



CIMAPS

(Cima Integrated MAPS)
Un sistema de visualización Web para
la integración y difusión de
conocimiento

Fernando García
fido@cima.fcen.uba.ar

En colaboración con Juan Ruiz, Carolina Vera,
Federico Robledo, Paola Salio y Yanina García Skabar

Objetivos:

Desde el punto de vista de los generadores de información:

- Facilitar la difusión de la información generada por la comunidad científica y por los diferentes organismos gubernamentales, utilizando herramientas de visualización web de última generación.
- Incorporar herramientas sencillas que permitan potenciar el análisis de conjuntos de datos complejos en un entorno amigable con los usuarios a los que va destinada la información.



Objetivos:

Desde el punto de vista de los usuarios destinatarios de la información:

- Acceder fácilmente a la información a través de un sitio web moderno, simple y de uso intuitivo.
- Integrar fácilmente diferentes fuentes de información de manera sencilla potenciando el nivel de adaptación de la herramienta a las necesidades de cada usuario.

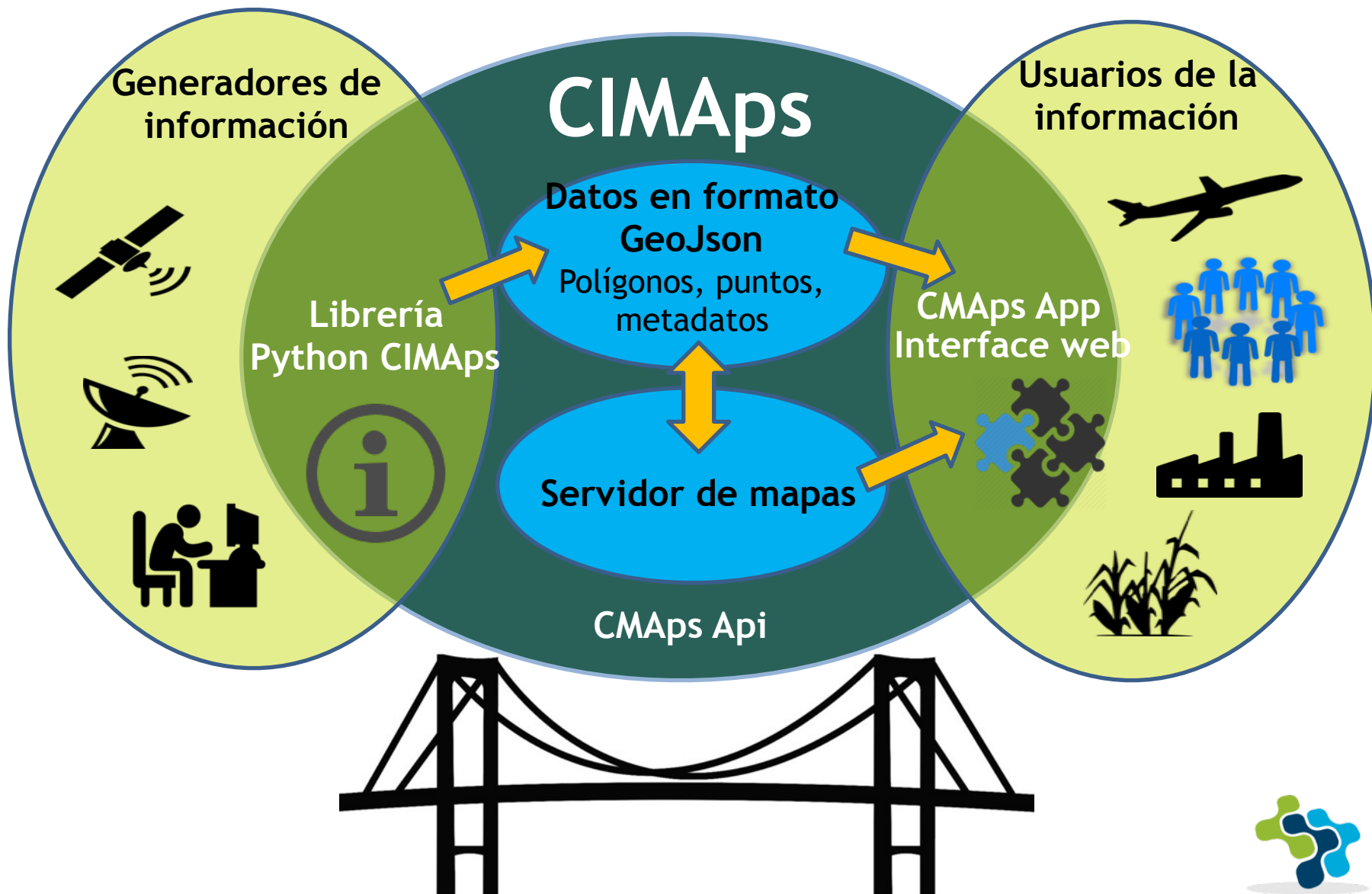


¿Qué es CIMAs?

- CIMAs es una herramienta que comenzó como una plataforma para la difusión de contenidos de diverso tipo y que actualmente es desarrollada por el CIMA en el marco del proyecto de Unidades Ejecutoras.
- CIMAs es un sistema de visualización y análisis de información georeferenciada que cuenta con interfaces simples, intuitivas y amigables tanto para los generadores de información como así también para los usuarios de dicha información.
- CIMAs permite además administrar en forma sencilla y eficiente diferentes sitios web y definir estructuras o categorías para organizar la información que se presenta al usuario.
- CIMAs está basado en tecnologías open-source de última generación como Python, NodeJS, ReactJS



¿Cómo funciona CIMAPS?



¿Cómo funciona CIMAPS?

Librería Python: CIMaps Lib

- El usuario generador lee datos utilizando herramientas de Python (ej. Numpy, PyGrib). Define el nombre del conjunto de datos y la categoría a la cual pertenecen los datos (ej. “Pronósticos del modelo WRF” que pertenece a la categoría “Modelos numéricos”).
- Define el nombre de las variables (ej. “Temperatura”). También se deben definir las dimensiones (vertical, temporal u otras) y los metadatos (descripción, unidades, etc). Una vez definido esto mediante una librería de Python los datos son enviados al servidor y procesados.



¿Cómo funciona CIMAPS?

CIMAPS Api

- En el servidor, CIMAPS procesa automáticamente los datos. Los datos son transferidos y almacenados en formato GeoJson, que representa un estándar desde el punto de vista de la visualización, transmisión y almacenamiento de la información.
- La unidad de procesamiento es el “layer” o capa de datos: un conjunto de polígonos o puntos que describen una única variable en diferentes puntos del espacio en un tiempo determinado (ej. La temperatura superficial del mar a las 12 UTC del 1 de enero de 2005).



¿Cómo funciona CIMApps?

CIMApps App

- El usuario de la información, se conecta a un sitio web y se le presenta un menú con los diferentes datos disponibles. La estructura de este menú lo genera CIMApps automáticamente en función de los datos provistos por los generadores de la información.
- El usuario despliega los datos seleccionados, geográficamente referenciados con funcionalidades de zoom y superposición de diferentes conjuntos de datos en hasta 4 pantallas simultáneas.



¿Cómo funciona CIMAPS?

Ejemplo de uso

- Con CIMAPS los usuarios generadores de datos deben poder leer y generar sus datos como variables de Numpy.
- Una vez hecho esto, las funciones de la librería CIMAPS les permitirá publicar sus datos en Internet en pocas líneas de código.



¿Cómo funciona CIMAPS?

Ejemplo de uso

```
import cimaps as cm

api = cm.Api('fido')

python = api['demo']['python']

ejemplos = python.create_project(name='ejemplos', description='Ejemplos')
ejemplos.create_variable(name='grib', description=u'Grib')
```



¿Cómo funciona CIMAPS?

```
import datetime
import pygrib
import cimaps as cm
import numpy as np

grib = '../data/gfs.t00z.pgrb2.0p25.f000'
grbs = pygrib.open(grib)
grbs.seek(0)
grb = grbs.select(name='2 metre temperature')[0]
data = grb.values
min = np.min(data)
max = np.max(data)
lat, lon = grb.latlons()
grbs.close()

gen = cm.PolyGen()
gen.lons = map(lambda x: x-360, lon)
gen.lats = lat
gen.values = data
gen.levels = np.arange(min, max, (max - min) / 100.0)
palette = cm.Palette(gen.levels)
polygons = gen.get_polygons()

api = cm.Api('fido')
grib_variable = api['demo']['python']['ejemplos']['grib']
layer = grib_variable.create_layer(date_time=datetime.datetime(year=2016, month=9, day=28),
                                  palette=palette, levels=gen.levels)

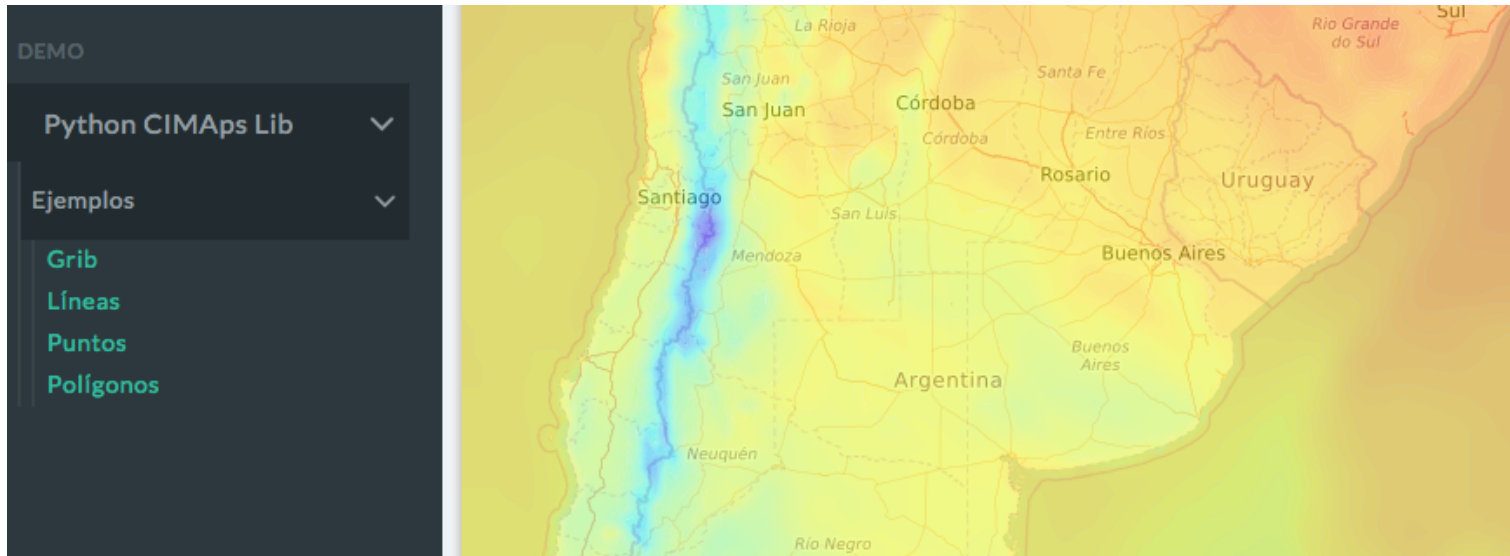
layer.add(polygons)
layer.upload()
```



¿Cómo funciona CIMAPS?

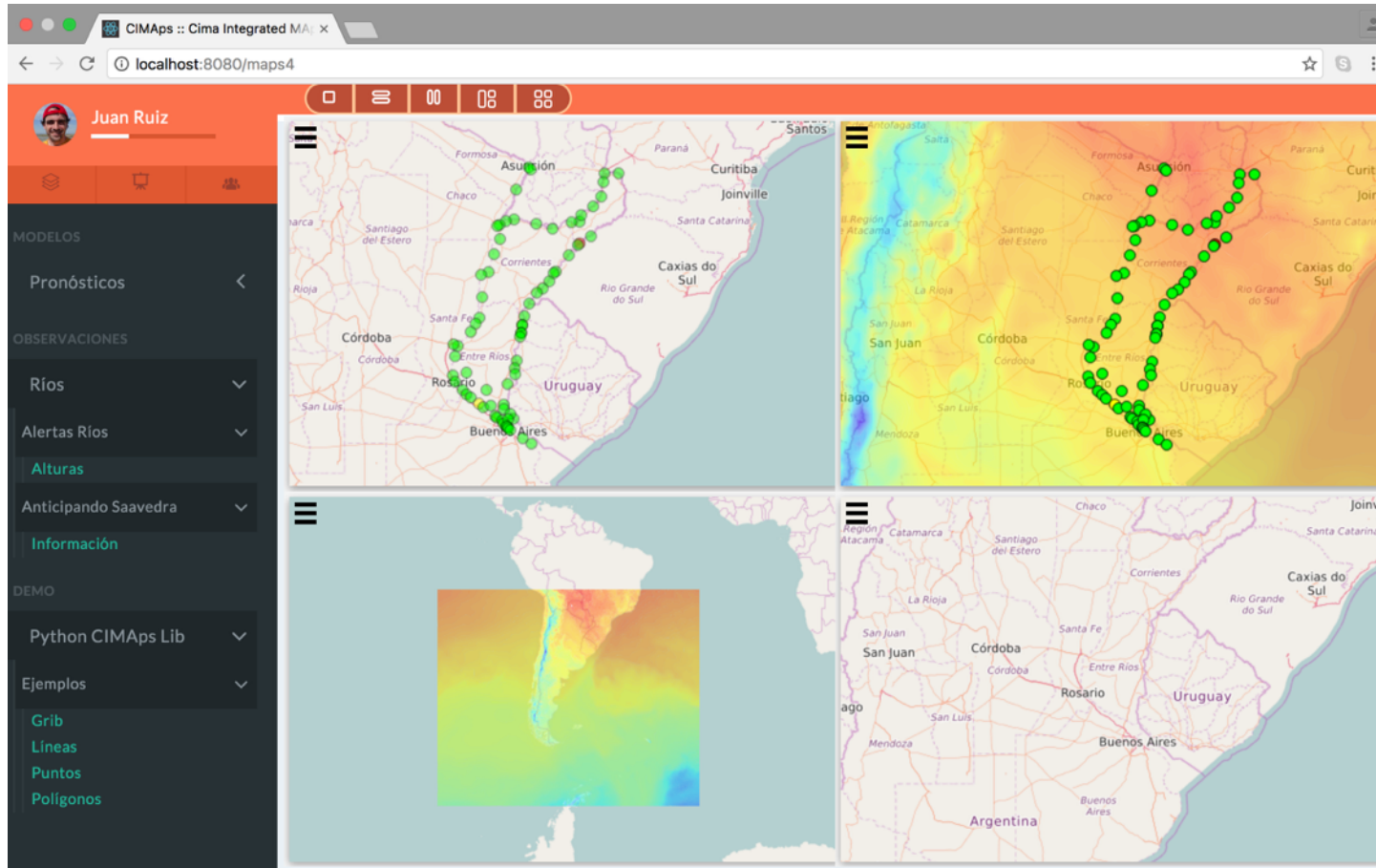
Ejemplo de uso

- Una vez que los datos son enviados al servidor a través de la librería de Python, el sistema mediante el menú de acceso permite visualizar los datos



¿Cómo funciona CIMaps?

Ejemplo de uso



¿Cómo funciona CIMAPS?

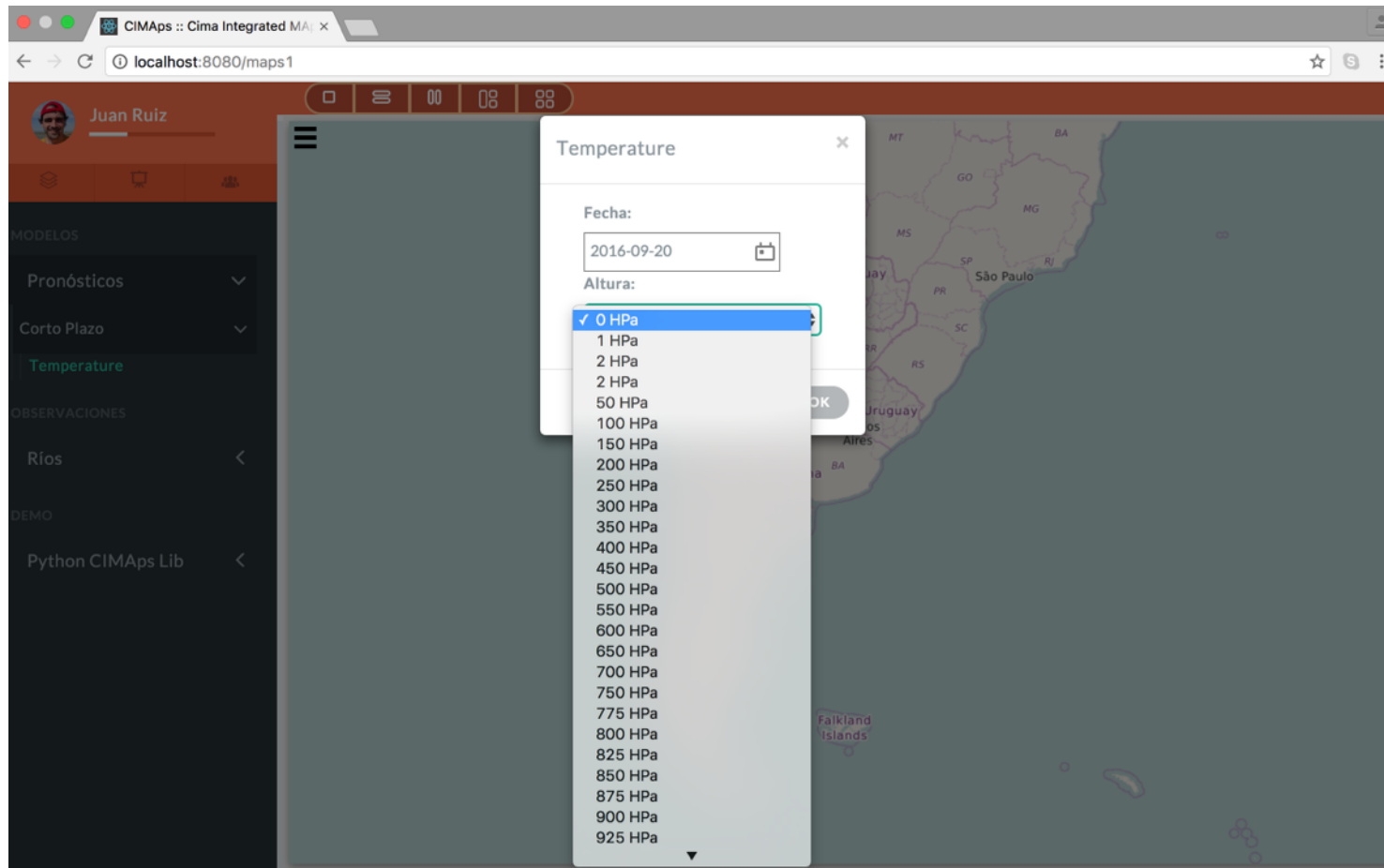
Ejemplo de uso

The screenshot displays the CIMAPS web application interface. The browser address bar shows `localhost:8080/maps5`. The user profile is **Juan Ruiz**. The interface includes a sidebar with navigation options: **Pronósticos** (with sub-items **Corto Plazo** and **Temperature**), **OBSERVACIONES** (with sub-items **Ríos**, **Alertas Ríos**, and **Alturas**), and **DEMO** (with sub-items **Python CIMAPS Lib**, **Ejemplos**, **Grib**, **Líneas**, **Puntos**, and **Polígonos**). The main area shows a map of Uruguay with a topographic overlay. A modal window titled **Alturas** is open, displaying a date selection interface. The date **2016-09-20** is selected. The calendar shows the month of **Sep 2016** with days of the week **SUN MON TUE WED THU FRI SAT**. The selected date **20** is highlighted in blue. Other dates are shown in red (e.g., 3, 10, 17, 24, 28) and black (e.g., 1, 2, 4, 5, 6, 7, 8, 9, 11, 12, 13, 14, 15, 16, 18, 19, 21, 22, 23, 25, 26, 27, 29, 30, 31). The modal includes **Today**, **Clear**, **OK**, and **Cancel** buttons.



¿Cómo funciona CIMAPS?

Ejemplo de uso



Desarrollos futuros pensados para CIMAPS

- Manejo de paletas, cantidad de contornos y frecuencia de contornos por parte de los usuarios finales.
- Realización de animaciones temporales.
- Opción para sincronizar o des-sincronizar espacial o temporalmente los diferentes paneles en los que se visualizan los datos.
- Realización de análisis sencillos de la información: generación de transectas, cortes verticales, series temporales a petición de los usuarios.
- Implementación de métodos de compresión para aumentar la eficiencia de la transmisión de la información a los usuarios.
- Establecimiento de permisos y roles para los diferentes usuarios del sistema.



Acciones futuras pensadas para CIMAs

- Implementación operativa experimental en la comunidad CIMA-DCAO.
- Transferencia de la herramienta a través de convenios con otras instituciones.
- Publicación de la documentación orientada a los usuarios generadores de la información y a los usuarios que acceden a la información.
- Realización de una jornada de capacitación en el uso de la herramienta.
- Generación de ejemplos o templates que se puedan adaptar a los diferentes tipos de datos.



CIMAPS es de todos, debe crecer en función de las necesidades y requerimientos de los usuarios, tanto de los generadores de información como los destinatarios de dicha información.



Todas sus opiniones, sugerencias y aportes son bienvenidos!!!!

