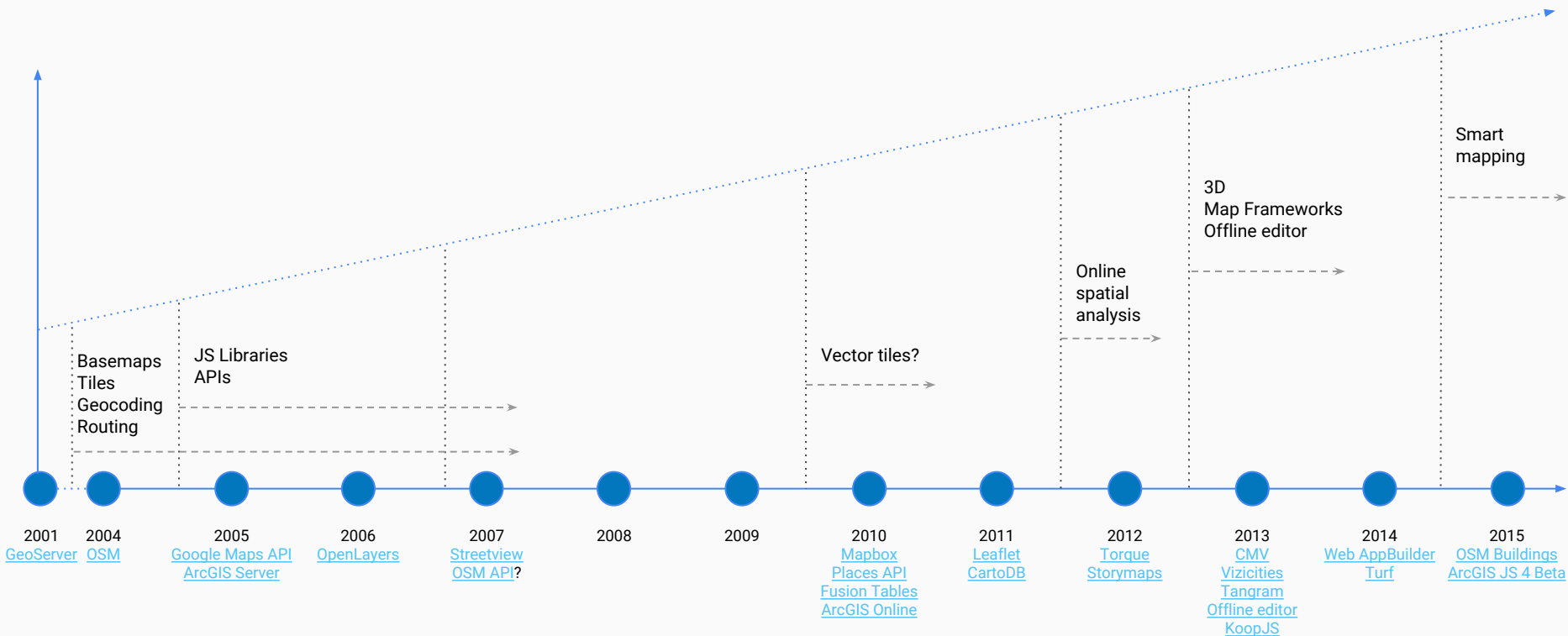
United Nations
Educational, Scientific and
Cultural Organization



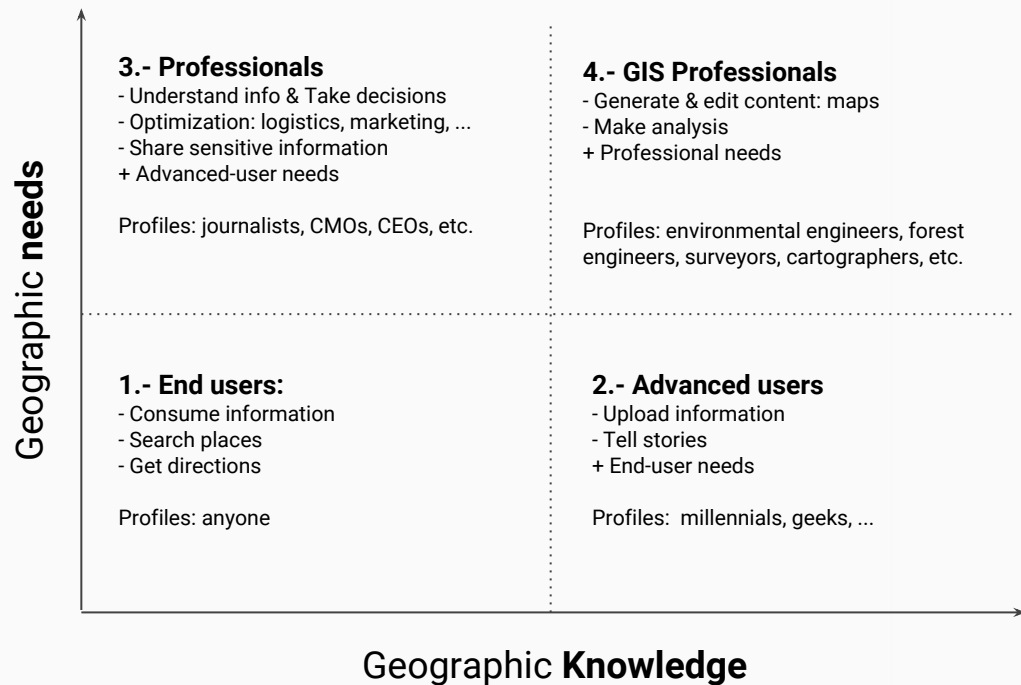
Intergovernmental
Oceanographic
Commission



Web mapping: The history



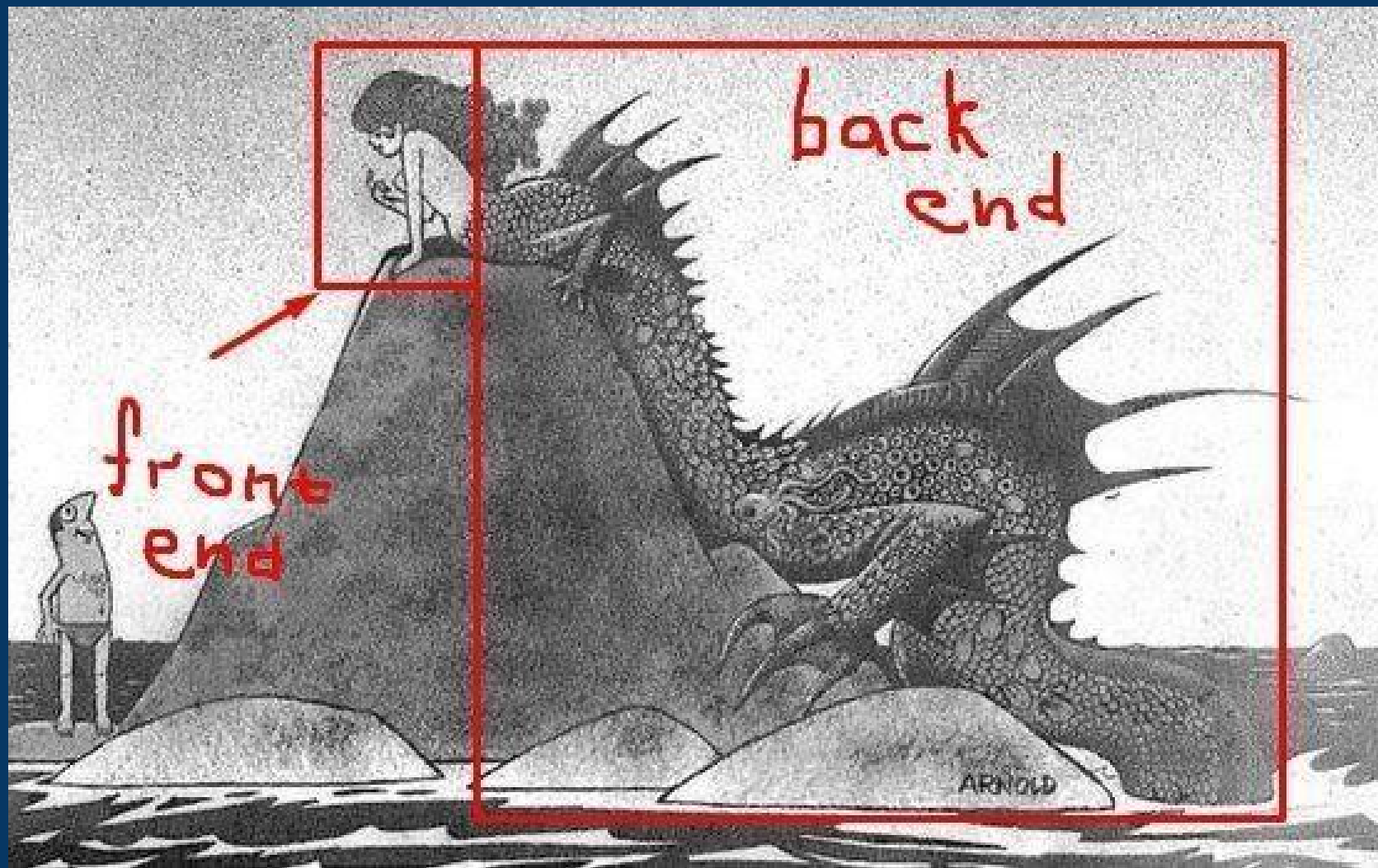
Why web mapping is so trendy



Many
NEEDS

Many
PROFESSIONAL PROFILES

Many
ROLES





Product	Short description
ArcGIS Online & Portal for ArcGIS	Online platform to manage and share: data, user, maps, etc.
ArcGIS Web App Templates	Javascript configurable applications : viewer, editor, social
Web AppBuilder for ArcGIS	GUI to create web app but it is also a framework to develop applications based on maps
ArcGIS API for JavaScript	JavaScript Library for ArcGIS
ArcGIS Online Content	Ready to use content available through APIs and GUI
ArcGIS Open Data	Online platform to search and manage Open Data
Koop	Koop extracts geographic data from third party providers, transforms it in various formats.
ArcGIS Server	Server for creating and managing GIS Web Services, applications and data
ArcGIS Online & ArcGIS Server REST APIs	

[Free developer accounts](#)



Name

ST_Buffer — (T) Returns a geometry covering all points within a given distance from the input geometry.

Synopsis

geometry **ST_Buffer**(geometry *g1*, float *radius_of_buffer*);

geometry **ST_Buffer**(geometry *g1*, float *radius_of_buffer*, integer *num_seg_quarter_circle*);






geometry **ST_Buffer**(geometry *g1*, float *radius_of_buffer*, text *buffer_style_parameters*);

geography **ST_Buffer**(geography *g1*, float *radius_of_buffer_in_meters*);

Description

Returns a geometry/geography that represents all points whose distance from this Geometry/geography is less than or equal to distance.

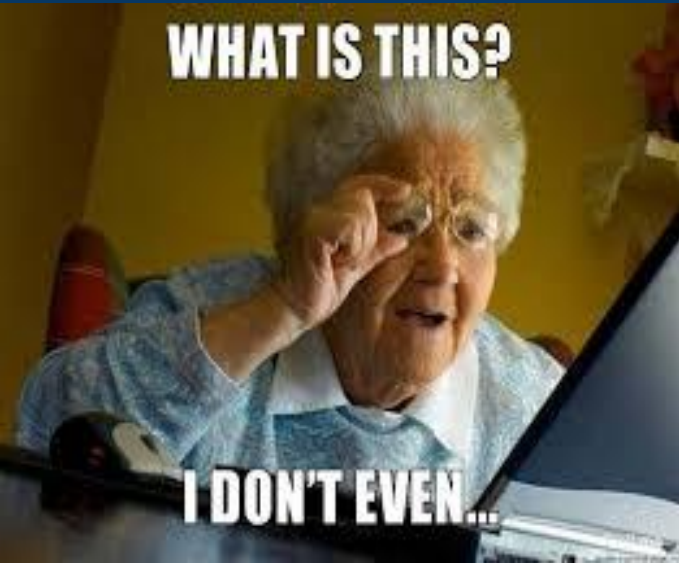
Geometry: Calculations are in the Spatial Reference System of the geometry. Introduced in 1.5 support for different end cap and mitre settings to control shape.

 quad_segs=8 (default) <pre>SELECT ST_Buffer(ST_GeomFromText('POINT(100 90)'), 50, 'quad_segs=8');</pre>	 quad_segs=2 (lame) <pre>SELECT ST_Buffer(ST_GeomFromText('POINT(100 90)'), 50, 'quad_segs=2');</pre>	
 endcap=round join=round (default) <pre>SELECT ST_Buffer(ST_GeomFromText('LINESTRING(80 80,180 180,180 80)'), 10, 'endcap=round,join=round');</pre>	 endcap=square <pre>SELECT ST_Buffer(ST_GeomFromText('LINESTRING(80 80,180 180,180 80)'), 10, 'endcap=square,join=round');</pre>	 endcap=flat <pre>SELECT ST_Buffer(ST_GeomFromText('LINESTRING(80 80,180 180,180 80)'), 10, 'endcap=flat,join=round');</pre>

SQL PostGIS Special Spatial Functions

NODE.JS SERVICES





Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

Node.js uses an **event-driven, non-blocking I/O** model that makes it lightweight and efficient.

Node.js' package ecosystem, **npm**, is the largest ecosystem of open source libraries in the world.

1. Web development in a dynamic language (JavaScript) on a VM that is incredibly fast (V8). It is **much faster than Ruby, Python, or Perl**. Ability to handle thousands of concurrent connections with minimal overhead on a single process.
2. **JavaScript is perfect for event loops** with first class function objects and closures. People already know how to use it this way having used it in the browser to respond to user initiated events. It is arguably **the most popular web programming language**.
3. **Using JavaScript on a web server as well as the browser** reduces the impedance mismatch between the two programming environments which **can communicate data structures via JSON** that work the same on both sides of the equation. Duplicate form validation code can be shared between server and client, etc.

JSON APIs

Building light-weight REST / JSON api's is something where node.js really shines. Its non-blocking I/O model combined with JavaScript make it a great choice for wrapping other data sources such as databases or web services and exposing them via a JSON interface.

Single page apps

If you are planning to write an AJAX heavy single page app (think gmail), node.js is a great fit as well. The ability to process many requests / seconds with low response times, as well as sharing things like validation code between the client and server make it a great choice for modern web applications that do lots of processing on the client.

WHY USE NODE.JS?

1. **Faster web Apps: Asynchronous events**
2. **NPM: Bigger open source libraries in the world**
3. **Javascript Frontend/Backend. JSON Apis.**
4. **Really now is COOL.**







★ arc-node public

Node module to work with ArcGIS Online and ArcGIS Server

How to install it

Just write this in your prompt: `npm install --save arc-node`

And you are ready to go, just instantiate the object like this:

```
var ArcNode = require('arc-node'),
    service = new ArcNode(<config object>);
```

Check here the description of the *<config object>* parameter.

Documentation

When you have instantiate the service you will have available methods to:

- [Get a new token](#)
- [Check if a feature service exists](#)
- [Create an empty feature service](#)
- [Determine the SQLType for an EsriType](#)
- [Create a JSON object describing a field in a layer](#)
- [Create a JSON object describing a layer](#)
- [Add layers to a feature service](#)
- [Add features to a layer](#)

 `npm install arc-node`

 [esries](#) published 3 months ago

0.2.9 is the latest of 27 releases

github.com/esri-es/ArcNode

AGPL-3.0 license

Collaborators



Stats

44 downloads in the last day

790 downloads in the last week

1,048 downloads in the last month

No open issues on GitHub

No open pull requests on GitHub

Try it out

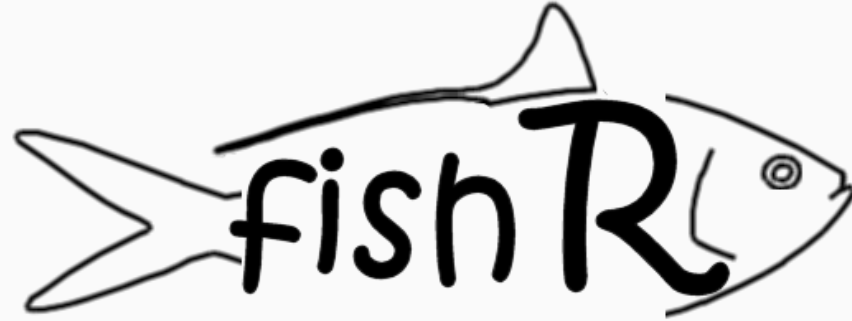


OHMAGIF.COM

USING R IN WEB SERVICES

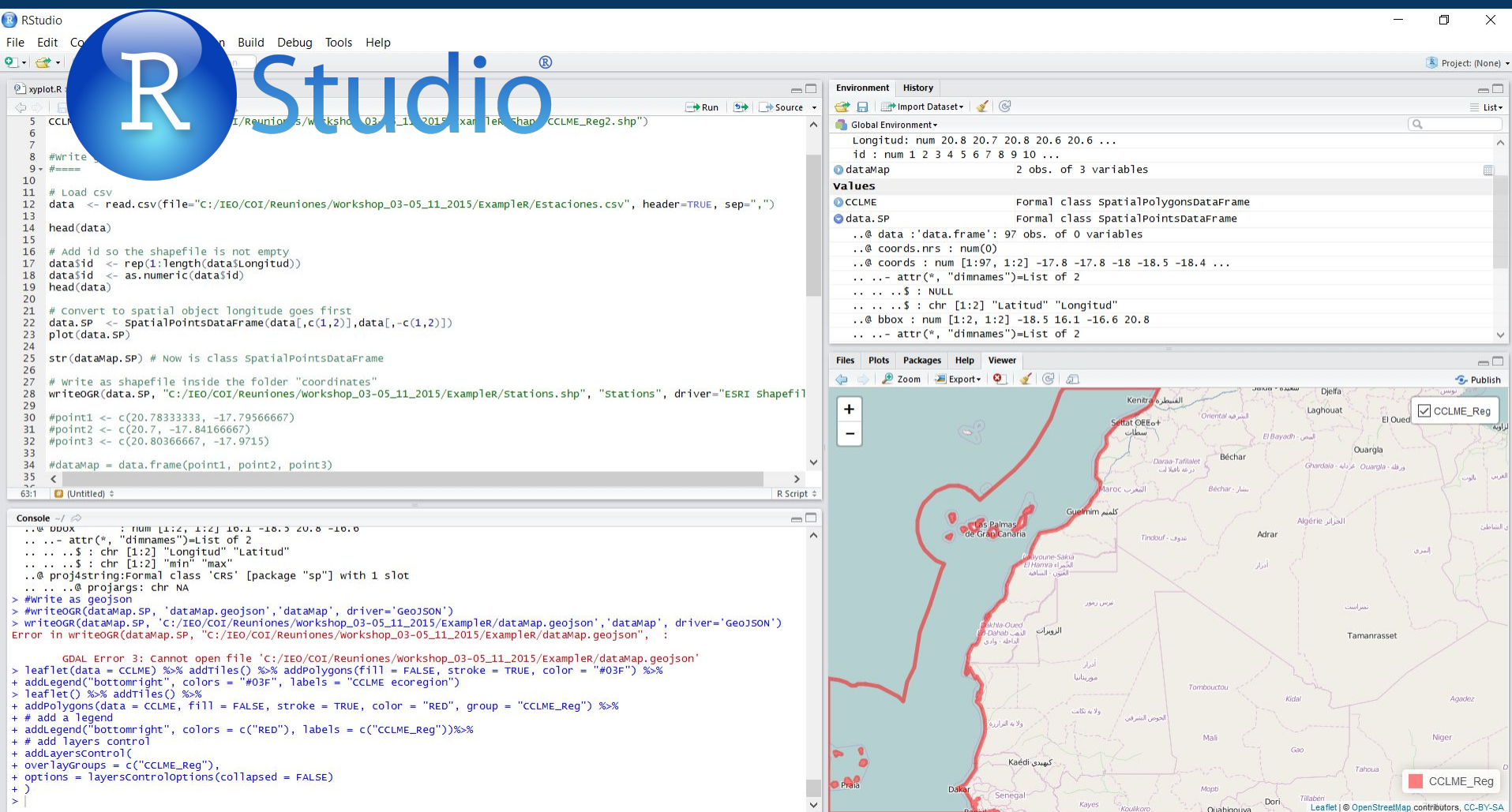


In oceanography and fisheries we have a lot of scripts developed with R



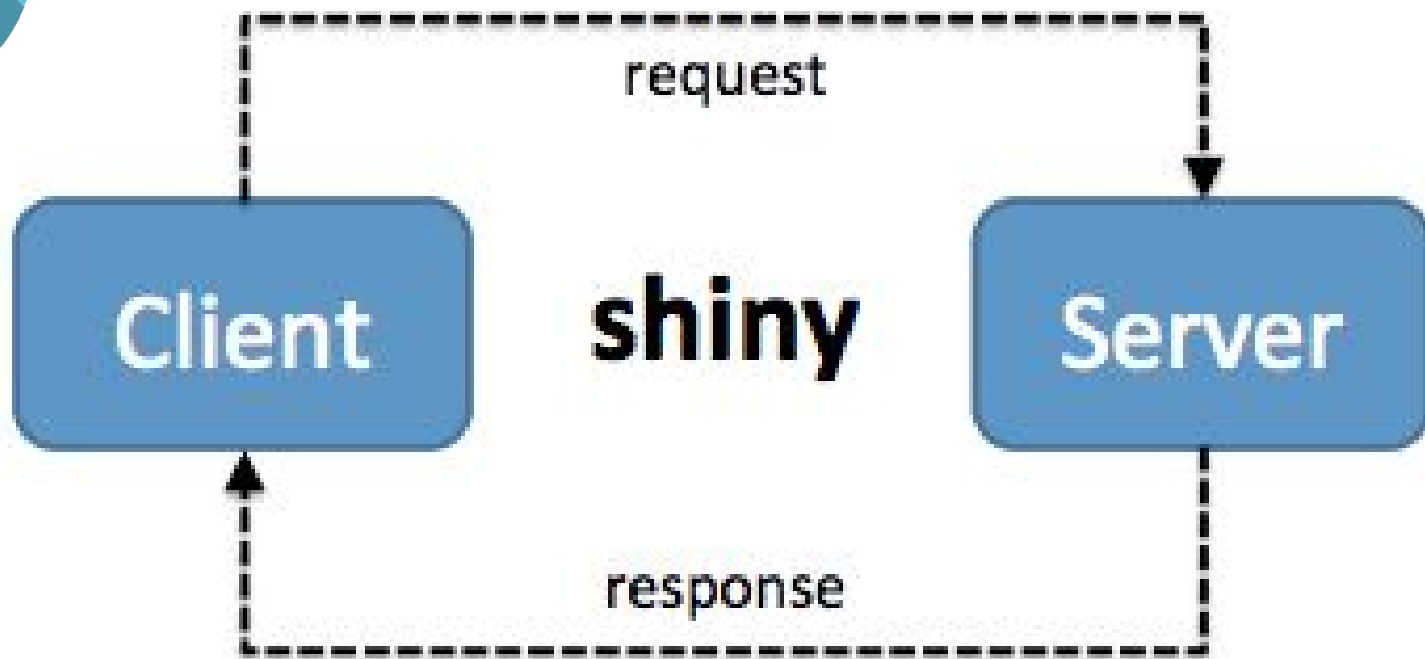
FLR-project

oceR





In my opinion, not enough





- Ubuntu 14.04 Droplet with 1 GB of RAM and 1 GB of swap space or 2 GB of RAM
- The latest version of R installed

Shiny Server is useful not only for hosting Shiny applications, but also for hosting interactive [R markdown documents](#).

By setting up Shiny Server, we are able to host Shiny applications and interactive R documents on the web in a way that is accessible to the public.



Gallery

This gallery contains useful examples to learn from. Visit the [Shiny User Showcase](#) to see an inspiring set of sophisticated apps.

Interactive visualizations

Shiny is designed for fully interactive visualization, using JavaScript libraries like [d3](#), [Leaflet](#), and [Google Charts](#).



SuperZip example



Bus dashboard



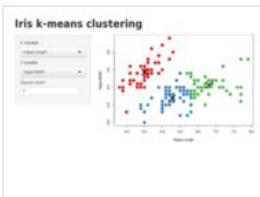
Movie explorer



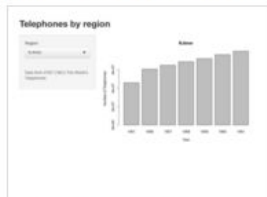
Google Charts

Start simple

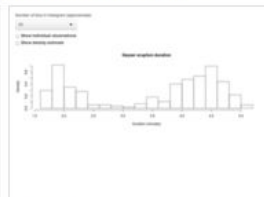
If you're new to Shiny, these simple but complete applications are designed for you to study.



Kmeans example



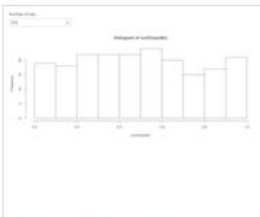
Telephones by region



Faithful



Word cloud



Single-file shiny app



Shiny Server

	Open Source Edition	Professional Edition
Overview	<ul style="list-style-type: none">• Great for hosting lightweight public applications• Does not support authentication or SSL• Single R process per application	<ul style="list-style-type: none">• All of the features of open source; plus:• Perfect for workgroups and enterprises• Supports authentication and SSL• Includes admin dashboard with both realtime and historical performance data• Can use multiple R processes per app
Documentation	Getting Started with Shiny Server	Shiny Server Professional Admin Guide
Support	Community forums only	<ul style="list-style-type: none">• Priority Email Support• 8 hour response during business hours (ET)
License	Open Source AGPL v3	RStudio License Agreement
Pricing	Free	<ul style="list-style-type: none">• 20 concurrent users: \$9,995/server/year• Additional 20 concurrent users: \$4,995/server/year• Additional 150 concurrent users: \$14,995/server/year* <p>Academic and Small Business discounts available</p>