

Geoportal para consulta del acervo institucional de información geográfica, incorporación de nuevas imágenes y funciones.

Proyecto TH1609.1

Informe final

**Subcoordinación de Planeación Hídrica
Coordinación de Hidrología**

Jefe de proyecto: M. en C. Jaime Velázquez Álvarez

PARTICIPANTES

Geóg. Jorge Enrique Brena Zepeda
Ing. Cervando Castillo Romano
M.I. José Alberto Báez Durán

Diciembre de 2016

ÍNDICE

RESUMEN EJECUTIVO	4
1. INTRODUCCIÓN	1
2. OBJETIVOS	2
3. METODOLOGÍA	3
3.1 Mejorar el protocolo para incorporar nuevas imágenes al geoportal.	3
3.2 Incorporar imágenes del satélite SPOT, de las versiones 6 y 7.	9
3.3 Incorporar imágenes del satélite LANDSAT.	12
3.4 Incorporar imágenes del satélite RapidEye	13
3.5 Desarrollar nuevas funciones para la consulta de la información.	14
3.5.1 Selección de imágenes, modelos de elevación u ortofotos mediante un polígono definido por el usuario, y selección de múltiples satélites	14
3.5.2 Selección de imágenes tomadas en un periodo de tiempo (día – mes inicial y día – mes final), para un periodo de años	22
3.5.3 Despliegue de la vista previa de una imagen sobre el mapa del país, cálculo de áreas y distancias, y definición de transparencia.	22
4. RESULTADOS	55
5. CONCLUSIONES Y RECOMENDACIONES	57
6. BIBLIOGRAFÍA	58

ÍNDICE DE FIGURAS

<i>Figura 3. 1 Pares de coordenadas que definen un cuadrante del área cubierta por una imagen</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura 3. 2 Código usado para copiar los archivos que componen la imagen.....</i>	6
<i>Figura 3. 3 Código usado para validar que los archivos de una imagen se hayan copiado correctamente a la unidad de almacenamiento.....</i>	6
<i>Figura 3. 4 Código usado para validar la correcta creación del cuadrante de una imagen</i>	7
<i>Figura 3. 5 Código usado para llamar a la función de borrado de un cuadrante. ¡Error! Marcador no definido.</i>	
<i>Figura 3. 6 Función para borrado del cuadrante de una imagen</i>	8
<i>Figura 3. 7 Código para verificación de la existencia de una imagen, previo a su inserción a la base de datos</i>	9
<i>Figura 3. 8 Área geográfica y el año al que corresponden las imágenes SPOT 4 incorporadas al geoportal...</i>	10
<i>Figura 3. 9 Área geográfica y el año al que corresponden las imágenes SPOT 5 incorporadas al geoportal...</i>	10
<i>Figura 3. 10 Área geográfica y el año al que corresponden las imágenes SPOT 6 incorporadas al geoportal.</i>	11
<i>Figura 3. 11 Área geográfica y el año al que corresponden las imágenes SPOT 7 incorporadas al geoportal.</i>	11
<i>Figura 3. 12 Área geográfica y el año al que corresponden las imágenes SPOT 7 incorporadas al geoportal.</i>	12
<i>Figura 3. 13 Cobertura actual de ortofotos disponible en el geoportal.....</i>	13
<i>Figura 3. 14 Interfaz de usuario para la consulta de imágenes, con la opción “Polígono” y selección de múltiples satélites.....</i>	15
<i>Figura 3. 15 Ejemplo del uso de la opción “Polígono” y resultados de una consulta</i>	15
<i>Figura 3. 16 Ejemplo de resultados de una consulta de imágenes.....</i>	21
<i>Figura 3. 17 Creación de un dataset de mosaico.....</i>	23
<i>Figura 3. 18 Agregar rásteres al dataset de mosaico.....</i>	24
<i>Figura 3. 19 Configuración del entorno para agregar rásteres</i>	25
<i>Figura 3. 20 Configuración del procesamiento en paralelo</i>	26
<i>Figura 3. 21 Tabla de atributos del dataset de mosaico.....</i>	27
<i>Figura 3. 22 Archivo en excel de los atributos a insertar.....</i>	29
<i>Figura 3. 23 Creación de la unión para obtener los registros coincidentes</i>	29
<i>Figura 3. 24 Publicación de un servicio en ArcGIS Server</i>	30
<i>Figura 3. 25 Analizar el dataset de mosaico.....</i>	31
<i>Figura 3. 26 Interfaz principal del visor de imágenes</i>	32
<i>Figura 3. 27 Ubicación geográfica de los 33 sitios seleccionados (Fuente: CONABIO).....</i>	49
<i>Tabla 1 cuencas que tienen presencia espacial en el manglar</i>	54
<i>Figura 4. 1 Ventana de diálogo principal de la nueva interfaz del geoportal.....</i>	56
<i>Figura 4. 2 Resultados obtenidos mediante búsqueda por punto</i>	57

RESUMEN EJECUTIVO

Este informe describe la metodología y los resultados obtenidos dentro del proyecto denominado “Geoportal para consulta del acervo institucional de información geográfica, incorporación de nuevas imágenes y funciones”, el cual tuvo como objetivos principales la incorporación de imágenes de los satélites LANDSAT, SPOT y RapidEye, el desarrollo de nuevas funciones para la consulta de información. y determinar las cuencas hidrológicas asociadas a 50 manglares identificados para formar parte del Plan Nacional de Restauración y Reforestación de Manglar (PNRRM) y sus cuencas hidrológicamente conectadas, y realizar 10 estudios para la determinación de reservas potenciales de agua para el medio ambiente en cuencas costeras asociadas a manglares.

Dentro de los principales resultados del proyecto se puede mencionar la carga de 489 imágenes SPOT, 173 imágenes LANSAT, 5,250 ortofoto s de 2 metros de resolución y 319 con resolución de 1.5 metros, y la carga de imágenes RapidEye del periodo 2013 - 2016.

Como parte de las nuevas funciones para la consulta de información, se desarrolló la opción para permitir la selección de múltiples satélites al momento de consultar las imágenes, el cálculo de área y distancias, y búsqueda mediante un polígono definido por el usuario. Se desarrolló una nueva versión del geoportal que usa un dataset de mosaico, lo cual hace más rápida la consulta. Para ello, se actualizó el software ArcGIS Server y se adquirió la extensión Imagen for Server.

Finalmente, se determinaron las cuencas hidrológicas asociadas a 33 manglares identificados para formar parte del (PNRRM) y sus cuencas hidrológicamente conectadas y se realizaron diez estudios para la determinación de reservas potenciales de agua para el medio ambiente en cuencas costeras asociadas a manglares. Se realizaron solo 33 porque éstos fueron los que la CONABIO consideró (posterior al inicio de este proyecto), como prioritarios.

1. INTRODUCCIÓN

La Subcoordinación de Planeación Hídrica, de la Coordinación de Hidrología de este Instituto, cuenta con un acervo digital de información satelital y cartográfica de cobertura nacional en diferentes escalas que ocupa un volumen aproximado de 20 TB, el cual se incrementa constantemente. Esta información constituye un insumo básico en gran parte de los proyectos desarrollados en las áreas técnicas del Instituto, por lo que se ha manifestado la necesidad de contar con la infraestructura especializada que permita el almacenamiento, organización, consulta y recuperación de dicha información en línea por parte de la comunidad usuaria del IMTA.

Para atender esta necesidad, durante 2014 se desarrolló el proyecto “Geoportal para consulta del acervo institucional de información geográfica” cuyos resultados fueron la adquisición del equipo de cómputo para el almacenamiento de la información, el desarrollo de la primera versión de un sistema de consulta y transferencia de información, así como su implantación en la Intranet del Instituto.

Posteriormente, en el año 2015 se desarrolló una segunda parte del proyecto, en donde se realizó la adecuación del sitio para colocar el equipo de cómputo que albergaría el Geoportal de imágenes del territorio, se instaló y configuró dicho equipo, se realizaron pruebas de funcionamiento y se incorporó otra información relevante, compuesta por modelos digitales de elevación del terreno generados a través de la tecnología LIDAR y fotos aéreas.

Durante la operación del Geoportal por parte de los especialistas del Instituto se ha detectado la necesidad de contar con otras funciones de consulta de la información, como el cálculo de áreas y distancias. Además, existen imágenes del acervo que no han sido cargadas al banco de datos del geoportal, algunas de ellas son de varios años atrás y otras son más recientes. Se considera que es importante ponerlas a disposición de los especialistas, ya que aportan información valiosa sobre la situación de la superficie terrestre en diferentes momentos, lo cual es de utilidad para los proyectos que se realizan en el Instituto. Finalmente, era también necesario realizar mejoras al protocolo de carga de las imágenes para asegurar la consistencia de la información.

Debido a lo anterior, se planteó el desarrollo de este proyecto, el cual estuvo enfocado a realizar la carga de imágenes SPOT y LANDSAT que faltaban por dar de alta al geoportal, a desarrollar nuevas funciones de consulta y a realizar mejoras al protocolo de carga de imágenes.

Por otro lado, el Geoportal contiene actualmente sólo información raster compuesta por modelos digitales de elevación, imágenes de satélite y fotos aéreas. Se ha expresado también por parte de los especialistas del Instituto la

necesidad de disponer también de información vectorial que pueda ser útil para el desarrollo de los proyectos. Uno de los temas de mayor interés actual es el de los manglares. Estos ecosistemas prestan una gran cantidad de servicios ambientales, siendo algunos de los principales: la filtración de los sedimentos y las sustancias contaminantes que llegan al mar desde ríos lejanos, amortiguamiento de las inundaciones (absorbiendo y almacenando el agua), y la reducción significativa del impacto de marejadas, huracanes y tsunamis, en algunos casos, además de que almacena cinco veces más carbono que los bosques tropicales.

En relación al tema de los manglares, en 2016, por instrucciones presidenciales, la SEMARNAT debía establecer el Plan Nacional de Restauración y Reforestación de Manglar (PNRRM) con la finalidad de preservar los servicios ambientales que estos proporcionan. Para integrar el PNRRM, desde principios de ese año se realizaron reuniones de un grupo de dependencias lideradas por CONABIO, entre las cuales se invitó al IMTA, para llevar a cabo la “Identificación de manglares alterados a incorporar en el PNRRM”. En ese momento se identificaron alrededor de 50 manglares de los cuales se requería hacer su identificación de reserva potencial de agua (RPA), que permitiera definir la sustentabilidad del flujo hídrico que lo alimenta. En una primera etapa se debía trabajar un grupo piloto de 10 cuencas hidrológicas que permitiera establecer una metodología que se pudiera generalizar para el resto de las cuencas. Esta tarea le fue asignada al IMTA.

En virtud de todo lo anterior, se consideró conveniente incluir en este proyecto una actividad para determinar las cuencas hidrológicas asociadas a los 50 manglares identificados para formar parte del PNRRM y sus cuencas hidrológicamente conectadas, realizar un estudio de factibilidad para establecer una RPA en ellos y realizar 10 estudios para la determinación de reservas potenciales de agua para el medio ambiente en cuencas costeras asociadas a manglares.

Cabe aclarar que al inicio del proyecto se habían identificado 50 manglares identificados para formar parte del PNRRM, sin embargo, posteriormente la CONABIO lo redujo a 33, que es el número de manglares que se reporta en este informe.

2. OBJETIVOS

Los objetivos del proyecto fueron los siguientes:

Incorporar conjuntos de imágenes de los satélites LANDSAT, SPOT, y Rapideye; y agregar nuevas funciones para la consulta de la información Determinar las cuencas hidrológicas asociadas a 50 manglares identificados para formar parte del Plan Nacional de Restauración y Reforestación de Manglar y sus cuencas hidrológicamente conectadas, y realizar 10 estudios para la determinación de

reservas potenciales de agua para el medio ambiente en cuencas costeras asociadas a manglares.

3. METODOLOGÍA

Las actividades que se consideraron para alcanzar los objetivos planteados en este proyecto fueron las siguientes:

1. Mejorar el protocolo para incorporar nuevas imágenes al geoportal.
2. Incorporar imágenes del satélite SPOT, de las versiones 6 y 7.
3. Incorporar imágenes del satélite LANDSAT.
4. Incorporar imágenes del satélite RapidEye.
5. Desarrollar nuevas funciones para la consulta de la información
6. Determinar las cuencas hidrológicas asociadas a 50 manglares identificados para formar parte del Plan Nacional de Restauración y Reforestación de Manglar (PNRRM) y sus cuencas hidrológicamente conectadas, y realizar diez estudios para la determinación de reservas potenciales de agua para el medio ambiente en cuencas costeras asociadas a manglares.
7. Elaborar un artículo de divulgación.
8. Elaborar documentación.

A continuación se describe lo realizado en cada actividad y los resultados obtenidos.

3.1 Mejorar el protocolo para incorporar nuevas imágenes al geoportal.

El protocolo que ya se tenía para incorporar imágenes nuevas al geoportal tenía las siguientes deficiencias:

- a) No verificaba que las coordenadas de las esquinas del cuadrante o polígono que define el área cubierta por la imagen estuvieran especificadas en el orden correcto.
- b) El protocolo básicamente realiza 2 acciones cuando se inserta una nueva imagen al geoportal: inserta los atributos de ella a la tabla que contiene el catálogo de imágenes e inserta a una capa de polígonos el cuadrante que define el área cubierta por la imagen. Si llegaba a ocurrir un error al momento de insertar el cuadrante no se eliminaba el registro insertado

previamente en el catálogo de imágenes, lo cual daba lugar a inconsistencias.

Para corregir la primera deficiencia se establecieron reglas que debían cumplir las coordenadas de las esquinas del cuadrante para considerarlas correctas. Si se tiene un conjunto de 4 pares de coordenadas que definen un cuadrante, como las mostradas en la figura 3.1, en donde:

X_1, Y_1 son las coordenadas de la esquina superior izquierda

X_2, Y_2 son las coordenadas de la esquina superior derecha

X_3, Y_3 son las coordenadas de la esquina inferior izquierda

X_4, Y_4 son las coordenadas de la esquina inferior derecha

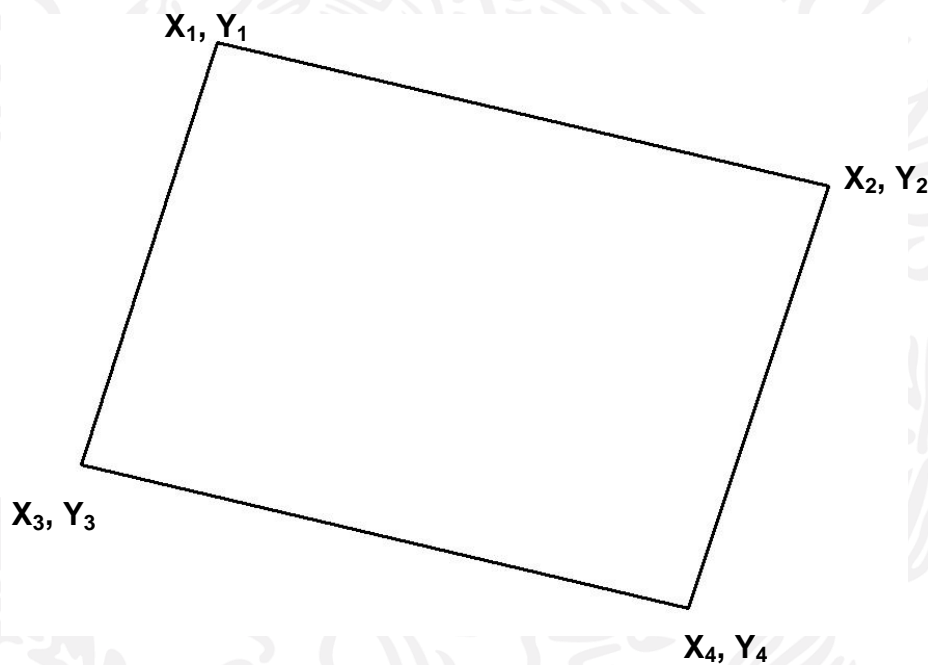


Figura 3. 1 Pares de coordenadas que definen un cuadrante del área cubierta por una imagen

Las reglas que deben cumplirse son las siguientes:

- a) $X_2 > X_1$
- b) $X_2 > X_3$
- c) $Y_2 > Y_4$

- d) $Y_1 > Y_3$
- e) $X_4 > X_3$
- f) $X_4 > X_1$

Se modificó el código en el programa de cómputo del protocolo para incorporar esas reglas.

Para corregir la segunda deficiencia se agruparon las líneas de código que realizan las acciones de inserción de atributos, copia de archivos e inserción de un cuadrante dentro de un bloque para considerar a todas ellas como una transacción. Una transacción, en el ámbito de las bases de datos, es un conjunto de órdenes o instrucciones que se ejecutan formando una unidad de trabajo, es decir, en forma indivisible o atómica. .

Si una de las órdenes o instrucciones que componen la transacción no se ejecuta correctamente, los cambios realizados a la base de datos por las instrucciones previas se anulan y la transacción se cancela. Este proceso de anulación de cambios para regresar la base de datos al estado en el que se encontraba antes de la transacción se denomina técnicamente “Rollback”. Por el contrario, si todas las instrucciones que componen la transacción se realizan correctamente, los cambios a la base de datos se hacen permanentes. Esta acción se realiza con un comando denominado “Commit”.

El programa de cómputo que permite ingresar nuevas imágenes a la base de datos realiza esto en 3 pasos:

- a) Copia los archivos que componen la imagen al directorio de la unidad de almacenamiento (NAS).
- b) Inserta a la capa de polígonos correspondiente el cuadrante de la imagen.
- c) Inserta los atributos de la imagen a la tabla correspondiente en la base de datos.

Como ya se mencionó, el código del programa se reorganizó para dejar estas operaciones como una transacción. El código que se emplea para copiar los archivos se muestra en la figura 3.2.

```
//copiar archivos
foreach (string archivo in archivos)
{
    //no copiar el csv
    if (Path.GetExtension(archivo) == ".csv")
    {

    }
    else
    {
        File.Copy(Path.Combine(path, archivo), Path.Combine(directorio, archivo));
    }
}

//crear cuadrante
CrearCuadrante cuadrantes = new CrearCuadrante(cuadranteSatelite, coordcuad, idEscena);
```

Figura 3. 2 Código usado para copiar los archivos que componen la imagen.

Posteriormente, se valida que los archivos que componen la escena se hayan copiado de forma correcta. Esto se hace con código mostrado en la figura 3.3.

```
//validar si se copiaron los archivos
bool banderacopiar = true;
foreach (string archivo in archivos)
{
    if (File.Exists(Path.Combine(directorio, archivo)))
    {

    }
    else
    {
        banderacopiar = false;
    }
}
}
```

Figura 3. 3 Código usado para validar que los archivos de una imagen se hayan copiado correctamente a la unidad de almacenamiento

Si los archivos se copiaron correctamente, el programa procede a crear el cuadrante de la imagen en la capa geográfica correspondiente y valida también que se haya hecho de forma correcta. La figura 3.4 muestra el código usado para la validación.

```

case "SPOT 6":
{
    string capa = "sde.Cuadrantes_SPOT_6";
    retcount = cuadExists(pkIdCuadrante, capa);
    if (retcount == 0)
    {
    }
    else
    {
        secreocudad = true;
    }
    break;
}

```

Figura 3. 4 Código usado para validar la correcta creación del cuadrante de una imagen

Si los 2 pasos anteriores se realizaron bien, el programa procede a realizar el tercer paso que consiste en la inserción a la base de datos de los atributos que describen a la imagen y los nombres de los archivos que la componen. Finalmente, si el tercer paso se realizó bien, el programa ejecuta una instrucción COMMIT para hacer permanentes los cambios a la base de datos.

Dado que las 3 operaciones son parte de una transacción, si hay un error en alguno de ellas el programa procede a deshacer los cambios que se hayan hecho previamente a la base de datos y a borrar los resultados de las operaciones previas también. Para realizar esto, se elaboró una función que borra los archivos correspondientes a una escena y otra que borra el cuadrante de la imagen. La 3.5 muestra el código que hace la llamada a la función para borrar un cuadrante cuyos argumentos son dos: la capa geográfica que contiene el cuadrante que se desea borrar y el identificador de éste. La figura 3.6 muestra la función que realiza el borrado.

```

case "SPOT 6":
{
    string capa = "sde.Cuadrantes_SPOT_6";
    retcount = cuadExists(pkIdCuadrante, capa);
    if (retcount == 0) { }
    else
    {
        DeleteCuadrante cuadrantes = new DeleteCuadrante(cuadranteSatelite, pkIdCuadrante);
    }
    break;
}

```

Figura 3. 5 Código usado para llamar a la función de borrado de un cuadrante

```
public void DeleteRows(IFeatureClass featureClass, string condicion)
{
    IQueryFilter queryFilter = new QueryFilterClass();
    queryFilter.WhereClause = "Pk_Id_Cuadrante = '" + condicion + "'";

    // Use IFeatureClass.Update to populate IFeatureCursor.
    IFeatureCursor updateCursor = featureClass.Update(queryFilter, false);
    IFeature feature = null;

    try
    {
        while ((feature = updateCursor.NextFeature()) != null)
        {
            updateCursor.DeleteFeature();
        }
    }
    catch (COMException comExc)
    {
        // Handle any errors that might occur on NextFeature().
        Console.WriteLine("Message: {0}", comExc.Message);
    }

    // If the cursor is no longer needed, release it.
    Marshal.ReleaseComObject(updateCursor);
}
}
```

Figura 3. 6 Función para borrado del cuadrante de una imagen

Adicionalmente, se realizaron cambios en el protocolo para verificar si una imagen ya existe en la base de datos, antes de proceder a insertarla. Para ello, se usó una llave primaria compuesta por el satélite, versión de éste, fecha y hora de toma de la imagen, y modo espectral de la misma. El protocolo ya tenía una función para verificar la existencia de una imagen pero no tomaba en cuenta el modo espectral. La figura 3.7 muestra el código que realiza esta verificación.

```

Actualizar.aspx.cs*
Actualizar
- btnActualizar_Click(object sender, EventArgs e)

cmd.CommandText = @"SELECT COUNT(*) FROM sde.SATELITEVERSION_SENSOR as sv, sde.SATELITE as s, sde.HECHOS_ESCENA_DE_SATELITE as hes, sde.MODOESPECTRAL as me,
sde.SENSOR as sen, sde.SENSOR_MODOESPECTRAL as senmod, sde.ARCHIVOESCENA_MODOESPECTRAL_NIVELDEPROCESAMIENTO as amn
WHERE sv.s.Pk_Id_Satellite = s.Pk_Id_Satellite
and s.Nombre+ '' + ISNULL(cast(Numero as CHAR(2)), '') = @satelite
and hes.Fecha_z = @fecha_z
and hes.Hora_z = @hora_z
and me.ModoEspectral = @modoespectral
and sv.s.Pk_Id_Sensor = sen.Pk_Id_Sensor
and sen.Pk_Id_Sensor = senmod.Pk_Id_Sensor and
senmod.Pk_Id_NodoEspectral = me.Pk_Id_ModoEspectral
and sv.s.Pk_Id_SatelliteVersion_Sensor = hes.Pk_Id_SatelliteVersion_Sensor
and me.Pk_Id_ModoEspectral = amn.Pk_Id_ModoEspectral
and amn.Pk_Id_Hechos_escena_de_satellite = hes.Pk_Id_Hechos_escena_de_satellite";

cmd.Parameters.AddWithValue("@satelite", registro[satelite]);
cmd.Parameters.AddWithValue("@fecha_z", registro[fecha]);
cmd.Parameters.AddWithValue("@hora_z", registro[hora]);
cmd.Parameters.AddWithValue("@modoespectral", registro[modoespectral]);

n_registro++;
string exit_message = "[" + n_registro + "];

int ret = 0;
ret = Convert.ToInt32(cmd.ExecuteScalar());

if (ret > 0)
{
    existe = true;
    n_registro--;
    n_error++;
}
    
```

Figura 3.7 Código para verificación de la existencia de una imagen, previo a su inserción a la base de datos

3.2 Incorporar imágenes del satélite SPOT, de las versiones 6 y 7.

En esta actividad se analizaron un total de 489 imágenes provenientes del satélite SPOT. Inicialmente se consideró dar de alta imágenes de las versiones 6 y 7, sin embargo, dentro del conjunto de imágenes que fueron proporcionadas se encontraron algunas provenientes de las versiones 4 y 5, las cuales se dieron de alta también. El cuadro mostrado abajo indica el número de imágenes de cada versión que se incorporaron al geoportal.

Versión del Satélite SPOT	Número de imágenes incorporadas a geoportal
4	2
5	268
6	130
7	89
Total	489

Cuadro 1. Número de imágenes de cada versión del satélite SPOT que fueron dadas de alta en el geoportal.

Las figuras 3.8, 3.9, 3.10 y 3.11 muestran el área geográfica y el año al que corresponden las imágenes SPOT que se incorporaron al geoportal.

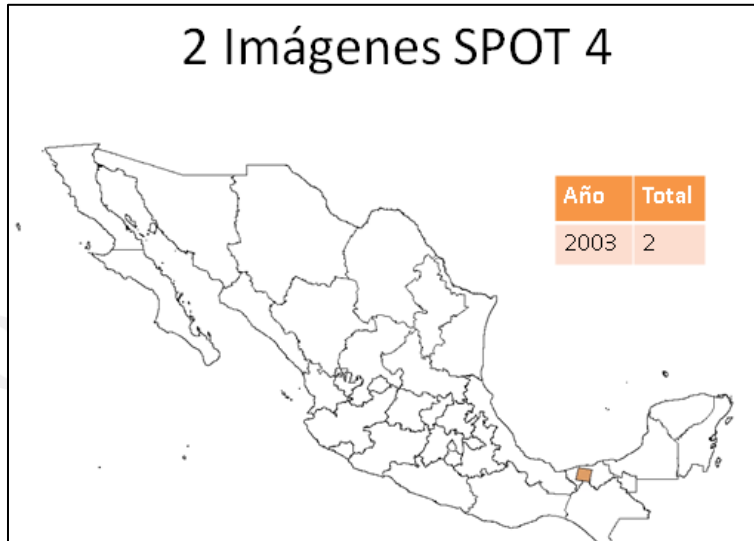


Figura 3. 8 Área geográfica y el año al que corresponden las imágenes SPOT 4 incorporadas al geoportal

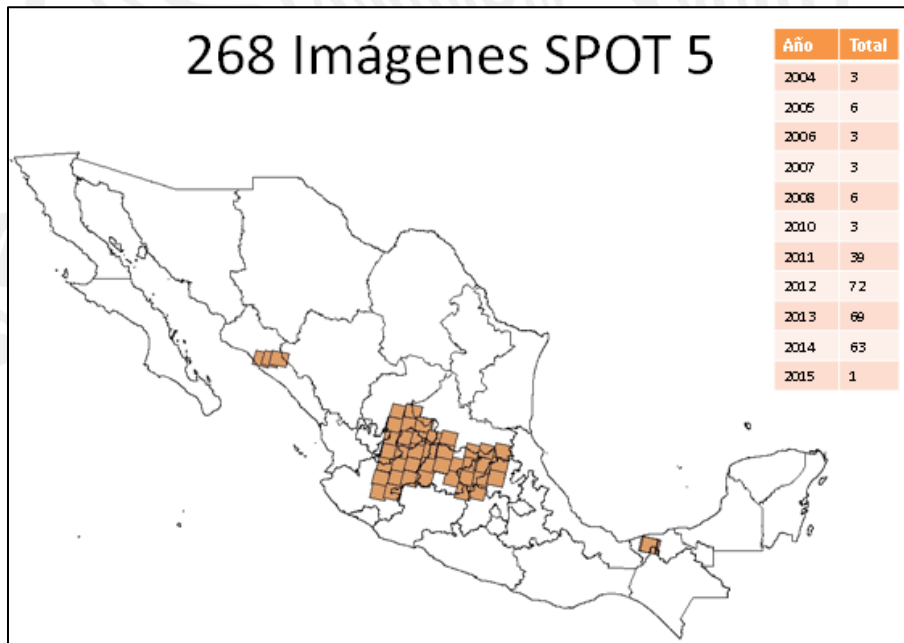


Figura 3. 9 Área geográfica y el año al que corresponden las imágenes SPOT 5 incorporadas al geoportal

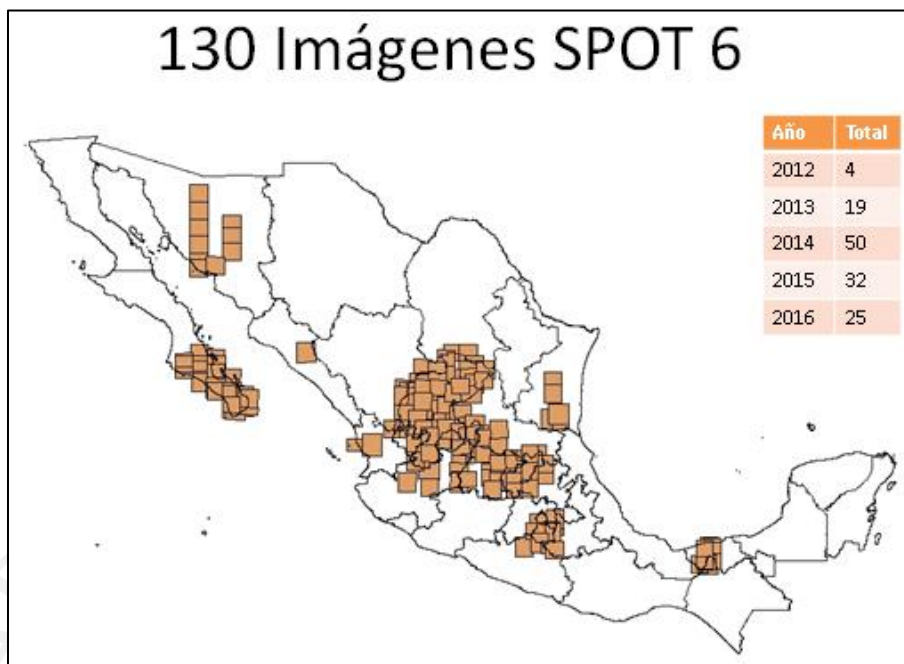


Figura 3. 10 Área geográfica y el año al que corresponden las imágenes SPOT 6 incorporadas al geoportal

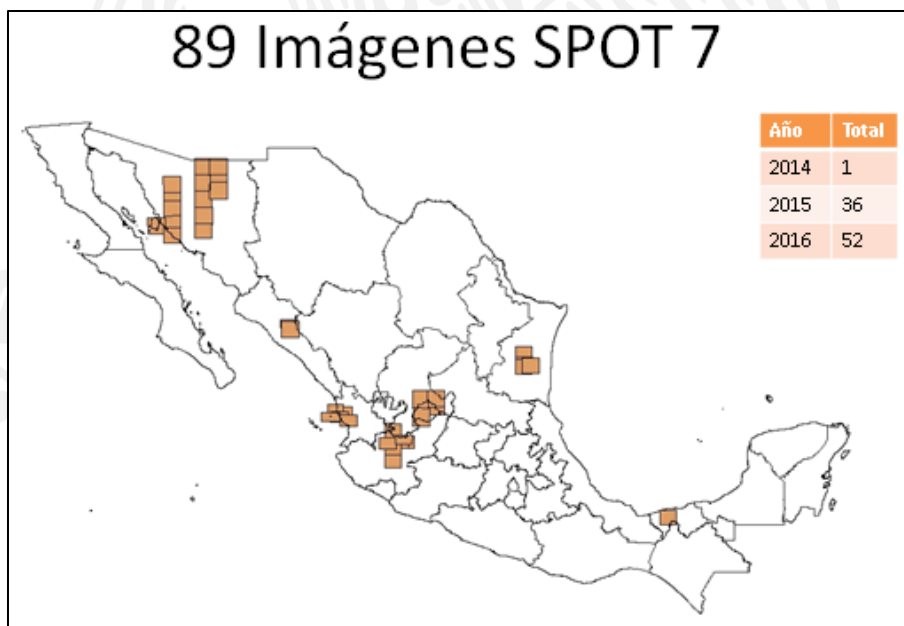


Figura 3. 11 Área geográfica y el año al que corresponden las imágenes SPOT 7 incorporadas al geoportal

3.3 Incorporar imágenes del satélite LANDSAT.

En esta actividad se analizaron 173 imágenes provenientes del satélite LANDSAT 5. Se ejecutaron los programas para extraer los atributos de las imágenes como la fecha, la hora y porcentaje de nubes, entre otros de interés. Se extrajeron los archivos de vista previa y se generaron los archivos comprimidos para aquellas imágenes que no contaban con él. La figura 3.12

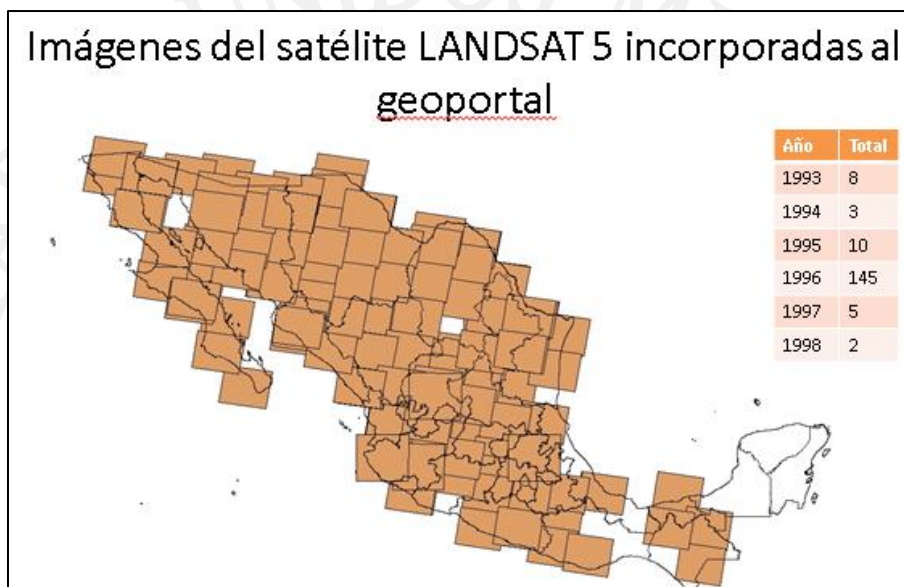


Figura 3. 12 Área geográfica y el año al que corresponden las imágenes SPOT 7 incorporadas al geoportal

Adicionalmente, dentro de esta actividad se analizaron los archivos de 5,250 ortofotos con resolución 1.5 m., y de 319 con resolución de 2 m. Se generaron los cuadrantes del área cubierta por cada ortofoto y los archivos que componen cada una. Posteriormente, estas ortofotos se dieron de alta en la base de datos del geoportal. Cabe mencionar que el año pasado se incorporó al geoportal sólo una parte de esta información, que cubría principalmente la parte sur del país.. Dada utilidad de las ortofotos, se consideró conveniente dedicar tiempo para el análisis y carga de las que faltaban por incorporarse al geoportal. La figura 3.13 muestra la cobertura actual que se tiene de ortofotos en este sistema.



Figura 3. 13 Cobertura actual de ortofotos disponible en el geoportal

3.4 Incorporar imágenes del satélite RapidEye

En esta actividad se envió una solicitud al Director de la Comisión Nacional Forestal (CONAFOR), el Ing. Jorge Rescala Pérez, en donde se le pide proporcione al IMTA las coberturas nacionales de imágenes RapidEye correspondientes a los años 2013, 2014, 2015 y 2016.

La respuesta a dicha solicitud fue enviada por la Ing. Mayra Ramírez Salgado, Jefe del Departamento de Geomática de la Gerencia del Sistema Nacional de Monitoreo Forestal de la CONAFOR. La respuesta fue positiva, únicamente la Ing. Ramírez solicitó que se le enviara a las oficinas de la CONAFOR los medios de almacenamiento, con capacidad para 8 TB, para copiar las coberturas solicitadas. En la segunda semana de diciembre se enviaron a la CONAFOR 4 discos duros, que en total representan una capacidad de 10 TB, para copiar la información mencionada.

Se habló a la CONAFOR para confirmar que habían recibido los discos, lo cual fue así. Sin embargo, la Ing. Ramírez comentó que debido a la carga de trabajo que tenían, por ser fin de año, no sería posible que copiaran los datos en esos días, y

que lo harían comenzando el próximo año. Finalmente, las imágenes llegaron en el mes de enero de 2017 y correspondían a los años 2013, 2014, 2015 y 2016. Estas imágenes fueron cargadas a la base de datos del geoportal, usando la aplicación que se desarrolló para este propósito.

3.5 Desarrollar nuevas funciones para la consulta de la información.

En esta actividad se desarrollaron las siguientes nuevas funciones a la interfaz de consulta del geoportal:

- a) Selección de imágenes, modelos de elevación u ortofotos mediante un polígono definido por el usuario, y selección de múltiples satélites
- b) Selección de imágenes tomadas en un periodo de tiempo (día – mes inicial y día – mes final), para un periodo de años
- c) Despliegue de la vista previa de una imagen sobre el mapa del país, cálculo de áreas y distancias, y definición de transparencia.

A continuación se describe en forma breve cada función.

3.5.1 Selección de imágenes, modelos de elevación u ortofotos mediante un polígono definido por el usuario, y selección de múltiples satélites

Las opciones que se tenían anteriormente en el geoportal para definir el área para la cual se desea seleccionar imágenes u otra información raster eran: un punto, un archivo shapefile, un estado y una Región Hidrológico – Admva. Sin embargo, hay ocasiones en las que el usuario requiere definir a mano un polígono irregular y seleccionar las imágenes que se cubren ese polígono, ya sea parcial o totalmente. Debido a esa necesidad, se desarrolló la función que permite definir el polígono. Asimismo, se agregó a la interfaz la posibilidad de seleccionar múltiples satélites para la consulta de imágenes, dado que los usuarios también han expresado esta necesidad.

La figura 3.14 muestra la interfaz de usuario para la consulta de imágenes, en donde se presenta al usuario la opción llamada “Polígono”, dentro de los modos de búsqueda. Cuando el usuario selecciona esta opción, la interfaz habilita la acción de crear el polígono definiendo primero los vértices del mismo (dando un clic con el botón izquierdo del mouse) y cerrándolo posteriormente (dando doble clic con ese botón). Una vez definido el polígono, éste se visualiza en el mapa. Enseguida el usuario puede elegir los satélites para los cuales desea imágenes, el modo espectral, el periodo de tiempo, el nivel de corrección y el porcentaje de nubes. La figura 3.15 muestra un ejemplo de la definición de un polígono y el resultado de una consulta.

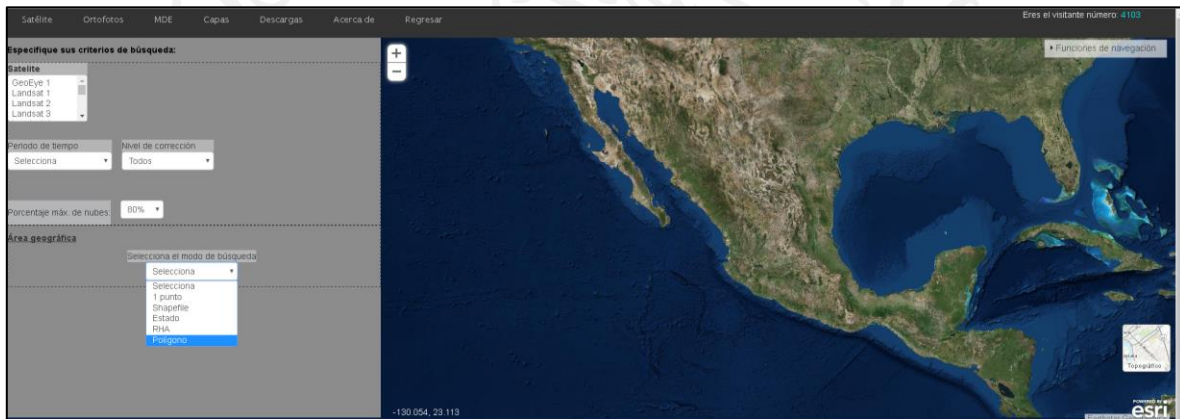


Figura 3. 14 Interfaz de usuario para la consulta de imágenes, con la opción “Polígono” y selección de múltiples satélites

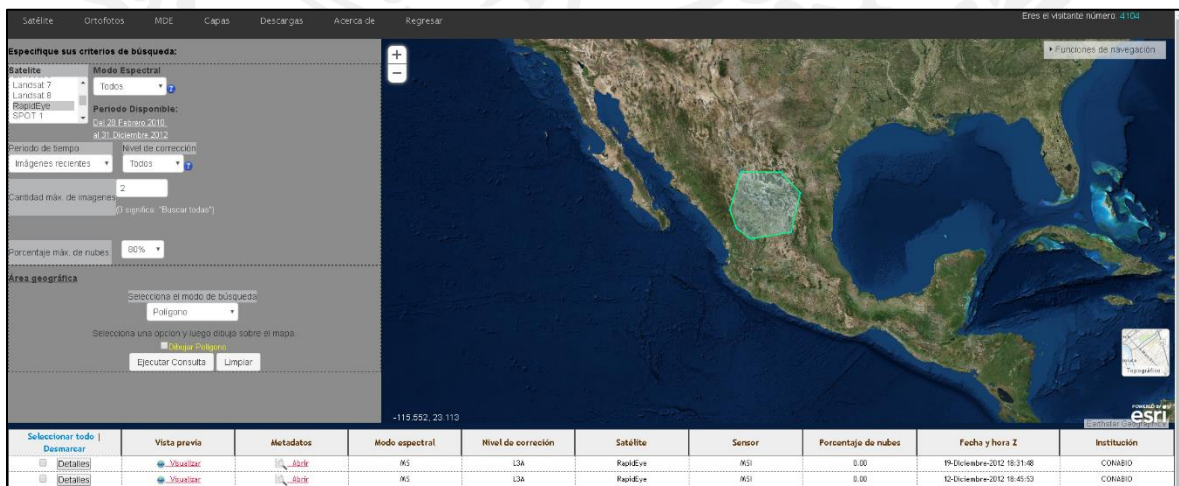


Figura 3. 15 Ejemplo del uso de la opción “Polígono” y resultados de una consulta

El código para implementar la opción del polígono definido por el usuario se elaboró en el lenguaje JavaScript, se hizo la clase PoligonoDibujado en C# y se reutilizaron algunos métodos de este último. La máscara que se muestra al usuario se realizó con HTML.

```
<div id="Dibujar_div" style="display:none;" runat="server" >
  <div id="info" >
    <div>Selecciona una opcion y luego dibuja sobre el mapa.</div>
    <input type='checkbox' name='change' id='Poligono' value='1000' runat="server" /><a><font color="yellow" size="2">Dibujar Poligono.</font></a><br />
    <input type="button" id="EjecutarPoligono" value="Ejecutar Consulta" class="btn btn-default"/>
    <input type="button" id="LimpiarPoligono" value="Limpiar" class="btn btn-default" /><br />
  </div>
</div>
```

En la función CambioMetodo se agregó código para visualizar el módulo del polígono y ejecutarlo, y también para deshabilitar los otros módulos y que no interfieran con las acciones de esta opción. El código se muestra abajo.

```
}
if (valor == 6) { //Opcion Poligono
document.getElementById("DynamicGrid").innerHTML = "";
document.getElementById("piee").innerHTML = '';
document.getElementById('SpotKJ').style.display = 'none';
document.getElementById('Spot_Punto').style.display = 'none';
document.getElementById('Spot_Shape').style.display = 'none';
document.getElementById('nvoShape').style.display = 'none';
document.getElementById('shapepedir').style.display = 'none';
document.getElementById('Spot_Estado').style.display = 'none';
document.getElementById('Spot_RHA').style.display = 'none';
document.getElementById('Dibujar_div').style.display = 'block'; //Muestra los botones para consultar por poligono
document.getElementById('busq_landsat').style.display = 'none';
document.getElementById('busq_spot').style.display = 'block';
document.getElementById('nota_busqueda').style.display = 'none';
}
}
```

Por otro lado, se declararon variables para dar formato a los puntos y líneas del polígono, dentro de la función correspondiente a la carga y visualización de los objetos de la API de ArcObjects; esto se hizo con el fin de que visualizarlas de manera agradable. El código se muestra en el cuadro siguiente.

```

var markerSymbolVertex2 = new SimpleMarkerSymbol(//-----formato Punto
SimpleMarkerSymbol.STYLE_CIRCLE, 10,
new SimpleLineSymbol(SimpleLineSymbol.STYLE_SOLID,
new Color([224, 224, 224]), 1),
new Color([51, 255, 51, 0.75]));

var fillSymbol = new SimpleFillSymbol(SimpleFillSymbol.STYLE_SOLID, //-----formato Poligono
new SimpleLineSymbol(SimpleLineSymbol.STYLE_SOLID, //STYLE_DASHDOT
new dojo.Color([0, 255, 128]), 2), new dojo.Color([148, 239, 249, 0.25]));

var markerSymbolVertex = new SimpleMarkerSymbol(//-----formato Puntos del poligono
SimpleMarkerSymbol.STYLE_CIRCLE, 5,
new SimpleLineSymbol(SimpleLineSymbol.STYLE_SOLID,
new Color([7, 76, 166]), 1),
new Color([0, 191, 255, 0.75]));

```

También, se utilizó la función `addGraphic` para agregar el polígono al mapa y para deshabilitar la bandera que permite se realice el trazado de un polígono, una vez que se termina de generar uno.

```

function addGraphic(evt) { //Al terminar de dibujar poligono agregamos el poligono al mapa
document.getElementById('Poligono').checked = false; //desactivamos el poligono
document.getElementById('checkPoligono').checked = false; //paramos de dibujar Poligono
tb.deactivate();
map.enableMapNavigation();

var symbol;

if (evt.geometry.type === "line" || evt.geometry.type === "polyline") {
symbol = lineSymbol;
}
else {
symbol = fillSymbol;
}

map.graphics.add(new Graphic(evt.geometry, symbol));
}

```

Se reutiliza también la función “Limpiar” y se llama desde un objeto `dojo`.

```
on(dojo.byId("LimpiarPoligono"), "click", limpiar); //LimpiarPoligono
```

```

function limpiar() {
map.graphics.clear();
updateLayerVisibility();
document.getElementById('piee').innerHTML = '';
document.getElementById('DynamicGrid').innerHTML = '';
document.getElementById('Filtrado').innerHTML = '';
}

```

Por otro lado, se agregaron componentes `dojo` para hacer más rápida la ejecución de ciertas funciones y se utilizaron las funciones predefinidas de los componentes para hacer dinámica la funcionalidad.

```

on(dojو.byId("Poligono"), "click", checkPoligon); //Chekeamos para identificar que se esta generando poligono

//=====CONSTRUCCION=====POLYGONO
$("#Poligono").change(function () { //Nuevo polyono
    if (this.checked) {

        tb.activate(Draw.POLYGON); //comenzamos a dibujar poligono

    } else {
        document.getElementById('checkPoligono').checked = false; //Cuando ya no esta chekeado se coloca checkPoligono en false para desactivar los puntos
    }
});

var coordenadasPoli = "";
function checkPoligon() {
    map.graphics.clear();
    coordenadasPoli = "";
    if (!document.getElementById('checkPoligono').checked) {
        document.getElementById('checkPoligono').checked = true;
    }
}

function dibujaPoligono(evt) {
    if (document.getElementById('checkPoligono').checked) {
        coordenadasPoli = coordenadasPoli + ',' + bandera;
        map.graphics.add(new Graphic(evt.mapPoint, markerSymbolVertex)); //-----Agregamos punto al poligono
        // alert("Coordenadas: " + coordenadasPoli + "Bandera" + bandera); // se
    }
}

```

Adicionalmente, con código en lenguaje C# fue realizada la clase PoligonoDibujado, que por medio de unas funciones de AJAX (propia de JavaScript) pasa valores de la API hacia el código de C# donde se hace un proceso con la biblioteca de ArcObjects para realizar consultas a la geobase de datos y devolver los resultados al usuario. Los cuadros mostrados abajo presentan el código para realizar esto.

```

using System;
using System.Data;
using ESRI.ArcGIS.esriSystem;
using ESRI.ArcGIS.Geodatabase;
using ESRI.ArcGIS.Geometry;
using Newtonsoft.Json;

/// <summary>
/// Descripción breve de PoligonoDibujado
/// </summary>
public class PoligonoDibujado
{
    public PoligonoDibujado()
    {
        //
        // TODO: Agregar aquí la lógica del constructor
        //
    }
}

```

```
public string CrearShape(string coordenadas, string capa_geografica)
{
    if (ESRI.ArcGIS.RuntimeManager.ActiveRuntime == null)
    {
        ESRI.ArcGIS.RuntimeManager.BindLicense(ESRI.ArcGIS.ProductCode.EngineOrDesktop);//carga licencia en el codigo
    }
    ESRI.ArcGIS.RuntimeManager.Bind(ESRI.ArcGIS.ProductCode.Desktop);
    string s = "";
    IFields fields = new FieldsClass();
    IPoint point = new PointClass();
    IPoint pointFinal = new PointClass();
    IPoint splitPoint = new PointClass();

    IPointCollection4 pointCollection = new PolylineClass();
    object missing = Type.Missing;
    //IClone clone = pointCollection as IClone;
    //=====
    IGeometry mypoligono = new PolygonClass();

    //=====

    //poly=clone.Clone() as IPolygon;

    bool isSplitted;
    int newPartIndex;
    int newSegmentIndex;
    string[] No_Coordenadas = coordenadas.Split(',');

```

```
for (int i = 0; i < No_Coordenadas.Length-1; i=i+2)
{
    double x = Convert.ToDouble(No_Coordenadas[i + 1]);
    double y = Convert.ToDouble(No_Coordenadas[i + 2]);
    if (i == 0)
    {
        pointFinal.PutCoords(x, y);
    }
    point.PutCoords(x, y);
    pointCollection.AddPoint(point, ref missing, ref missing);

}
pointCollection.AddPoint(pointFinal, ref missing, ref missing);
//pointCollection.EnumVertices.Next;

IClone clone = pointCollection as IClone;
ESRI.ArcGIS.Geometry.IPolyline poly = clone.Clone() as IPolyline;

//poly.SplitAtPoint(point, true, false, out isSplitted, out newPartIndex, out newSegmentIndex);

ConexionGDB datos = new ConexionGDB();

```

En esta misma clase se tiene un método para intersectar los cuadrantes de cada satélite seleccionado.

```
IGeometryCollection geometryCollection = poly as IGeometryCollection;
String report = "\n";
for (int i = 0; i < geometryCollection.GeometryCount; i++)
{
    report = report + "Part : " + i;
    ISegmentCollection segmnetCollection = geometryCollection.get_Geometry(i) as ISegmentCollection;
    for (int j = 0; j < segmnetCollection.SegmentCount; j++)
    {
        report = report + " Segment : " + j;
        ISegment segment = segmnetCollection.get_Segment(j);
        report = report + "From Point X, Y = " + segment.FromPoint.X + ", " + segment.FromPoint.Y + "\n" +
            "From Point X, Y = " + segment.ToPoint.X + ", " + segment.ToPoint.Y + "\n";
    }
}

datos.metodo(capa_geografica);
s = Intersecta1(poly, datos.featureClassMalla);
return s;
}

```

```
public string Intersecta1(IPolyline point, IFeatureClass featureClassMalla)
{
    DataTable tabla = new DataTable("Datos");
    DataRow row;
    tabla.Columns.Add("id", typeof(String));
    //tabla.Columns.Add("path_row", typeof(String));
    tabla.Columns.Add("OID", typeof(String));
    String ShapeFieldName;
    IFeatureCursor cursor;
    IFeature feature;
    SpatialFilter pSpatialFilter = new SpatialFilter();

    pSpatialFilter.Geometry = point.Envelope; //pfeature.Shape//envelop; Shape
    pSpatialFilter.SpatialRel = esriSpatialRelEnum.esriSpatialRelIntersects;
    ShapeFieldName = featureClassMalla.ShapeFieldName;
    pSpatialFilter.GeometryField = featureClassMalla.ShapeFieldName;
    cursor = featureClassMalla.Search(pSpatialFilter, false);

    while ((feature = cursor.NextFeature()) != null)
    {
        int iod = feature.OID;
        int dato1 = feature.Fields.FindField("Pk_Id_Cuadrante");
        object id1 = feature.get_Value(dato1);
        //int path_row = feature.Fields.FindField("OBJECTID*");
        // object pathrow = feature.get_Value(path_row);
        row = tabla.NewRow();
        row["id"] = id1.ToString();
        //row["path_row"] = " "; //pathrow.ToString();
        row["OID"] = iod.ToString();
        tabla.Rows.Add(row);
    }
    string s = JsonConvert.SerializeObject(tabla);

    return s;
}
```

En esta parte, se reutilizó también la clase para la conexión a la base de datos, cuyo código se muestra en las figuras siguientes.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using ESRI.ArcGIS.Geodatabase;
using ESRI.ArcGIS.esriSystem;

/// <summary>
/// Descripción breve de ConexionGDB
/// </summary>
public class ConexionGDB
{
    public IFeatureClass featureClassMalla;
    public IFeatureClass featureClassPlantas;
    public IFeatureClass featureClassEstados;
    public IFeatureClass featureClassRHA;
    public IWorkspace tr;
    public ESRI.ArcGIS.Geodatabase.IFeatureWorkspace featureWorkspace;
    public ConexionGDB()
    {
        ConnectToTransactionalVersion("172.16.2.210", "sde:sqlserver:WIN-BAPISCV6JIK", "sa", "Imta2012", "Acervo_Geografico", "sde.DEFAULT");
        //ConnectToTransactionalVersion("172.16.2.210", "sde:sqlserver:172.16.2.210", "sa", "Imta2015", "Acervo_Geografico", "sde.DEFAULT");
    }

    public IWorkspace ConnectToTransactionalVersion(String server, String instance, String user, String password, String database, String version)
    {
        if (ESRI.ArcGIS.RuntimeManager.ActiveRuntime == null)
        {
            ESRI.ArcGIS.RuntimeManager.BindLicense(ESRI.ArcGIS.ProductCode.EngineOrDesktop);
        }
    }
}
```



```

IPropertySet propertySet = new PropertySetClass();
propertySet.SetProperty("SERVER", server);
propertySet.SetProperty("INSTANCE", instance);
propertySet.SetProperty("DATABASE", database);
propertySet.SetProperty("USER", user);
propertySet.SetProperty("PASSWORD", password);
propertySet.SetProperty("VERSION", version);

Type factoryType = Type.GetTypeFromProgID("esriDataSourcesGDB.SdeWorkspaceFactory");
IWorkspaceFactory workspaceFactory = (IWorkspaceFactory)Activator.CreateInstance
(factoryType);
tr = workspaceFactory.Open(propertySet, 0); //se crea el workfactory

featureWorkspace = (ESRI.ArcGIS.Geodatabase.IFeatureWorkspace)tr;

ESRI.ArcGIS.Geodatabase.QueryFilter queryfilter = new ESRI.ArcGIS.Geodatabase.QueryFilterClass();
featureClassEstados = featureWorkspace.OpenFeatureClass("Acervo_Geografico.SDE.Entidad_federativa_MGM_2010");
featureClassRHA = featureWorkspace.OpenFeatureClass("Acervo_Geografico.SDE.Region_hidrologico_administrativa_2010");
Funciones da = new Funciones();
da.pruebas(featureClassMalla);
return tr;
}
public void metodo(string valor)
{
featureClassMalla = featureWorkspace.OpenFeatureClass(valor);
}
}
    
```

Dentro del código de la opción “Polígono”, se envían, por medio de Ajax, las consultas a la base de datos mediante la clase “Consulta”, pasando por la clase WebService para llamar a cada uno de los métodos de la clase PoligonoDibujado, de esta forma se retornan los valores y se reciben. Para que estos valores sean leídos por JavaScript se convierten a formato JSON y así es como se visualiza la información en el mapa. La figura 3.16 muestra un ejemplo de los resultados de una consulta.

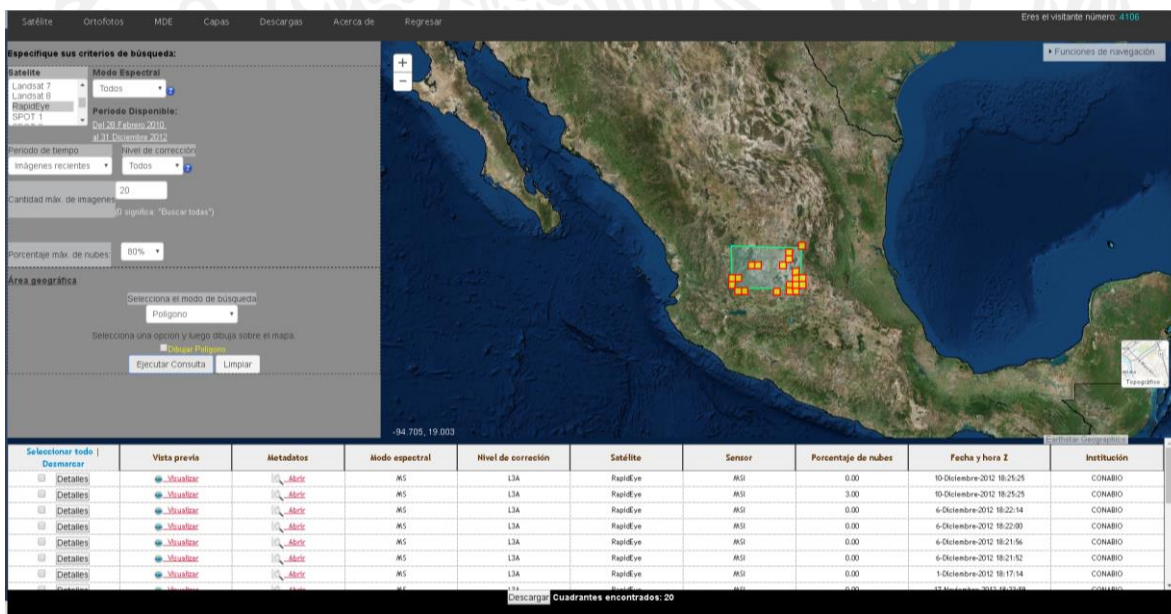


Figura 3. 16 Ejemplo de resultados de una consulta de imágenes.

3.5.2 Selección de imágenes tomadas en un periodo de tiempo (día – mes inicial y día – mes final), para un periodo de años

En esta actividad se mejoró el algoritmo de consulta de imágenes para un periodo de tiempo definido por el usuario. El algoritmo anterior daba resultados con algunas inconsistencias, las cuales fueron corregidas. El algoritmo actual permite al usuario definir un periodo de tiempo de interés, dado por un mes y día de inicio, y un mes y día de término, y un periodo de años, dado por un año inicial y un año final. Esto permite, por ejemplo, obtener las imágenes que están disponibles para una cierta época del año (por ejemplo, la época de lluvias) de todo un conjunto de años. Es posible también obtener las imágenes que están en un periodo que comprende años consecutivos (por ejemplo, del 1 de noviembre al 15 de febrero), también para un conjunto de años.

3.5.3 Despliegue de la vista previa de una imagen sobre el mapa del país, cálculo de áreas y distancias, y definición de transparencia.

Como parte de esta actividad se realizó una investigación para determinar cuáles herramientas de ArcGIS se requerían para implementar estas funciones. Se hizo una visita a las oficinas de SIGSA para platicar con personal técnico que desarrolla aplicaciones similares. Se estableció contacto también con personal del USGS de los Estados Unidos, vía correo electrónico y se recibió asesoría también de esta oficina.

La conclusión a la que se llegó fue que la forma más eficiente de almacenar, administrar y visualizar imágenes en ArcGIS es a través de datasets de mosaico (mosaic datasets). Por otro lado, para crear servicios de imágenes a partir de estos datasets se determinó que se requiere la extensión Image for Server, la cual no se tenía disponible en el IMTA. En virtud de lo anterior, y dado que existían recursos en otros proyectos para actualizar el software ArcGIS Server y adquirir la extensión mencionada, se procedió a elaborar los documentos para realizar esto. Finalmente, se actualizó el ArcGIS Server 10 que se tenía al nuevo producto ArcGIS for server, versión 10.4.1, y se adquirió la extensión Image for Server. Posteriormente, se procedió a implementar las funciones mencionadas. Cabe mencionar que se desarrolló una nueva interfaz para consulta de las imágenes, muy similar a la del USGS, ya que agregar el código de estas funciones a la interfaz que se tenía presentaba algunas dificultades técnicas, por la diferencia en

las versiones del ArcGIS Server. A continuación se explica que es un dataset de mosaico y la forma de implementarlo.

¿QUÉ ES UN DATASET DE MOSAICO ?. Es un modelo de datos dentro de una geodatabase que se utiliza para administrar una colección de datasets ráster (por ejemplo, imágenes de satélite o modelos de elevación) que se almacenan como un catálogo y se visualizan como un mosaico. Los datasets de mosaico tienen avanzadas funciones de procesamiento y consulta de información ráster, y también se pueden utilizar como origen para crear servicios de imágenes con el software ArcGIS for Server.

¿QUÉ VENTAJAS NOS BRINDA UN DATASET DE MOSAICO?. Un dataset de mosaico permite almacenar, administrar, ver y consultar colecciones de ráster y datos de imagen, de tamaño pequeño hasta muy grande.

A continuación se explica la forma de implementar un dataset de mosaico con el software ArcGIS.

PASO 1: IMPLEMENTACIÓN DEL DATASET DE MOSAICO PARA LA NUEVA INTERFAZ DEL GEOPORTAL

Se utilizó una geobase de archivos, denominada “VisorIMTA.gdb”, y ubicada en el servidor de mapas (172.16.2.212), dentro del directorio C:\Users\jaimevel\Documents\. La forma de crear un dataset de mosaico se muestra en la figura de abajo.

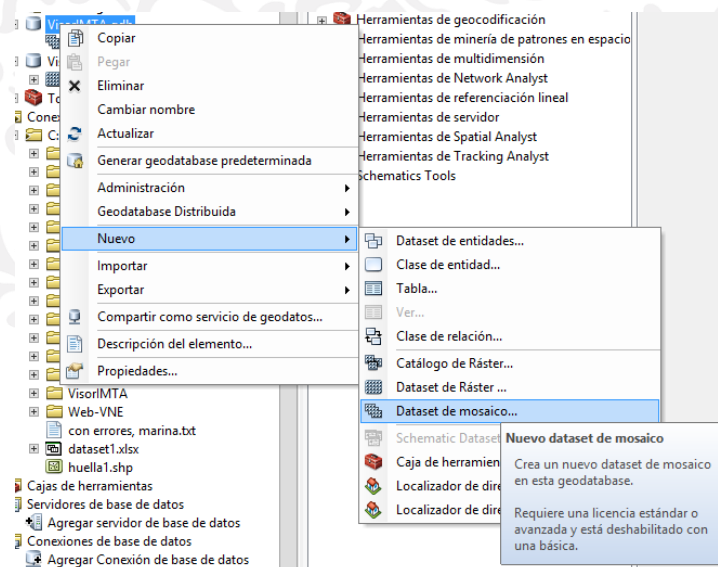


Figura 3. 17 Creación de un dataset de mosaico

ASPECTOS A CONSIDERAR

Una vez creado el dataset se deben crear los campos que se requieren para describirlo. En este caso se crearon campos para almacenar el porcentaje de nubes, modo espectral, etc. (especificados más adelante). Si no se crean los campos, éstos no se verán al momento de publicar el servicio de imágenes.

PASO 2.-AGREGAR RÁSTERES

Los rásteres, no son más que las vistas previas (quicklooks) de cada una de las escenas de imágenes, las cuales se agregan posteriormente al dataset de mosaico usando la aplicación ArcCatalog. Se utilizaron las vistas previas porque son más pequeñas y esto permite que se visualicen más rápido en la interfaz web:

NOTA IMPORTANTE: Para agregar rásteres e información a la tabla de atributos, se debe editar el dataset local, es decir, el dataset de mosaico contenido en la geobase, no desde el servicio publicado con ArcGIS for Server. Como ya se mencionó, la geobase se almacenó en el directorio C:\Users\jaimelve\\Desktop\VisorIMTA de la máquina virtual (usada como servidor de mapas) mencionada en el paso 1. Las dos figuras mostradas a continuación muestran la opción usada para agregar los rásteres y la ventana de diálogo que se despliega para ello.

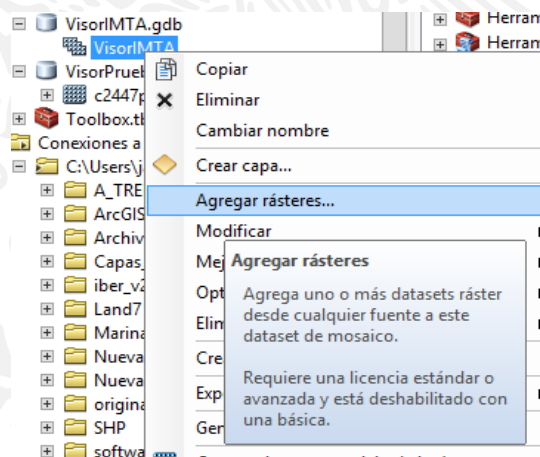


Figura 3. 18 Agregar rásteres al dataset de mosaico

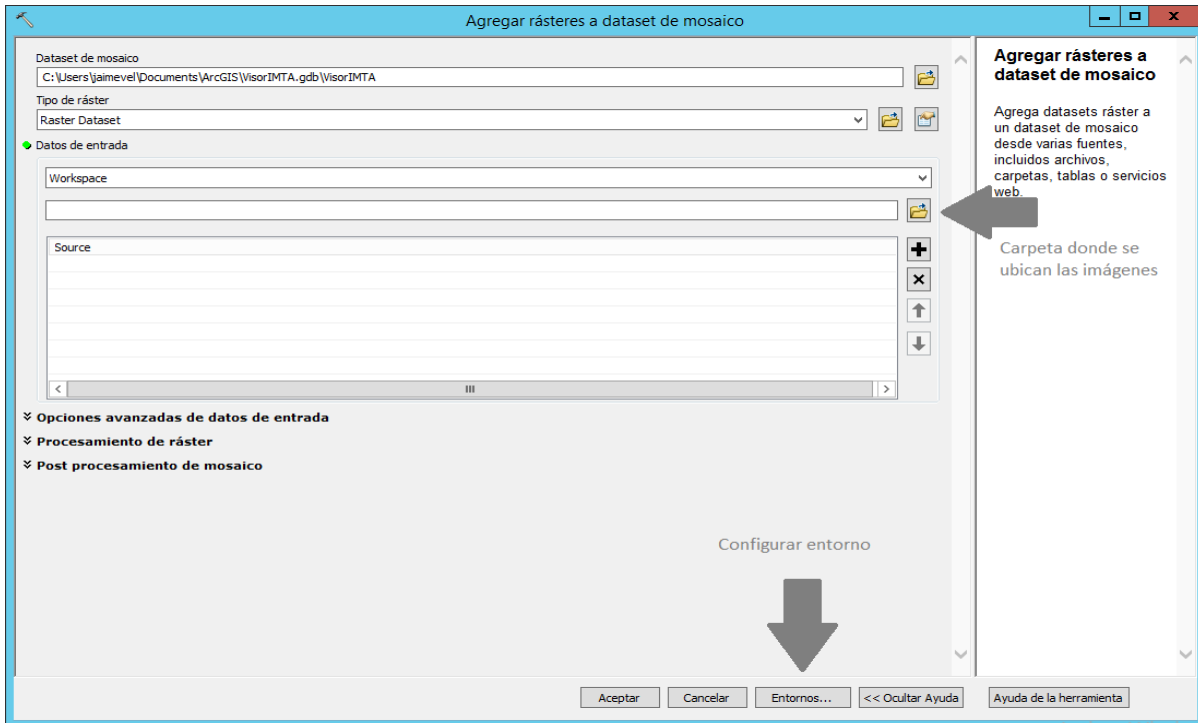


Figura 3. 19 Configuración del entorno para agregar rásteres

ASPECTOS A CONSIDERAR

- ✚ Antes de agregar los rásteres se configuró el entorno. En este caso sólo se configuró el factor de procesamiento en paralelo, como se muestra en la figura de abajo. De hecho, se recomienda que para cualquier herramienta de ArcToolBox que se use, el entorno se debe configurar así.

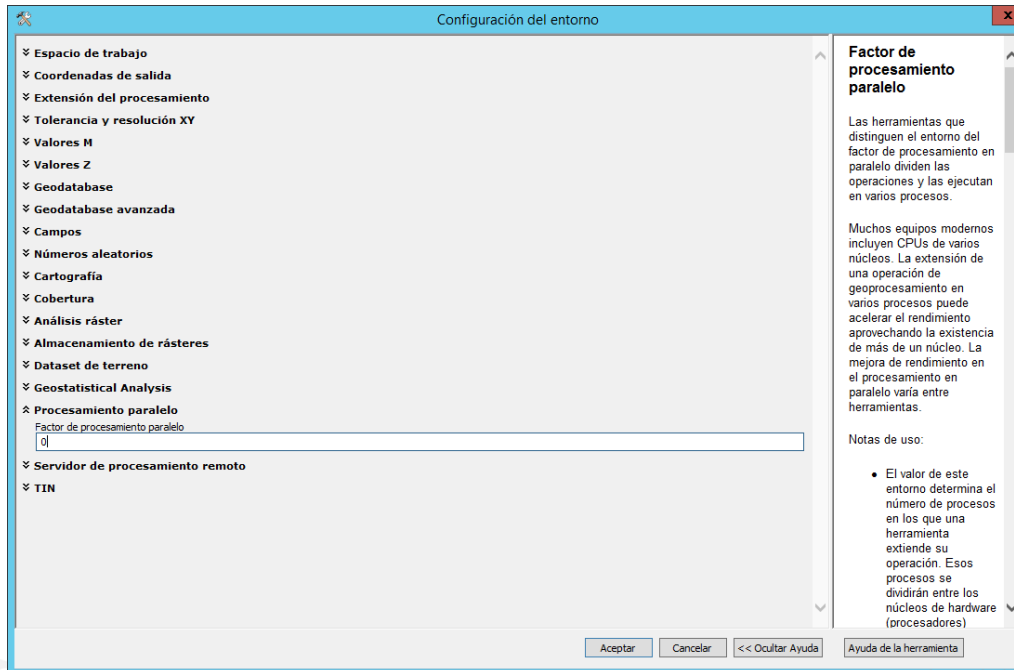


Figura 3. 20 Configuración del procesamiento en paralelo

Los rasters deben estar georreferenciados en WGS 1984, de lo contrario no se pueden agregar al dataset. Para reproyectar todos las vistas previas que se agregaron se utilizó el código en Python mostrado a continuación:

```
import os
import arcpy
from arcpy import env

#Variable para la ruta al directorio
path = 'C:\Visor\IMTA'
#Lista vacia para incluir los ficheros
lstFiles = []
#Lista con todos los ficheros del directorio:
lstDir = os.walk(path) #os.walk()Lista directorios y ficheros
#Crea una lista de los ficheros jpg que existen en el directorio y los incluye a la lista.
for root, dirs, files in lstDir:
    for fichero in files:
        (nombreFichero, extension) = os.path.splitext(fichero)
        if(extension == ".tif"):
            lstFiles.append(nombreFichero+extension)
```

```
arcpy.ProjectRaster_management(
"C:/CarpetaFuente/"+nombreFichero+extension,
"C:/CarpetaDestino/"+nombreFichero+extension,"C:/GCS_WGS_1984.prj",
"BILINEAR","#","#","#","#")
```

```
print ('LISTADO FINALIZADO')
```

El archivo GCS_WGS_1984.prj contiene lo siguiente:

```
GEOGCS["GCS_WGS_1984",DATUM["D_WGS_1984",SPHEROID["WGS_1984",
6378137,298.257223563]],PRIMEM["Greenwich",0],UNIT["Degree",0.0174532925
19943295]]
```

Para comprobar que se agregaron los rásteres, se abre la tabla de atributos del dataset

OBJECTID	Shape	Raster	Name	MinPS	MaxPS	LowPS	HighPS	Category	Tag	GroupName	ProductName	CenterX	CenterY	ZOrder	Raster Type ID	ItemTS	UriHash	Shape_Length	Shape_Area
1	Polygon	raster	2011-01-07T182440_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.405958	25.272178	<Null>	1	42955.88499	28F7B383460033100504F8304D83	0.99293	0.057381
2	Polygon	raster	2011-01-07T182514_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.39035	25.150269	<Null>	1	42955.88499	4830A0F08AA690F4C3944A025178	0.994503	0.057983
3	Polygon	raster	2011-01-07T182614_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.39035	25.150269	<Null>	1	42955.88499	4830A0F08AA690F4C3944A025178	0.994503	0.057983
4	Polygon	raster	2011-01-07T182617_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.39035	25.150269	<Null>	1	42955.88499	4830A0F08AA690F4C3944A025178	0.994503	0.057983
5	Polygon	raster	2011-01-07T182622_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.445019	25.709523	<Null>	1	42955.88499	905484541497A0E8B789031C74C028	0.991051	0.057585
6	Polygon	raster	2011-01-07T182625_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.445019	25.709523	<Null>	1	42955.88499	905484541497A0E8B789031C74C028	0.991051	0.057585
7	Polygon	raster	2011-01-07T182629_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.252105	25.819818	<Null>	1	42955.885	358AE214E61E8A138B038212E7521	0.992031	0.05777
8	Polygon	raster	2011-01-07T182633_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.252105	25.819818	<Null>	1	42955.885	358AE214E61E8A138B038212E7521	0.992031	0.05777
9	Polygon	raster	2011-02-06T183819_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.439328	25.138926	<Null>	1	42955.885	84059108728A102FA1F0287730E	0.992895	0.057790
10	Polygon	raster	2011-02-06T183821_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.439328	25.138926	<Null>	1	42955.885	84059108728A102FA1F0287730E	0.992895	0.057790
11	Polygon	raster	2011-02-06T183823_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.439328	25.138926	<Null>	1	42955.885	84059108728A102FA1F0287730E	0.992895	0.057790
12	Polygon	raster	2011-02-06T183825_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.439328	25.138926	<Null>	1	42955.885	84059108728A102FA1F0287730E	0.992895	0.057790
13	Polygon	raster	2011-02-06T183827_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.439328	25.138926	<Null>	1	42955.885	84059108728A102FA1F0287730E	0.992895	0.057790
14	Polygon	raster	2011-02-06T183829_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.439328	25.138926	<Null>	1	42955.885	84059108728A102FA1F0287730E	0.992895	0.057790
15	Polygon	raster	2011-02-06T183831_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.439328	25.138926	<Null>	1	42955.885	84059108728A102FA1F0287730E	0.992895	0.057790
16	Polygon	raster	2011-02-06T183833_REE_SAA_NAC_965990_137790_browse	0	0.004738	0.000474	0.000474	Primary	Dataset			-103.439328	25.138926	<Null>	1	42955.885	84059108728A102FA1F0287730E	0.992895	0.057790
17	Polygon	raster	2011-03-18T184120_REE_SAA_NAC_967990_137902_browse	0	0.004754	0.000475	0.000475	Primary	Dataset			-103.681879	25.92489	<Null>	1	42955.885	1088144F8B80D00080C890909C	0.992488	0.05777
18	Polygon	raster	2011-03-18T184124_REE_SAA_NAC_967990_137902_browse	0	0.004754	0.000475	0.000475	Primary	Dataset			-103.681879	25.92489	<Null>	1	42955.885	1088144F8B80D00080C890909C	0.992488	0.05777
19	Polygon	raster	2011-03-18T184127_REE_SAA_NAC_967990_137902_browse	0	0.004754	0.000475	0.000475	Primary	Dataset			-103.681879	25.92489	<Null>	1	42955.885	1088144F8B80D00080C890909C	0.992488	0.05777

Figura 3. 21 Tabla de atributos del dataset de mosaico

Esta tabla será la misma que se mostrará en la interfaz web. Esta tabla contiene un campo denominado "Name" que almacena el nombre de cada raster almacenado en el dataset de mosaico. Para este trabajo se agregaron a esta tabla los siguientes campos:

1. Pk_Id_Hechos_escena_de_satelite (entero largo)
2. Satellite (texto)
3. ModoEspectral (texto)
4. Resolucion (entero corto)
5. Fecha (date)
6. Hora (texto)
7. Nivel (texto)
8. Bandas (entero corto)
9. Porcentaje_de_nubes (entero corto)

PASO 2.1.- Una vez creados los campos, es necesario llenarlos con los datos correspondientes. En el geoportal existe una base de datos en SQL Server que contiene tablas que almacena los datos que describen a cada imagen y que son

los que se requieren para llenar los campos creados en el dataset de mosaico. Para obtener estos datos, se usó el siguiente comando de SQL:

```
SELECT amn.Archivo_vista_previa, hes.Pk_Id_Hechos_escena_de_satelite,
s.Nombre + ' ' + isnull(CONVERT(varchar, svs.Numero), '') as 'Satelite',
me.ModosEspectral, senmod.Resolucion_general_metros, hes.Fecha_z,
hes.Hora_z, dim.Nivel, senmod.bandas, hes.Porcentaje_de_nubes
FROM sde.SATELITEVERSION_SENSOR as svs,
sde.SATELITE as s,
sde.HECHOS_ESCENA_DE_SATELITE as hes,
sde.MODOESPECTRAL as me,
sde.SENSOR as sen,
sde.SENSOR_MODOESPECTRAL as senmod,
sde.ARCHIVOSESCENA_MODOESPECTRAL_NIVELDEPROCESAMIENTO as
amn,
sde.DIM_NIVEL_DE_PROCESAMIENTO as dim,
sde.SATELITE_NIVELDEPROCESAMIENTO as sn
WHERE svs .Pk_Id_Satelite = s .Pk_Id_Satelite
and svs .Pk_Id_Sensor = sen .Pk_Id_Sensor
and sen .Pk_Id_Sensor = senmod .Pk_Id_Sensor
and senmod .Pk_Id_ModosEspectral = me .Pk_Id_ModosEspectral
and svs .Pk_Id_SateliteVersion_Sensor = hes .Pk_Id_SateliteVersion_Sensor
and me .Pk_Id_ModosEspectral = amn .Pk_Id_ModosEspectral
and amn .Pk_Id_Hechos_escena_de_satelite = hes
.Pk_Id_Hechos_escena_de_satelite
and dim .Pk_Id_Nivel_de_procesamiento = sn .Pk_Id_Nivel_de_procesamiento
and s .Pk_Id_Satelite = sn .Pk_Id_Satelite
and amn .Pk_Id_Satelite_Nivel = sn .Pk_Id_Satelite_Nivel
```

PASO 2.2.- Después, se copió y pegó el resultado de la consulta a un archivo de Excel. En el resultado de la consulta, el campo “Archivo_vista_previa” contiene el nombre del archivo de vista previa de cada imagen (QuickLook) e incluye la extensión en el nombre (por ejemplo “.jpg”). Una vez que se copia el resultado de la consulta a un archivo de Excel, el campo “Archivo_vista_previa” se renombró a “Nombre”. Este campo (“Nombre”) se usa para hacer el join con la tabla de atributos del dataset de mosaico; el campo del dataset de mosaico que se usa para este join es el campo “Name”. Dado que los nombres de vistas previas no tienen extensión en el campo “Name”, se usó la función **SUSTITUIR** de Excel para quitar la extensión a los nombres almacenados en el campo “Nombre” y así poder realizar el join.

C	D	E	F	G	H	I	J
elite	Satellite	ModoEspectral	Resolucion	Fecha	Hora	Nivel	Bandas
1	SPOT 6	PMS	1.5	04/12/2013	17:12:40	ORTHO	4
Nombre							
PREVIEW_SPOT6_PMS_201312041712402_ORT_SPOT6_20140113_15245913rh49unxj1_1							

Figura 3. 22 Archivo en excel de los atributos a insertar

Posteriormente, en ArcMap se cargó el archivo de Excel con el resultado de la consulta y se hizo un join para unir los campos contenidos en este archivo a la tabla del dataset de mosaico. Para realizar el join se usaron, como campos comunes, el campo “Name” de la tabla de atributos del dataset de mosaico y el campo “Nombre” del archivo de Excel,

Una vez que se realiza el join, los registros que coinciden se guardan en una tabla.dbf, la cual se carga también en ArcMap. Enseguida, se copian los valores de los campos de esa tabla hacia los campos de la tabla de atributos del dataset de mosaico.

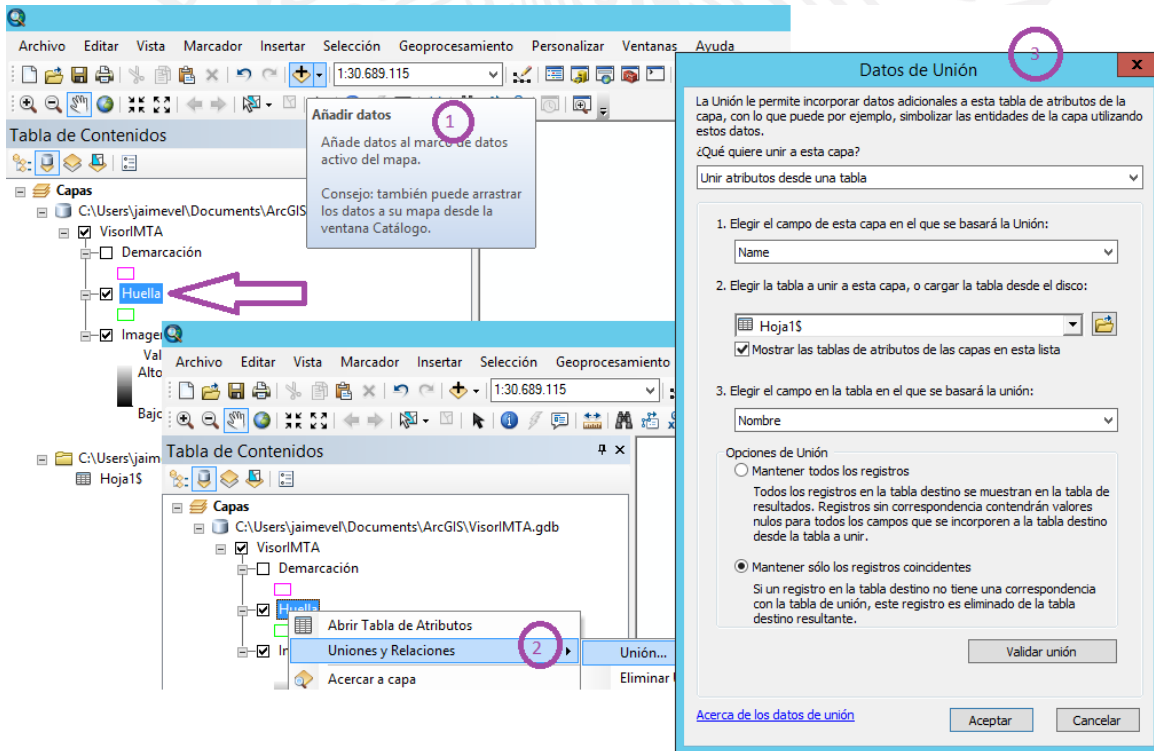


Figura 3. 23 Creación de la unión para obtener los registros coincidentes

PASO 2.3.-RECOMENDACIONES PARA FACILITAR EL PEGADO DE CAMPOS EN LA TABLA DE ATRIBUTOS

- ✓ Guardar en una tabla con extensión dbf el resultado de la unión (join). Luego copiar cada columna de la tabla dbf y pegarla directamente sobre la columna vacía de la tabla de atributos del dataset de mosaico. Los campos de la tabla ddbf se copian a su correspondiente campo en la tabla del dataset, por ejemplo: el campo “Satelite” de la tabla dbf se debe copiar al campo “Satelite” de la tabla del dataset de mosaico. Este copiado se debe hacer sin usar el modo de edición, ya que si se activa este modo, ArcMap se tardará más realizando esta operación o, cuando se pega un número grande de registros, no responderá.

PASO 2.4.- PUBLICACIÓN DEL SERVICIO DE IMÁGENES EN ArcGIS for Server

Para crear el servicio de imágenes usando el dataset de mosaico, en la ventana de diálogo de ArcCatálogo, se dio clic en el dataset de mosaico y se eligió la opción para publicarlo como un servicio. La figura siguiente muestra la ventana de diálogo de esta opción.

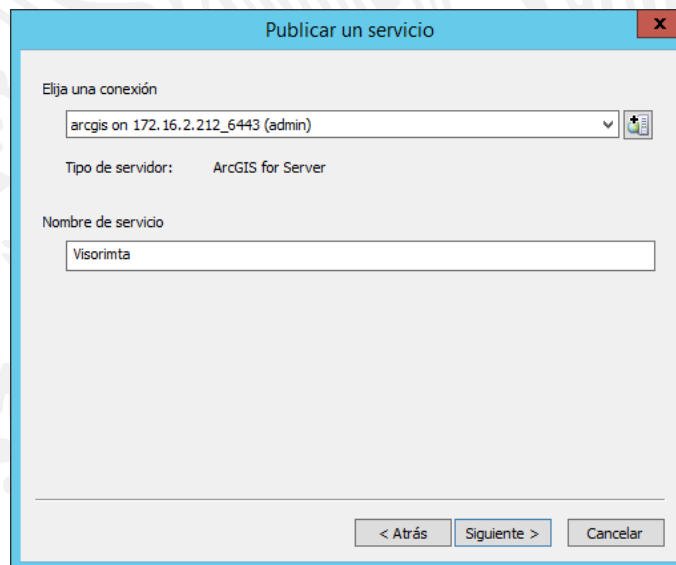


Figura 3. 24 Publicación de un servicio en ArcGIS Server

Antes de publicar el mosaico, se resuelven las advertencias que puede tener el dataset y que pueden impedir que el servicio se publique correctamente. Ya

resueltas, el servicio está listo para publicarse. La figura siguiente muestra la ventana de diálogo.

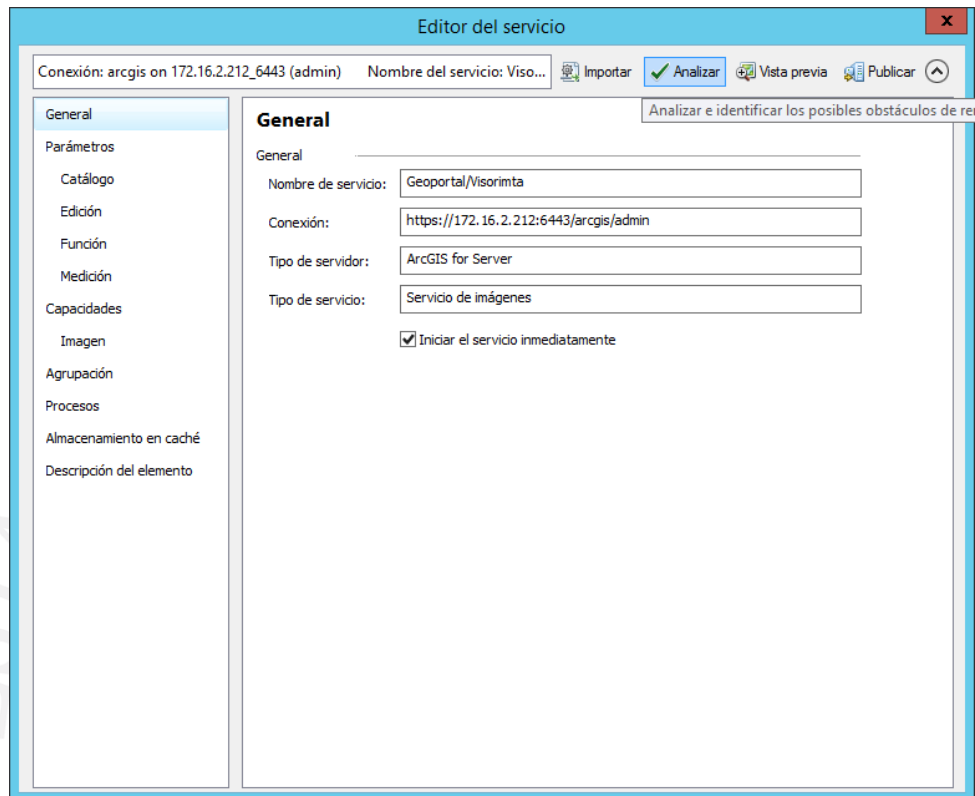


Figura 3. 25 Analizar el dataset de mosaico

PASO 3.-ESTRUCTURA DEL CÓDIGO FUENTE

En este punto se describe de forma general el código fuente que se desarrolló para implementar la nueva interfaz del geoportal, que usa el dataset de mosaico como la fuente de datos para visualizar las imágenes y descargarlas.

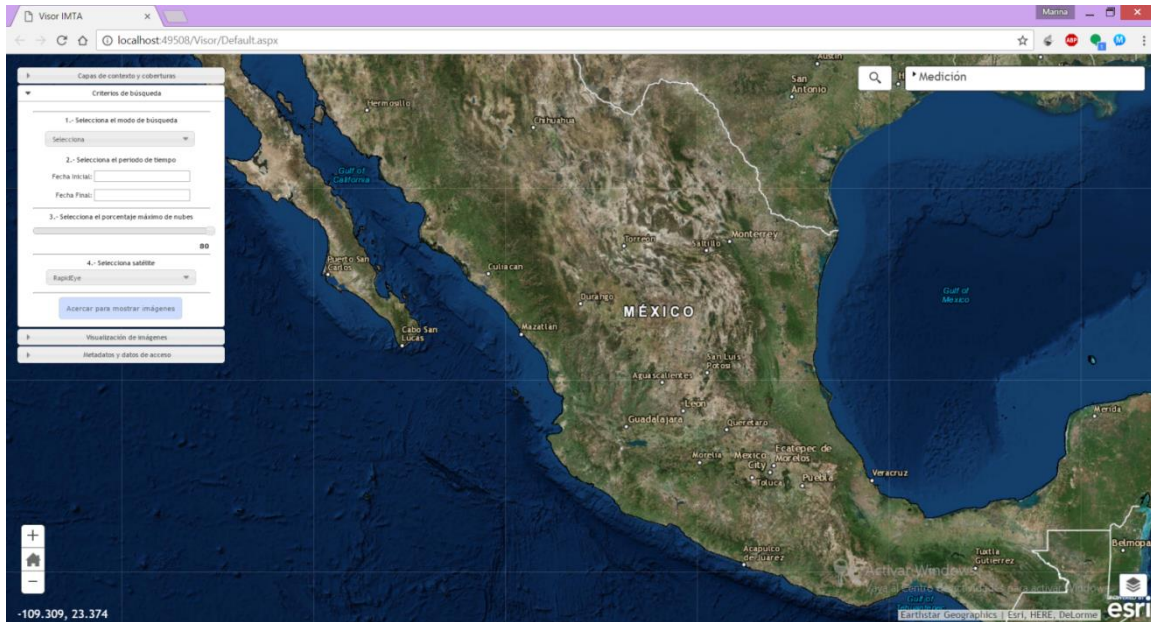


Figura 3. 26 Interfaz principal del visor de imágenes

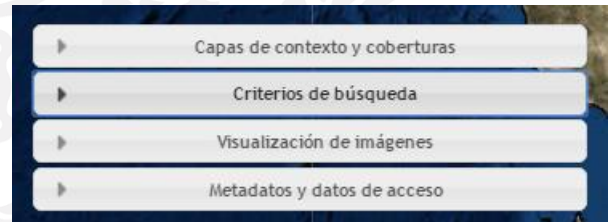
MENÚ

La API que se usó para darle funcionalidades al menú fue “Jquery UI”, el archivo que contiene estas se encuentra en js/JScript.js. Una parte de la funcionalidad principal (hecha con la API de JavaScript para ArcGIS Server) se encuentra en el archivo Default.aspx

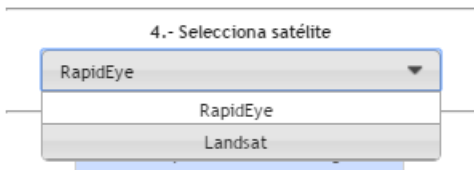
Widgets utilizados de la API de Jquery UI:

ACCORDION: Muestra paneles de contenido plegables para presentar información en una cantidad limitada de espacio.

```
$("#accordion").accordion({
    heightStyle: "content",
    collapsible: true,
    active: 1
});
```

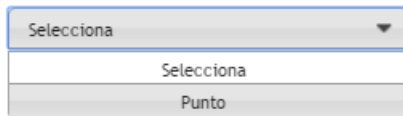


SELECTMENU: Representa un control que muestra un menú de opciones



```
$( "#satelite" ).selectmenu();
```

1.- Selecciona el modo de búsqueda



```
$( "#selectSearch" ).selectmenu({
  change: function( event, data ) {
    if(data.item.value === "Selecciona"){
      document.getElementById('selectvacio').style.display = 'block';
      document.getElementById('selectpunto').style.display = 'none';
      document.getElementById('selectresult').style.display = 'none';
    }
    if(data.item.value === "Punto"){
      document.getElementById('selectpunto').style.display = 'block';
    }
  }
});
```

DATEPICKER: Permite seleccionar una fecha de un calendario emergente o en línea.

```
$( ".datepicker" ).datepicker({
  changeMonth: true,
  changeYear: true,
});
```

Cambiar el idioma del calendario:

```
$.datepicker.regional['es'] = {
  closeText: 'Cerrar',
  prevText: '<Ant',
  nextText: 'Sig>',
  currentText: 'Hoy',
  monthNames: ['Enero', 'Febrero', 'Marzo', 'Abril', 'Mayo', 'Junio', 'Julio', 'Agosto',
'Septiembre', 'Octubre', 'Noviembre', 'Diciembre'],
  monthNamesShort: ['Ene', 'Feb', 'Mar', 'Abr', 'May', 'Jun', 'Jul', 'Ago', 'Sep', 'Oct',
'Nov', 'Dic'],
  dayNames: ['Domingo', 'Lunes', 'Martes', 'Miércoles', 'Jueves', 'Viernes',
'Sábado'],
```

```

dayNamesShort: ['Dom', 'Lun', 'Mar', 'Mié', 'Juv', 'Vie', 'Sáb'],
dayNamesMin: ['Do', 'Lu', 'Ma', 'Mi', 'Ju', 'Vi', 'Sá'],
weekHeader: 'Sm',
dateFormat: 'dd/mm/yy',
firstDay: 1,
isRTL: false,
showMonthAfterYear: true,
yearRange: "-50:+0",
yearSuffix: ""
};
$.datepicker.setDefaults($.datepicker.regional['es']);

```

2.- Selecciona el periodo de tiempo

Fecha Inicial:

Fecha Final:

3.- Selecciona

Lu	Ma	Mi	Ju	Vi	Sá	Do
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

4.

BUTTONSET: Permite agrupar un conjunto de botones relacionados.

Mejoras en la imagen

Mejoras en la imagen

Ninguno	Recorte porcentual	Desviaciones típicas
---------	--------------------	----------------------

La función realiza dos principales técnicas de realce de las imágenes..

```
$("#radio").buttonset();
```

En default.aspx:

```

$("#input:radio[name=realce]").click(function(){
    var value = $(this).attr("id");

    if(value == "clip"){
        var clip = new RasterFunction();
        clip.functionName = "Stretch";
        clip.functionArguments = {

```

```

"StretchType" : 6,
"MinPercent" : 2,
"MaxPercent" : 2,
"Gamma" : [2.25, 2.25, 2.25],
"DRA": true,
"Min": 100,
"Max": 255,
"Raster" : "$$"
};

clip.outputPixelFormat = "U8";
clip.variableName = "Raster";
imageServiceLayer.setRenderingRule(clip);
}
else if (value == "desv"){
var desv = new RasterFunction();
desv.functionName = "Stretch";
desv.functionArguments = {
"StretchType" : 3,
"NumberOfStandardDeviations" : 2.5,
"UseGamma": true,
"Gamma" : [1.25, 1.25, 1.25],
"Raster" : "$$"
};
desv.outputPixelFormat = "U8";
desv.variableName = "Raster";
imageServiceLayer.setRenderingRule(desv);
}
else {
imageServiceLayer.setRenderingRule();
}
});

```

SLIDER: Control deslizante básico horizontal con un pequeño botón que se puede mover con el ratón o usando las teclas de flecha.

Porcentaje máximo de nubes

3.- Selecciona el porcentaje máximo de nubes

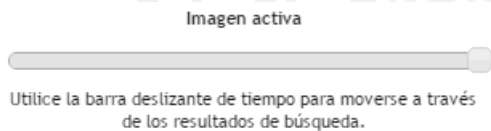
```
$(function () {
```

```

$("#slider-range-min").slider({
  range: "min",
  min: 0,
  max: 80,
  value: 80,
  step: 10,
  slide: function (event, ui) {
    $("#amount").val(ui.value);
  }
});
$("#amount").val($("#slider-range-min").slider("value"));
});

```

Imagen activa

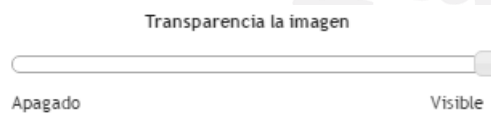


```

$(function () {
  $("#slider-range-max").slider({
    range: "min",
    min: 0,
    max: 80,
    value: 80,
    step: 10,
    slide: function (event, ui) {
      $("#amount1").val(ui.value);
    }
  });
  $("#amount1").val($("#slider-range-max").slider("value"));
});

```

Transparencia de la imagen



En default.aspx:

```

$(function () {

```



```

$("#slider").slider({
  min: 0,
  max: 100,
  value: 100,
  slide: function (event, ui) {
    $("#amount2").val(ui.value);
  }
});
$("#amount2").val($("#slider").slider("value"));
$("#slider").on("slidestop", function( event, ui ) {
  var total = 0;
  var selection = 0;
  selection = ui.value;
  total = selection/100;
  imageServiceLayer.setOpacity(total);
});
});

```

PASO 3.1.- WIDGETS UTILIZADOS DE LA API DE ARCGIS (UBICADOS EN DEFAULT.ASPX)

BÚSQUEDA BÁSICA

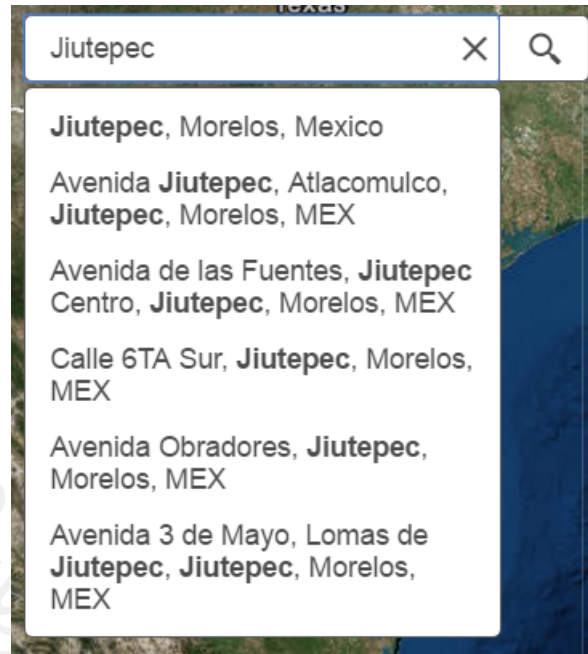
Widget para simplificar el proceso de búsqueda de ubicaciones en un mapa.



```

var search = new Search({
  enableButtonMode: true,
  map: map,
  enableInfoWindow: false
}, "search");

```



El código mostrado abajo se usó para limitar el área geográfica para una búsqueda, de modo que sólo se busquen localidades de México.

```
var extent = new Extent({
  "spatialReference": {
    "wkid": 4326
  },
  "xmin": -117.230,
  "xmax": -86.776,
  "ymin": 14.470,
  "ymax": 32.763
});

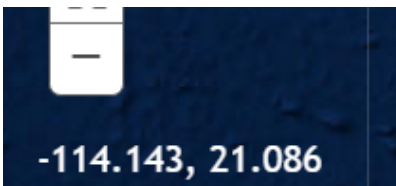
search.sources[0].searchExtent = extent;

search.startup();
```

div del widget:

```
<div id="search"></div>
```

MOSTRAR COORDENADAS X, Y



```
map.on("load", function() {
  map.on("mouse-move", showCoordinates);
  map.on("mouse-drag", showCoordinates);
});

function showCoordinates(evt) {
  var mp = webMercatorUtils.webMercatorToGeographic(evt.mapPoint);
  dom.byId("info").innerHTML = mp.x.toFixed(3) + ", " + mp.y.toFixed(3);
}
```

span del widget:

```
<span id="info" style="position:absolute; left:15px; bottom:5px; color:#fff; z-index:50;"></span>
```

BOTÓN DE INICIO Y BOTONES DE ZOOM



```
var home = new HomeButton({
  theme: "homeButton",
  map: map
}, "HomeButton");
home.startup();
```

Dentro de la variable map, se agregan las propiedades para activar y posicionar el slider del zoom.

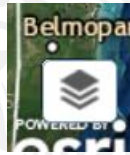
```
map = new Map("map", {
  basemap: "hybrid",
  center: [-104.6508, 23.5452],
  zoom: 6,
  slider: true,
  sliderPosition: "bottom-left",
});
```

El siguiente código ubica el botón de inicio entre los botones del zoom.

```
dojo.connect(map, "onLoad", function(){
    var initExtent = map.extent;

    dojo.create("div", {
        className: "esriSimpleSliderHomeButton",
        onclick: function(){
            map.setExtent(initExtent);
        }
    }, dojo.query(".esriSimpleSliderIncrementButton")[0], "after");
});
```

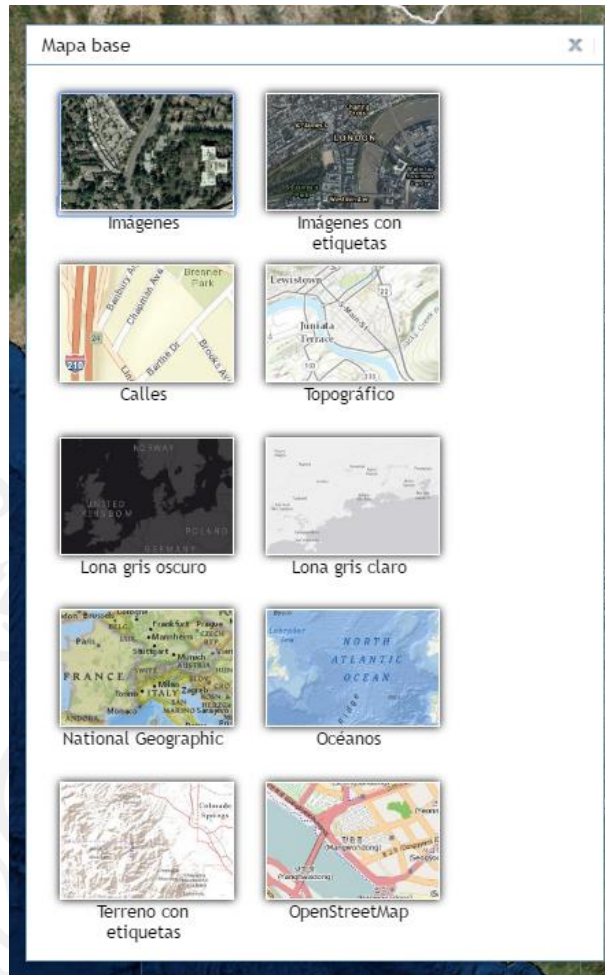
GALERÍA DE MAPAS DE FONDO.



```
var basemapGallery = new BasemapGallery({
    showArcGISBasemaps: true,
    map: map
}, "basemapGallery");
basemapGallery.startup();

basemapGallery.on("error", function (msg) {
    console.log("basemap gallery error: ", msg);
});
```

div del widget:



```
<div style="position:absolute; right:1%; bottom:5%; z-Index:999; font-size:smaller;">
```

```
  <button data-dojo-type="dijit/form/Button" onClick="mySizedDialog.show();" data-dojo-props="iconClass:'plusIcon', showLabel: false" type="button" title="Mapa base"></button>
```

```
  <div data-dojo-id="mySizedDialog" data-dojo-type="dijit/Dialog" title="Mapa base" style="width:auto;height:auto;">
```

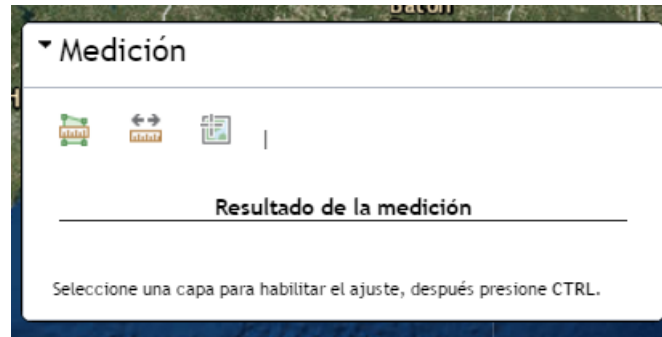
```
    <div id="basemapGallery" ></div>
```

```
  </div>
```

```
</div>
```

MEDICIÓN

Proporciona herramientas para determinar áreas y distancias, y dar la ubicación x, y, dada en longitud, latitud.



```

var sfs = new SimpleFillSymbol(
    "solid",
    new SimpleLineSymbol("solid", new Color([195, 176, 23]), 2),
    null
);

var parcelsLayer = new
FeatureLayer("http://sisgrh.imta.mx/ArcGIS/rest/services/a_shapes/MapServer", {
    mode: FeatureLayer.MODE_ONDEMAND,
    outFields: ["*"]
});
parcelsLayer.setRenderer(new SimpleRenderer(sfs));
map.addLayers([parcelsLayer]);
var snapManager = map.enableSnapping({
    snapKey: has("mac") ? keys.META : keys.CTRL
});
var layerInfos = [{
    layer: parcelsLayer
}];
snapManager.setLayerInfos(layerInfos);

var measurement = new Measurement({
    map: map
}, dom.byId("measurementDiv"));
measurement.startup();

var imageParameters = new ImageParameters();
imageParameters.layerIds = [19];
imageParameters.layerOption = ImageParameters.LAYER_OPTION_SHOW;
layer = new
ArcGISDynamicMapServiceLayer("http://sisgrh.imta.mx/ArcGIS/rest/services/Geop
ortal84/MapServer",
    { "imageParameters": imageParameters });

```

```
var imageParameters = new ImageParameters();
imageParameters.layerIds = [19];
imageParameters.layerOption = ImageParameters.LAYER_OPTION_SHOW;
layer = new
ArcGISDynamicMapServiceLayer("http://sisgrh.imta.mx/ArcGIS/rest/services/Geop
ortal84/MapServer",
{ "imageParameters": imageParameters });
map.addLayer(layer);
```

div del widget:

```
<div style="position:fixed; right:1%; top:2%; z-Index:999; font-size:smaller;">
  <div id="titlePane" data-dojo-type="dijit/TitlePane" data-dojo-
  props="title:'Medición', closable:false, open:false">
    <div id="measurementDiv"></div>
    <span style="font-size:smaller;padding:5px 5px;">Seleccione una capa
    para habilitar el ajuste, después presione <b>CTRL</b>.</span>
  </div>
</div>
```

ELEGIR PUNTO PARA BÚSQUEDA DE IMÁGENES

Muestra coordenadas x,y en un ventana de información cuando el usuario hace clic en el mapa.

1.- Selecciona el modo de búsqueda

Punto

Especificar coordenadas con el cursor
Al terminar, desmarque la casilla

Checkbox para habilitar el clic.

```
checkBox = new CheckBox({
  name: "checkBox",
  value: "agreed",
  checked: false,
  onChange: function(b){
    var mapclickhandle;
```

```

    if(b===true){
        mapclickhandle = on(map, "click", addPoint);
    }
    else{
        dojo.disconnect(mapclickhandle);
        document.getElementById('selectpunto').style.display = 'none';
        document.getElementById("selectresult").innerHTML = "lon/lat : " +
coordx + ", " + coordy;
        document.getElementById('selectresult').style.display = 'block';
    }
}
}, "checkBox");

```

Función para agregar el punto

```

function addPoint(evt) {
    var latitude = evt.mapPoint.getLatitude();
    var longitud = evt.mapPoint.getLongitude();
    map.infoWindow.resize(180,100);
    map.infoWindow.setTitle("Coordenadas");
    map.infoWindow.setContent(
        "lon/lat : " + longitud.toFixed(2) + ", " + latitude.toFixed(2)
    );
    map.infoWindow.show(evt.mapPoint,
map.getInfoWindowAnchor(evt.screenPoint));
    coordx = "";
    coordy = "";
    coordx = longitud.toFixed(2);
    coordy = latitude.toFixed(2);
}

```

Divs para esta función:

```

<div id="selectpunto" style="display:none">
<input id="checkBox" /> Especificar coordenadas con el cursor<br>Al
terminar, desmarque la casilla
</div>
<div id="selectresult" style="display:none"> </div>

```

Función que habilita los divs anteriores

```

$("#selectSearch" ).selectmenu({
    change: function( event, data ) {
        if(data.item.value === "Selecciona"){

```



```

document.getElementById('selectvacio').style.display = 'block';
document.getElementById('selectpunto').style.display = 'none';
document.getElementById('selectresult').style.display = 'none';
}
if(data.item.value === "Punto"){
document.getElementById('selectpunto').style.display = 'block';
}
}
});

```

TABLA DE RESULTADOS

Muestra los atributos de una capa de entidad y establece interacciones entre el mapa y la tabla.

Esta funcionalidad crea un feature layer para tomar los datos que se mostrarán al usuario (mismos que se agregaron a la tabla de atributos de la huella del servicio de imágenes) mediante la unión en ArcMap.

```

var myFeatureLayer = new
FeatureLayer("http://172.16.2.212/servicios/rest/services/Geoportal/VisorIMTA/ImageServer", {
mode: FeatureLayer.MODE_ONDEMAND,
visible: true,
outFields: [
"OBJECTID", "Satelite", "ModoEspectral", "Resolucion", "Fecha", "Hora",
"Nivel", "Bandas", "Porcentaje_de_nubes"
],
id: "fLayer"
});
map.addLayer(myFeatureLayer);

```

Por último, se agrega el featureTable

```

myFeatureTable = new FeatureTable({
featureLayer : myFeatureLayer,
outFields: ["OBJECTID", "Satelite", "ModoEspectral", "Resolucion",
"Fecha", "Hora", "Nivel", "Bandas", "Porcentaje_de_nubes"],
fieldInfos: [
{
name: 'Satelite',
alias: 'Satélite'
},
{

```

```

    name: 'ModoEspectral',
    alias: 'Modo espectral'
  },
  {
    name: 'Resolucion',
    alias: 'Resolución'
  }
  {
    name: 'Porcentaje_de_nubes',
    alias: 'Porcentaje de nubes'
  }
],
map : map,
showGridMenu : false,
syncSelection : true,
zoomToSelection : true
}, "myTableNode");

myFeatureTable.startup();

```

div del widget

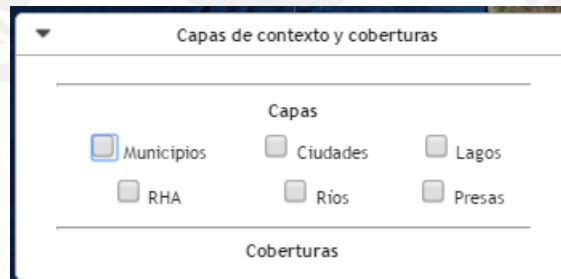
```

<div data-dojo-type="dijit/Dialog" data-dojo-id="dialogResults" data-dojo-
props="title: 'Todos los resultados de la consulta' style="width: 100%; height: 30%;
align-content: bottom;">
  <div data-dojo-type="dijit/layout/ContentPane" id="div1" data-dojo-
props="style: {overflow: 'none', padding: 10, height: '100%', width: '100%'}">
    <div id="myTableNode"></div>
  </div>
</div>

```

CAPAS DE CONTEXTO

Muestra una lista de capas en un servicio de mapas. Este servicio se encuentra alojado en el servidor SISGRH.



Checkbox de los elementos

```
on(dom.byId("layer1CheckBox"), "change", updateLayerVisibility);
on(dom.byId("layer2CheckBox"), "change", updateLayerVisibility);
on(dom.byId("layer3CheckBox"), "change", updateLayerVisibility);
on(dom.byId("layer4CheckBox"), "change", updateLayerVisibility);
on(dom.byId("layer5CheckBox"), "change", updateLayerVisibility);
on(dom.byId("layer6CheckBox"), "change", updateLayerVisibility);
```

La siguiente función contiene la lógica que activa y desactiva las capas.

```
function updateLayerVisibility() {
    var inputs = query(".list_item");
    var inputCount = inputs.length;
    visibleLayerIds = [19];
    for (var i = 0; i < inputCount; i++) {
        if (inputs[i].checked) {
            visibleLayerIds.push(inputs[i].value);
        }
    }
    if (visibleLayerIds.length === 0) {
        visibleLayerIds.push(-1);
    }
    layer.setVisibleLayers(visibleLayerIds);
}
```

div de la función.

```
<div id="capascontexto">
<span id="layer_list">
<table class="pure-table" style="width:100%">
<tr>
<td colspan="3"><center><b>Capas</b></center></td>
</tr>
<tr>
<td>
<input type='checkbox' class='list_item' id='layer3CheckBox'
value=5 />Municipios
</td>
<td>
<input type='checkbox' class='list_item' id='layer1CheckBox' value=3
/>Ciudades
```

```

</td>
<td>
  <input type='checkbox' class='list_item' id='layer2CheckBox'
value=4 />Lagos
</td>
</tr>
<tr>
  <td>
    <input type='checkbox' class='list_item' id='layer4CheckBox'
value=6 />RHA
  </td>
  <td>
    <input type='checkbox' class='list_item' id='layer5CheckBox'
value=1 />Ríos
  </td>
  <td>
    <input type='checkbox' class='list_item' id='layer6CheckBox'
value=0 />Presas
  </td>
</tr>
</table>
</span>
</div>

```

3.6 Determinar las cuencas hidrológicas asociadas a 50 manglares identificados para formar parte del Plan Nacional de Restauración y Reforestación de Manglar (PNRRM) y sus cuencas hidrológicamente conectadas, y realizar diez estudios para la determinación de reservas potenciales de agua para el medio ambiente en cuencas costeras asociadas a manglares.

El Plan Nacional de Restauración y Reforestación de Manglar (PNRRM) establece 33 sitios prioritarios para su restauración; en este estudio para cada sitio de manglar se define la red hidrológica superficial que aporta agua a la zona donde está presente el manglar con la finalidad de calcular el caudal ecológico que es necesario para preservar la existencia de los ecosistemas y de las especies animales y vegetales, siendo el manglar parte de esta última. Los manglares habitan en humedales costeros, que es la zona de transición entre el mar y la parte continental, en él conviven ecosistemas de diferentes condiciones biológicas, tanto animales como vegetales, y son ricos en recursos como agua y peces.

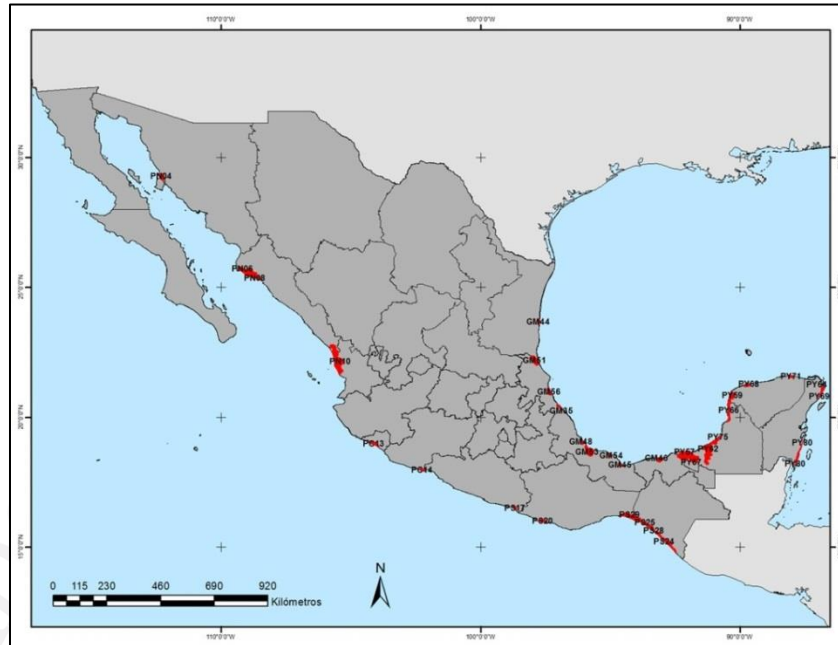


Figura 3. 27 Ubicación geográfica de los 33 sitios seleccionados (Fuente: CONABIO)

En este estudio el cálculo del caudal ecológico se llevó a cabo de acuerdo con lo establecido en la NMX-AA-159-SCFI-2012; en este caso los valores de caudal ecológico se asocian con el requerimiento de agua que el manglar necesita para su conservación; el volumen de agua disponible por cuenca se toma de lo publicado en DOF en 2016 y que la CONAGUA ha difundido por medio de una base de datos con la información publicada. La disponibilidad en la cuenca se considera como el agua existente para satisfacer o no el requerimiento del manglar.

Para el cálculo de la Reserva Potencial de Agua (RPA) para cada cuenca se aplicó la metodología propuesta por el documento “Identificación de reservas potenciales de agua para el medio ambiente en México” publicado por la SEMARNAT en 2011; los datos utilizados para este cálculo se actualizaron de acuerdo con lo publicado por cada dependencia que los emite, como la disponibilidad de agua por cuenca, los sitios RAMSAR, la presencia de grandes presas, etc. Cabe mencionar que para llevar a cabo esta actividad se realizó una orden de servicio denominada “Realizar diez estudios para la determinación de reservas potenciales de agua para el medio ambiente en cuencas costeras asociadas a manglares.”, cuyos informes parcial y final están incluidos en el DVD que acompaña a este informe.

Los resultados de esta actividad se presentan en la tabla 1, mostrada a continuación, donde se especifica el nombre de la cuenca o cuencas que tienen

presencia espacial en el manglar. En todos los casos son cuencas costeras, por lo tanto en ellas confluye la red hidrológica superficial, ya sea por cuenca propia o la que nace desde aguas arriba. La primeras cuencas con corrientes propias de escurrimiento (las que no reciben escurrimientos de cuencas aguas arriba), se realzan en la tabla en color café claro, en este caso son 14 sitios de manglar que cumplen esta condición. La otra condición considera a las cuencas que reciben escurrimientos de cuencas que están aguas arriba, en estos caso la corriente principal es la que aporta la mayor cantidad del recurso hídrico a la zona de manglar; sin embargo en todos los sitios se presentan cuencas adyacentes que presentan escurrimientos de cuenca propia, ya sea para confluir con la corriente principal o desembocar directamente al mar o alguna laguna costera, los sitios que cumplen esta condición se realzan en color amarillo.

REGIÓN	CLAVE CONABIO	NOMBRE DEL SITIO	CUENCA DONDE SE PRESENTA EL MANGLAR	DISPONIBILIDAD PARA CAUDAL ECOLÓGICO	OBJETIVO AMBIENTAL	FACTIBILIDAD RPA
Pacífico Norte	PN04	Estero El Sargento-Isla Tiburón	Arivaipa-Puerto Libertad	Sin disponibilidad		Se excluye
	PN06	Isla Santa María-Topolobampo-Ohuira	Grupo de corrientes Topolobampo	Disponibilidad	Muy bueno	Media
	PN08	Ignacio-Navachiste-Macapule	Río Sinaloa 2, Grupo de corrientes de Topolobampo	Disponibilidad	Moderado, Muy bueno	Se excluye, Media
	PN10	Teacapán-Agua Brava-Marismas Nacionales	No evaluada			
Pacífico Centro	PC13	Laguna de Cuyutlán	Armería, Río Marabasco B	Disponibilidad	Bueno, Muy bueno	Se excluye, Se excluye
	PC14	Laguna El Caimán	Río Bajo Balsas, Río Acapulcan	Disponibilidad	Muy bueno, Muy bueno	Se excluye, No candidata
Pacífico Sur	PS17	Barra de Tecoanapa	Río Ometepec 4, Río La Arena	Disponibilidad	Muy bueno, Muy	Media, Media

REGIÓN	CLAVE CONABIO	NOMBRE DEL SITIO	CUENCA DONDE SE PRESENTA EL MANGLAR	DISPONIBILIDAD PARA CAUDAL ECOLÓGICO	OBJETIVO AMBIENTAL	FACTIBILIDAD RPA
			3		bueno	
	PS20	Chacagua-Pastoría	Río Verde, Río San Francisco, Río Grande	Disponibilidad	Muy bueno, Muy bueno, Muy bueno	Media, Media, No candidata
	PS24	La Encrucijada	San Diego, Pipijiapan, Margaritas y Coapa, Novillero Alto, Sesecapa, Cacaluta, Laguna del Viejo y Temblader, Despoblado, Huixtla, Huehuetán, Coatán	Disponibilidad	Muy bueno, Muy bueno, Muy bueno, Muy bueno, Muy bueno, Muy bueno, Muy bueno, Muy bueno, Muy bueno, Muy bueno, Muy bueno	Media, Media, Media, Media, Media, Media, Alta, Media, Se excluye
	PS25	La Joya	Laguna de la Joya	Disponibilidad	Muy bueno	Media
	PS28	Los Patos Solo Dios	Laguna de la Joya, Jesús, San Diego, El Porvenir	Disponibilidad	Muy bueno, Muy bueno, Muy bueno, Muy bueno	Media, Media, No candidata

REGIÓN	CLAVE CONABIO	NOMBRE DEL SITIO	CUENCA DONDE SE PRESENTA EL MANGLAR	DISPONIBILIDAD PARA CAUDAL ECOLÓGICO	OBJETIVO AMBIENTAL	FACTIBILIDAD RPA
	PS29	Mar Muerto	Río Ostuta 2, Laguna del Mar Muerto A, Tapanatepec, Laguna Mar Muerto B, Las Arenas, La Punta, Laguna Mar Muerto C, Sanatenco	Disponibilidad	Muy bueno, Muy bueno, Muy bueno, Muy bueno, Muy bueno, Muy bueno, Muy bueno, Muy bueno	No candidata, No candidata, No candidata, No candidata, Media, Media, Media
Península de Yucatán	PY57	Atasta Norte	Laguna de Términos	Disponibilidad	Muy bueno	Alta
	PY59	Celestún	Yucatán	Disponibilidad	Muy bueno	Alta
	PY62	Isla Aguada-Boca de Pargos	Bajo río Candelaria, Laguna de Términos, Mamatel, Cumpan	Disponibilidad	Muy bueno, Muy bueno, Muy bueno, Muy bueno	Media, Alta, Media, Alta
	PY64	Nichupté	Yucatán	Disponibilidad	Muy bueno	Alta
	PY66	Peténes	Yucatán, Río Verde, Campeche	Disponibilidad	Muy bueno, Muy bueno,	Alta, Media, Se excluye
	PY67	Pom-Atasta	San Pedro y San Pablo, Laguna del Este, Cumpan,	Disponibilidad	Muy bueno, Muy bueno, Muy	Media, Alta, Alta, Alta

REGIÓN	CLAVE CONABIO	NOMBRE DEL SITIO	CUENCA DONDE SE PRESENTA EL MANGLAR	DISPONIBILIDAD PARA CAUDAL ECOLÓGICO	OBJETIVO AMBIENTAL	FACTIBILIDAD RPA
			Laguna del Pom y Atasta		bueno, Muy bueno	
	PY68	Progreso	Yucatán	Disponibilidad	Muy bueno	Alta
	PY69	Puerto Morelos-Punta Maroma	Yucatán	Disponibilidad	Muy bueno	Alta
	PY71	Río Lagartos	Yucatán	Disponibilidad	Muy bueno	Alta
	PY75	Sabancuy-Chen kan	Laguna de Términos, Malinche, Vicente Guerrero	Disponibilidad	Muy bueno, Muy bueno,	Alta, Media, Se excluye
	PY80	Costa Maya	Chinchancanab	Disponibilidad	Muy bueno	Alta
Golfo de México	GM35	Ciénega del Fuerte	Llanuras de Tuxpan, Río Nautla	Disponibilidad	Muy bueno, Muy bueno	Alta, Se excluye
	GM44	Laguna de Morales	Río Soto La Marina 3, Laguna Morales	Disponibilidad	Bueno, Muy bueno	No candidata, No candidata
	GM45	Laguna Ostión	Río Huazuntlán, Llanuras de Coatzacoalcos	Disponibilidad	Muy bueno, Muy bueno	Media, No candidata
	GM46	Lagunas Mecoacan-Julivá-Santa Anita	El Carrizal, Caxcuchapa	Disponibilidad	Muy bueno, Muy bueno	Alta, No candidata
	GM48	Mandinga	Jamapa-Cotaxtla	Disponibilidad	Muy bueno	Alta

REGIÓN	CLAVE CONABIO	NOMBRE DEL SITIO	CUENCA DONDE SE PRESENTA EL MANGLAR	DISPONIBILIDAD PARA CAUDAL ECOLÓGICO	OBJETIVO AMBIENTAL	FACTIBILIDAD RPA
	GM5 1	Pueblo Viejo-El Chairel	Río Pánuco 2, Arroyo Tamacuil o La Llave, Río Tamesí	Disponibilidad	Muy bueno, Deficiente, Deficiente	Media, Media, Se excluye
	GM5 3	Sistema Lagunar de Alvarado	Llanuras del Papaloapan, Río Blanco	Disponibilidad	Muy bueno, Deficiente	Alta, Se excluye
	GM5 4	Sontecomapan	Llanuras del Papaloapan	Disponibilidad	Muy bueno	Alta
	GM5 5	Tecolutla	Río Tecolutla, Llanuras de Tuxpan	Disponibilidad	Muy bueno, Muy bueno	Se excluye, Alta
	GM5 6	Tuxpan	Llanuras de Tuxpan, Estero Galindo, Río Tuxpan	Disponibilidad	Muy bueno, Muy bueno, Muy bueno	Alta, No candidata, Se excluye

Tabla 1 cuencas que tienen presencia espacial en el manglar

3.7 Elaborar un artículo de divulgación.

En esta actividad se elaboró el artículo de divulgación titulado “Sistema de información de imágenes de satélite de los recursos naturales para soporte al estudio de los recursos hídricos”, el cual se envió a la Sociedad Latinoamericana de Percepción Remota y Sistemas de Información Espacial (SELPER-México) para su publicación en el boletín electrónico de esta sociedad. Adicionalmente, se elaboró una nota técnica sobre el sistema, la cual se publicó en el portal del IMTA en el sitio Web del Gobierno Federal.

4. RESULTADOS

El cuadro mostrado abajo presenta un resumen de las imágenes que se cargaron al geoportal.

Satélite	Número de imágenes cargadas
SPOT	439
LANDSAT	173
RAPID EYE	25,702

Al finalizar la etapa de la carga de imágenes, se verificó que los atributos de las escenas se encontraran en las tablas correspondientes de la geobase, los archivos de las imágenes se encontraran en la NAS y los cuadrantes estuvieran en las capas geográficas correspondientes en la geobase, indicando con un mensaje cuáles fueron insertadas correctamente y cuáles no.

Por otro lado, el cuadro mostrado abajo muestra el número de fotos aéreas u ortofotos que se cargaron al geoportal. Esta información no se consideró en la apertura de proyecto, sin embargo se dio de alta porque es útil para los proyectos que se desarrollan en el Instituto.

Resolución de la ortofoto	Número de ortofotos cargadas
2 metros	5,250
1.5 metros	319

Las mejoras realizadas a la interfaz de consulta incluyen el elegir más de dos satélites, buscar por periodo de tiempo, y definir un área mediante un polígono, todo lo cual permite obtener más resultados en la búsqueda y reducir el número de consultas. Esto facilita grandemente el uso del sistema a los usuarios.

Por otra parte, las 2 figuras mostradas abajo presentan la interfaz de la nueva aplicación que se desarrolló y un ejemplo de consulta con ella. La interfaz se

desarrolló usando el dataset de mosaicos que se creó y la API de ArcGIS para JavaScript. El servicio de imágenes se creó con el software ArcGIS for Server 10.4.1. Cabe mencionar que el mosaico de imágenes contiene sólo imágenes RapidEye (coberturas nacionales del periodo de años 2011 - 2016) e imágenes LANDSAT 7, ya que éstas son las únicas que tienen sus vistas previas georreferenciadas. Para visualizar el resto de las imágenes contenidas en el geoportal en la nueva interfaz, se requiere georeferenciar sus vistas previas. Si se continúa este proyecto el próximo año, se deberán considerar recursos para realizar esta actividad.

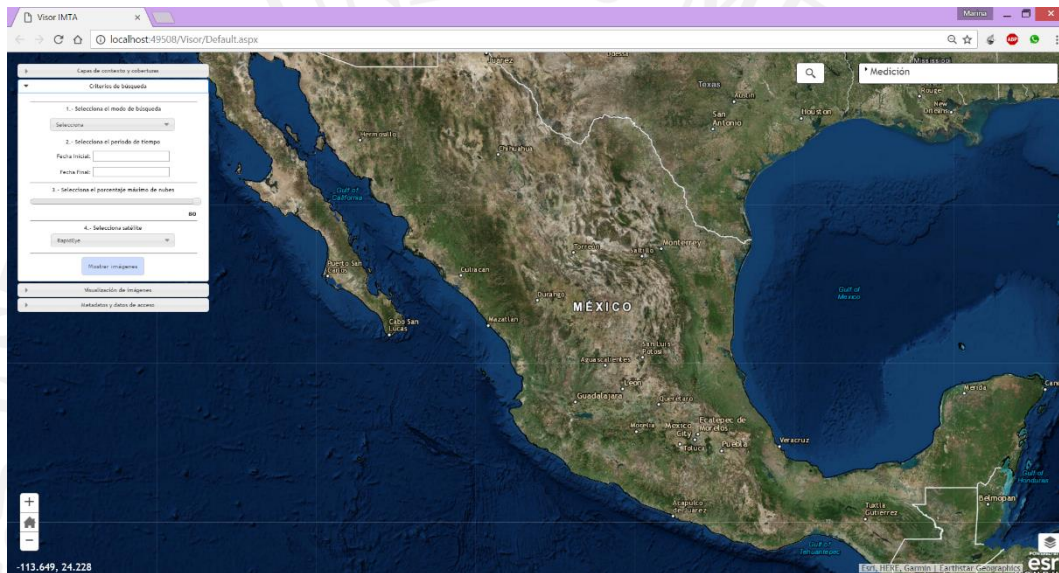


Figura 4. 1 Ventana de diálogo principal de la nueva interfaz del geoportal

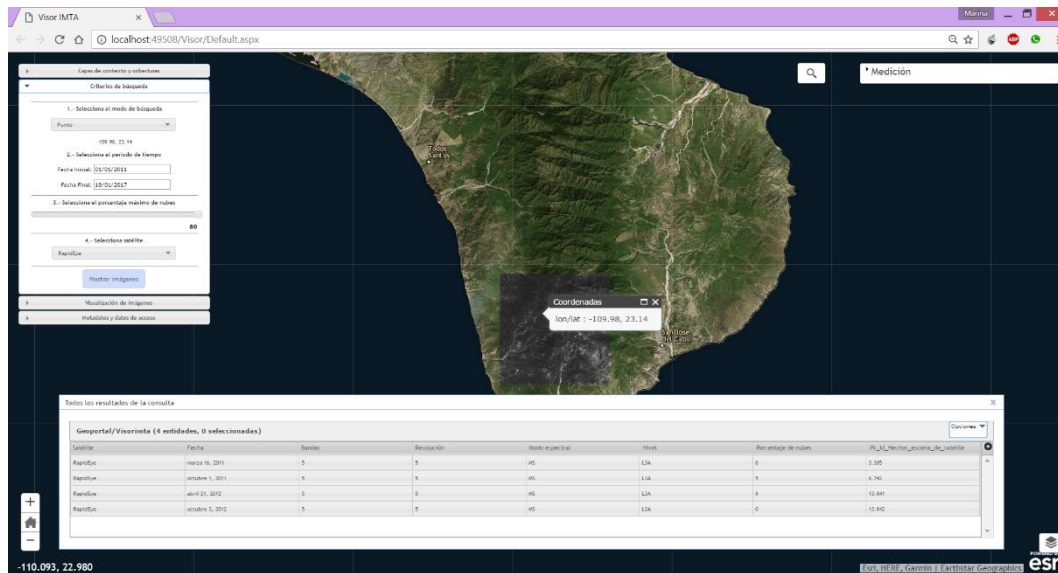


Figura 4. 2 Resultados obtenidos mediante búsqueda por punto

El cálculo de áreas y distancias se encuentra en esta una nueva aplicación, que a su vez permite ver las imágenes sobre el mapa y aplicar diferentes realces.

Finalmente, en relación a la actividad denominada “Determinar las cuencas hidrológicas asociadas a 50 manglares identificados para formar parte del Plan Nacional de Restauración y Reforestación de Manglar (PNRRM) y sus cuencas hidrológicamente conectadas, y realizar diez estudios para la determinación de reservas potenciales de agua para el medio ambiente en cuencas costeras asociadas a manglares”, se debe aclarar que sólo se hizo el estudio para 33, ya que éstos fueron los que la CONABIO consideró (posterior al inicio de este proyecto), como prioritarios.

5. CONCLUSIONES Y RECOMENDACIONES

En este proyecto se adquirió, instaló y configuró una versión más reciente de ArcGIS Server junto con la extensión ArcGIS Image Extension for Server. Adicionalmente, se cargaron las imágenes provenientes de los satélites SPOT y Landsat, y las fotos aéreas que faltaban, y se pusieron disponibles al personal del Instituto a través de la intranet.

El desarrollo de una nueva versión del geoportal, en donde se usa un dataset de mosaico para almacenar las imágenes, permitió explorar las funciones con las que cuenta el software ArrGIS for Server y la API que este ofrece para desarrollo de aplicaciones. La nueva versión permite una consulta más rápida de los datos y ofrece más funciones al usuario. Se recomienda continuar con el desarrollo de esta versión para incorporar más imágenes de satélite, modelos de elevación y fotos aéreas. Una labor importante que hay que realizar para hacer esto es georeferenciar las vistas previas de estos datos.

Por otro lado, se mejoró el protocolo de carga de imágenes al geoportal, añadiendo reglas de validación que permiten excluir las escenas que ya han sido insertadas anteriormente y asegurar la consistencia en las tres partes que implican la carga de imágenes (creación del cuadrante, copia de archivos en el servidor e inserción de los atributos en la base de datos).

La aplicación informática del Geoportal permite ahora (al momento de realizar una consulta) seleccionar múltiples satélites, definir un periodo de tiempo y crear un polígono que indica el área de interés del usuario. Finalmente, se recomienda continuar con el proyecto con el fin de incorporar también datos vectoriales, los cuales son de utilidad también en los proyectos que se realizan en el Instituto.

6. BIBLIOGRAFÍA

Zeiler, M. y J. Murphy (2010), Modeling Our World. The ESRI Guide to Geodatabase Concepts (2a. Edición), ESRI Press, Redlands, California, USA.