

Instructivo sobre las capas de datos del ISO para el Golfo de Guayaquil

Marzo 6, 2015

Al aplicar la metodología sugerida por *CI* al estudio regional del ISO en el Golfo de Guayaquil, se comenzó con las tareas de comprender las metas y su filosofía y estudiar las capas de datos que describen diferentes aspectos de las metas y del cálculo de los índices para ellas.

Entre las primeras decisiones a tomar estuvo el decidir cuáles *capas de datos del estudio global* de Ecuador se utilizarían, sustituyendo los datos globales por datos locales y cuáles *capas de datos nuevas* se definirían para los cálculos locales.

Una vez recolectados los datos de cada meta y definidas las capas de datos a utilizar, el siguiente paso es revisar los datos, preparar los archivos que corresponden a cada capa de datos. Es importante recordar que una capa de datos puede describir: el estado de una meta, una tendencia, una presión o una resiliencia.

Este documento explica cuáles son los **pasos técnicos para elaborar los archivos necesarios** (sección 1) y **para colocarlos en la plataforma donde se realizarán los cálculos** (sección 2).

Este instructivo se elaboró para no profesionales de la computación, sin embargo, es necesario que quien elabore los archivos de datos y los coloque en la plataforma de cálculos maneje algunos conceptos y herramientas fundamentales, estos son:

1. Un editor de *texto plano*, es decir un software que permita colocar caracteres alfanuméricos en un archivo, sin agregarles formato. Puede ser Notepad o algo equivalente, no se debe usar Word.
2. La noción de *directorio o carpeta* en su computador local. El disco de almacenamiento de todo computador está organizado en una jerarquía. Cada usuario tiene un *directorio home* que es la cabeza de la jerarquía de sus archivos almacenados en ese computador. Dentro (o debajo) de ese *directorio home*, el usuario puede crear otros directorios y archivos. En general, dentro de cada directorio puede haber otros directorios y archivos. (Nota: en Windows el usuario puede crear directorios y moverse a través de la jerarquía usando Explorer, en Mac se hace con Finder o con la línea de comandos desde un *Terminal*.)
3. La noción de *path o camino* dentro de la jerarquía de archivos y directorios del usuario. El camino para llegar a un archivo se define como la secuencia de directorios (unos contenidos

dentro de los otros) que se debe seguir para acceder un archivo. Generalmente el camino se indica colocando los nombres de los directorios en orden, separados por el smbolo “/”. Por ejemplo, el *path* /Users/abadmota/ProyectoECUADOR es el camino completo para llegar al documento llamado instructivoCapasDatos.pdf en la computadora personal de Soraya Abad Mota. Para practicar, vale la pena revisar cuál es el camino completo de varios archivos en su computador. En Windows, un ejemplo de camino puede ser:

C:\Users\lelys\Documents\GitHub

4. *Instalación de software en su computador*, en particular, para lograr la colocación de los archivos de las capas de datos, se requiere instalar: *Git* y *GitHub* en los computadores individuales de cada consultor.

1. Archivos de la capa de datos

En esta etapa, cada consultor debe crear dos **archivos por cada capa de datos** que describa su meta o una presión o una resiliencia.

1. un archivo con los datos de esa capa,
2. un archivo descriptivo de la capa de datos, mejor conocido como archivo *README*.

Nombres de los archivos

En esta subsección se dan las reglas sobre cómo nombrar los archivos que contendrán los datos de cada capa y da algunos lineamientos de cómo colocar los datos dentro de esos archivos y cómo entregarlos.

Cada archivo que contiene los datos para una capa de datos, tiene una nomenclatura especial para su nombre. El nombre del archivo tiene varios componentes, cada componente se separa del siguiente con un guión bajo (_) así:

prefijo_nombreCapa_sufijo.extension

No se deben colocar espacios en blanco en ninguna parte del nombre del archivo.

En el **prefijo** se colocan las siglas de una meta (dos letras) o las siglas de una sub-meta (tres letras), todas las siglas posibles se presentan a continuación:

Meta

Food Provision

Artisanal Fishing Opportunity

Natural Products

Coastal Protection

FP:

AO

NP

CP

Sub-metas

Fisheries **FIS** and Mariculture **MAR**

Carbon Storage	CS	
Livelihoods and Economies	LE:	Livelihoods LIV y Economies ECO
Tourism and Recreation	TR	
Sense of Place:		Lasting Special Places LSP y Iconic Species ICO
Clean Waters	CW	
Biodiversity	BD:	Habitats HAB y Species SPP

Por ejemplo, todos los archivos para capas de datos para la meta *pesca artesanal* deben tener nombres que comiencen con *ao* así:

`ao_nombreCapa_sufijo.extension`

El componente del nombre llamado **nombreCapa** está formado por una o más palabras o abreviaturas de palabras que ayudan a entender de qué se trata la capa, todas esas palabras deben ir separadas por guiones bajos (-) sin espacios en blanco en ninguna parte.

Un ejemplo de los más sencillos, tomado de los artículos que describen las capas, es el nombre general de la capa del porcentaje de desempleo de la meta de Turismo y Recreación, el nombre de esta capa es: **tr_unemployment**. El prefijo es **tr** que son las siglas de la meta Turismo, el nombre de la capa consiste de una sola palabra unemployment.

Hay otras capas con nombres más largos, por ejemplo: *cp_habitat_extent_rank* es una capa de la meta Protección Costera descrita por tres palabras: *habitat*, *extent* y *rank*. La capa se refiere a la extensión del habitat por categoría, coral o manglar.

Para el sufijo, como se trata de datos del Golfo de Guayaquil para el año 2015, colocaremos: **_gye2015** después del nombre de la capa. Adicionalmente, como parte de este sufijo, colocaremos las iniciales del consultor que provee esos datos, de modo que el prefijo, nombre de capa y sufijo del archivo de turismo indicado anteriormente, si lo genera Antonio Perrone sería:

`tr_unemployment_gye2015_AP`

Los archivos de datos que existen en Github para Ecuador actualmente son todos globales, por ello, casi todos tienen el sufijo *_gl2014*.

El último componente del nombre de un archivo, de una capa de datos, se denomina la **extensión del archivo**, está separado por un punto (.) del sufijo e indica el tipo de archivo. En este proyecto estaremos creando archivos donde los datos están separados por comas y la extensión de ese tipo de archivo es *csv* (comma separated value). En consecuencia, el nombre completo del archivo para la capa de datos de turismo indicada anteriormente es:

`tr_unemployment_gye2015_AP.csv`

Contenido de los archivos

Cuando un consultor está dispuesto a crear una capa de datos pues tiene datos para ello, crea un archivo en excel con el nombre que corresponda, según la capa de datos a que se refieran

los datos, como se explica en la sección sobre el nombre de los archivos.

Los detalles del contenido del archivo para una capa de datos, depende de si la capa de datos ya existe en el repositorio para Ecuador o si se trata de una nueva capa definida por el consultor.

Si la capa de datos ya existía en el estudio global de Ecuador, realizado por CI, el archivo excel que se cree para los datos regionales, debe tener las mismas columnas, con los mismos nombres, que la capa con datos globales que existe en Github.

Por ejemplo, para la meta Pesca Artesanal, existe un archivo llamado *ao_need_gl2014.csv*, que tiene las siguientes columnas: `rgn_id` `year` `value`
el consultor que trabaja con esta meta debe colocar un valor para esta capa, por cada región y por cada año disponible en su archivo en excel y llamar a las columnas con esos mismos nombres.

La columna **rgn_id** aparece en todas las capas de datos; cuando aparece al lado de un valor indica que ese valor es para tal región. Las regiones que estaremos usando para el Golfo de Guayaquil, en los archivos de datos con su `rgn_id` son:

rgn_id	label
1	El Oro
2	Guayas
6	Santa Elena

Nótese que hay un salto del id utilizado para Guayas a Santa Elena, de 2 a 6, la razón es que ya hay algunos datos que tenían 5 regiones, donde la única que no estaba era Santa Elena. Para darle continuidad a esos valores de id, utilizamos el 6 para Santa Elena.

A medida que se incluyen datos en el archivo, se puede salvar como un archivo excel (.xlsx). Luego de terminar de cargar los datos se vuelve a salvar el archivo, en formato *CSV* usando el menu de almacenamiento de excel.

Un ejemplo, el archivo *ao_access_gl2014.csv* contiene:

```
"rgn_id","value"  
1,0.555771907253878  
5,0.555771907253878  
4,0.555771907253878  
2,0.555771907253878  
3,0.555771907253878
```

La primera línea del archivo tiene los nombres de las columnas que contiene ese archivo, esos nombres deben ser escritos de la misma forma en que aparecen en la aplicación o en el archivo de capas del Github. En ningún caso pueden esos nombres tener espacios en blanco, de modo que es importante recordar que **ni los nombres de las columnas, ni los nombres de los archivos pueden tener espacios en blanco, ni acentos.**

En el caso de capas de datos nuevas (que no existen en Github) se deben crear nuevos nombres para las nuevas columnas.

Los archivos con los datos globales, se pueden acceder utilizando *Github*. Los pasos a seguir para ello son:

1. Entrar a <https://github.com/OHI-Science/ecu>
2. Aparece una lista de carpetas disponibles en ese sitio. Se selecciona la carpeta llamada: `subcountry2014`
3. Ahora aparecen varias carpetas y varios archivos. Se selecciona la carpeta llamada: `layers`
4. Aparecen todos los archivos `.csv` en orden alfabético. Cada uno contiene los datos de la capa de datos correspondiente al nombre del archivo.

Los archivos con extensión `.csv` cuyo nombre tenga como sufijo `_gl2014` contienen datos globales del año 2014. Algunos de estos archivos serán sustituidos por otros con datos regionales y su sufijo será `_gye2015`, como se indicó más arriba.

5. Al seleccionar un archivo, aparecen los datos visualizados en una tabla. La tabla tiene la lista de los campos que contiene como encabezado.
6. En esa pantalla hay tres pestañas: Raw, Blame y History, la única relevante para los consultores es Raw. Al seleccionar esa pestaña, se puede ver el archivo plano con los datos y su primera línea de anotación. Se puede salvar esa pantalla con lo cual se obtendrá el archivo `.csv` correspondiente.

Datos crudos para construir una capa

Si no se tiene una capa de datos ya calculada, sino que se tienen datos crudos de varias fuentes, a partir de los cuales se pueden calcular los datos para la capa, con ciertas manipulaciones de archivos y ejecución de programas en R, el consultor puede preparar varios archivos con los datos crudos y un archivo explicativo de todos los archivos, y subir todos esos archivos al repositorio. Luego, el equipo de computación podrá construir y ejecutar las rutinas necesarias para formar el archivo final de la capa de datos en cuestión.

Este instructivo aplica igualmente a esos archivos, inclusive los estándares para nombrar los archivos, pero en lugar de poner un nombre de capa de datos, se colocará un nombre apropiado a los datos crudos que el archivo contenga y en el documento explicativo, se colocan los detalles, como se explica en la próxima sección.

Documento explicativo de los archivos de datos

Además de los campos (columnas) de cada archivo de datos, es importante que se documente lo siguiente de los datos recolectados, que se van a incluir en la plataforma del estudio regional.

1. Institución recolectora original (cuál institución hizo la recolección de los datos de su fuente).
2. Institución que provee los datos a este proyecto (si es distinta de la recolectora).
3. Citas detalladas de la fuente, si se trata de uno o más artículos científicos.
4. Método de captura de los datos. Si fue un estudio realizado para obtener esos datos, si se hizo utilizando algún instrumento de medición, cuál método se aplicó, si se utilizó una encuesta o formulario para recogerlos, etc.
5. Período de tiempo que cubren los datos (año, mes, día desde y año, mes, día hasta).
6. Unidad en la cual vienen los valores.
7. Significado, explicación de cómo se deben interpretar los valores.
8. Observación o comentario adicional sobre los datos.
9. Significado y explicación de las columnas que se incluirán en el archivo, si son diferentes de las que se especifican en OHI-ecu o si la capa de datos es nueva, creada ad hoc para este estudio regional.

Los archivos de datos que se creen para una meta, deben ir acompañados de un documento (archivo de texto, no de word) donde se especifiquen los aspectos (1 al 9) enumerados más arriba. Esos archivos de documentación de los datos, se incluirán en el GitHub también, junto con los datos.

El nombre del archivo que contenga los datos descriptivos de la capa de datos, debe ser **README_nombreCapaDatos.txt**, donde nombreCapaDatos son las palabras que componen el nombre de la capa de datos, solamente, es decir sin los prefijos ni sufijos del nombre del archivo de datos.

Para crear y editar este archivo de texto se necesita usar un editor como *Notepad*, que no le agregue formato al archivo sino simplemente guarde las palabras que describen a los datos.

2. Cómo colocar los archivos en la plataforma Git

En la plataforma Git, en el repositorio de Ecuador, accesible a través de la página: <https://github.com/OHI-Science/ecu> se encuentra un directorio llamado *subcountry2014*, dentro de ese directorio se ha creado otro directorio llamado *prep*. Ese directorio *prep* contiene un directorio por cada meta o sub-meta, con los siguientes nombres:

1.1_FIS
1.2_MAR
10.1_SPP
10.2_HAB
2_AO
3_NP
4_CS
5_CP
6.1_LIV
6.2_ECO
7_TR
8_CW
9.1_ICO
9.2_LSP
presiones
resiliencia

Para ser más precisos, Git es el software completo que tiene todas las funciones de almacenamiento de archivos, pero el usuario consultor, va a interactuar es con *GitHub*, que es una interfaz a Git. El ícono de *GitHub* es un gato morado . *Github* es una aplicación independiente que abre su propia ventana, que no es una ventana del navegador. Los resultados de las operaciones de “subir” archivos que realizarán los consultores, también se pueden ver en un navegador, como se ha hecho hasta ahora en el proyecto, pero para “subir” los archivos se debe usar el *gatico morado*.

En el repositorio *ecu*, se encuentran los datos globales del estudio de Ecuador, distribuidos en cinco regiones costeras, como ilustración. Para el estudio regional en el Golfo de Guayaquil, se creó otro repositorio específico llamado *gye*, accesible de la misma forma en <https://github.com/OHI-Science/gye> y cuya carpeta para este proyecto regional es *region2015*, el cual también contiene una carpeta llamada *prep*, donde los consultores han de colocar todos sus archivos de datos, necesarios para hacer los cálculos de índices.

El escenario que vamos a describir consiste de:

1. *un consultor en su computador local*, donde tiene almacenados los archivos de datos y de texto, contruidos y nombrados, como se indica en la sección 1 de este instructivo.
2. Este consultor quiere “subir” sus archivos al repositorio donde se realizarán los cálculos.
3. El computador local debe tener instalados los software de *Git* (control de versiones) y *GitHub* (la aplicación que funciona de manera independiente en el computador, esta es la interfaz GUI a Git).
4. En ese computador local, debajo del directorio home del usuario, debe crearse un directorio llamado *github*. Por ejemplo, en Mac el *camino completo* de este directorio para el usuario abadmota, sería: `/Users/abadmota/github`
En Windows, para el usuario *lelys* sería:

`C:\Users\lelys\github`

NOTA: Es importante recordar que, el contenido de cualquier repositorio Git se puede observar en <https://github.com/usuario/nombre-repo>, y para nuestro caso, el usuario dueño de estos repositorios es OHI-Science y los dos repositorios que nos ocupan son *ecu* y *gye*, dentro de esos repositorios, nos interesan la carpeta *subcountry2014* en *ecu*, *region2015* en *gye* y en ambos la carpeta *prep*. En otras palabras, se pueden ver los archivos del repositorio *ecu* estudio *subcountry2014* y carpeta *prep*, usando el url: <https://github.com/OHI-Science/ecu/subcountry2014/prep>.

5. Antes de poder “subir” archivos al repositorio del estudio, es necesario “clonar” el repositorio (ver pasos más abajo), lo cual copia la estructura del repositorio y todos sus archivos al computador local del consultor. Este proceso de “clonar” se hace una sola vez y la idea es clonar ese repositorio en el directorio *github* del usuario que se creó en el paso anterior.

Para “**clonar**” un repositorio la primera vez, se hace lo siguiente:

1. Abrir la aplicación GitHub (la del *gatico morado*).
2. Arriba y a la izquierda debajo de las opciones, aparece un + y al lado una flecha que abre un menu. Se abre ese menu que tiene tres opciones: **Add Create Clone**.
3. Seleccionar **Clone** en ese menu; en la línea que dice **filter**, colocar el nombre del repositorio a clonar, en nuestro caso *gye*.
4. Al encontrar el repositorio, se selecciona y el botón cambia a **Clone nombre-repositorio**. Al oprimir este botón, el software pregunta cuál nombre se le pone al clon y en cuál directorio se quiere colocar. Allí se deja el mismo nombre al clon y se selecciona el directorio dónde colocarlo; en el apartado **Where:** se debe elegir el directorio *github* creado para el usuario en pasos anteriores, si no aparece automáticamente ese nombre.

5. En esta pantalla donde se especificaron el nombre y el directorio, se oprime el botón **Clone....** El software responde con el resultado del proceso. Si todo sale bien, todos los directorios y archivos del repositorio se deben encontrar en el computador local, en el directorio indicado arriba.

Una vez comprendidas y ejecutadas todas las tareas especificadas antes, los pasos para finalmente **colocar archivos en un repositorio Git** son los siguientes:

1. Ubicarse en el directorio: usuario/github o en Windows en:

`C:\Users\lelyys\github\`

2. Abrir la aplicación Github (la del gato morado), seleccionar el repositorio donde se van a colocar (“subir”) los archivos.
3. Sincronizar con el repositorio a donde se van a subir los archivos, esto se hace oprimiendo el botón **Sync** (arriba a la derecha). Al hacer Sync la pantalla de GitHub, en **Changes**, aparecen los cambios que hayan ocurrido desde la última vez que se sincronizó el repositorio.
4. En el computador local, ubicarse en el directorio local donde se encuentran los archivos a subir.
5. Copiar todos los archivos a “subir” del directorio local donde están, al directorio usuario/github, es decir, el directorio donde se clonó el repositorio en la carpeta específica de prep en el directorio de la meta o sub-meta que corresponda, es decir, se coloca en: usuario/github/gye/region2015/prep/directorio-meta (para Linux o Mac) y en Windows es:

`C:\Users\lelyys\github\gye\region2015\prep\directorio-meta`

6. Al copiar estos archivos, el software se da cuenta y coloca un nuevo cambio. En la parte de abajo de la pantalla, el consultor debe colocar un comentario que consiste de un resumen de cuál fue el cambio y una descripción un poco más explicativa de qué se hizo. Al completar ese comentario, se oprime el botón de **Commit to draft** para registrar ese cambio en el repositorio.
7. Se sincroniza el repositorio, oprimiendo **Sync**. Si el proceso culmina exitosamente, desaparece el cambio y todos pueden ver el nuevo archivo en el repositorio.