DESARROLLO DE UN PORTAL WEB COLABORATIVO PARA EL REGISTRO DE AVISTAMIENTOS DE CETÁCEOS EN EL LITORAL CATALÁN: CETCAT 2.0

Núria Julià Selvas y Joan Masó Pau
Centre de Recerca Ecològica i Aplicacions Forestals (CREAF)
Edifici C, Universitat Autònoma de Barcelona (UAB)
08193 Bellaterra (Barcelona)

Palabras clave: web 2.0, cetáceos, WFS-Transactional.

Resumen

En la web 2.0 los usuarios del servicio contribuyen aportando información que queda a disposición de los demás usuarios. Una manera de vehicular esto es estableciendo un portal colaborativo, en el que cada usuario, de manera individual, contribuye con la aportación de nuevos datos, de los cuales realiza su mantenimiento, construyéndose así una base de datos, muchas veces de calidad y volumen notables, a disposición de cualquiera, con un coste mínimo tanto de recursos humanos como económicos.
En este artículo se presenta un portal colaborativo basado en la web 2.0 y desarrollado para el registro de avistamientos "oportunistas" de cetáceos en el litoral catalán, con el que se pretende construir una gran base de datos de forma automatizada y usando los mínimos recursos; que posteriormente pueda ser usada para la realización de estudios de distribución espacial y temporal de las principales especies de cetáceos en Catalunya: el CetCat 2.0 (consultable en http://www.opengis.uab.cat/wms/cetcat).
El portal sigue una arquitectura cliente-servidor y los estándares de servicios Web Map Service (WMS), Web Feature Service (WFS) y WFS-Transactional del Open Geospatial Consortium (OGC). Ha sido implementado utilizando los lenguajes HTML, JavaScript, AJAX, C y SQL. Se trata de un portal de navegación, que incluye cartografía de referencia e imágenes de satélite, y que permite el registro, visualización y consulta de los avistamientos en tiempo real. El usuario rellena un formulario de entrada que recoge los datos y los envía al servidor mediante una petición de inserción que sigue los estándares GML y WFS-Transactional. Este protocolo es una extensión del WFS que define tres transacciones básicas para el mantenimiento de una estructura de datos: *insert*, *update* y *delete* de entidades (*features*) de una colección de objetos o capa. Los datos se envían al servidor de forma asíncrona mediante tecnología AJAX y siguiendo el método HTTP-POST, optimizando así el funcionamiento del portal. El servidor es una aplicación de tipo CGI que procesa las transacciones mediante estructuras XML-DOM, transformando los datos al formato requerido e insertándolos en una base de datos común. Todo este sistema cliente-servidor ha sido desarrollado siguiendo estándares de comunicación y de datos que dotan al portal de interoperabilidad, al tiempo que se ha realizado la implementación de forma flexible para que sea útil para otros usos.

1 Introducción

El proyecto Ninam es una plataforma dedicada al estudio de los cetáceos presentes en la Mediterránea norte occidental, que dispone de un catamarán dedicado a la recogida de datos de avistamientos de cetáceos que se producen sobretodo en la zona del Cap Norfeu y del Cap de Creus (en el nordeste de Catalunya). Uno de sus principales objetivos es realizar estudios para conocer la densidad poblacional y la distribución temporal y espacial de las diferentes especies de cetáceos presentes en el litoral catalán [13].

Existen varios factores medioambientales (fisicoquímicos, climatológicos y geomorfológicos), bióticos (depredación, comportamiento, distribución de la alimentación) y antropogénicos (presencia humana, actividades humanas, polución) que influyen en la distribución espacial y temporal de las especies [1]. Caracterizar la contribución relativa de cada uno de estos factores puede ser una tarea dura y difícil, sobretodo cuando se trata de estudiar especies marinas que se mueven en un territorio muy grande y desconocido como es el mar.
Los Sistemas de Información Geográfica (SIG) y la teledetección proporcionan varios datos y herramientas que se han demostrado muy útiles para modelizar estos factores y realizar estudios geospaciales. Existen varios estudios que intentan obtener patrones de distribución realizando diferentes análisis geospaciales (basados en regresiones múltiples, Kriging, etc.) a partir de datos de avistamientos y de la modelización de los diversos factores que afectan la distribución de las especies como la batimetría, las corrientes de agua y nutrientes, la salinidad, la temperatura superficial del agua, la densidad o la clorofila *a* (indicadora de producción primaria) [11]. En muchos casos estos factores son caracterizados a partir de imágenes satélite, como por ejemplo las obtenidas con el sensor MODIS embarcado en el satélite TERRA (EOS AM) y AQUA (EOS PM), que permiten obtener productos oceanográficos (clorofila y temperatura superficial del mar) con una resolución temporal de un día y una resolución espacial de un quilómetro [10].

Los sistemas de navegación de las embarcaciones también proporcionan herramientas indispensables para determinar la posición de los avistamientos. En la navegación costera, el método clásico de localización de la

embarcación se realiza mediante técnicas de posicionamiento basadas en el uso de cartas náuticas y la observación de demoras o distancias a puntos notables de la costa por medios visuales (faros, cabos, boyas, etc.) o por métodos electrónicos (demoras de radar a rácones, transpondedores, etc.). Actualmente, el método de navegación más usado para obtener la localización de la embarcación se basa en los sistemas de posicionamiento global, como el GPS o el GLONASS. El GPS, incorporado en la mayoría de barcos, es el sistema más usado para georeferenciar la posición en la que se dan los avistamientos.
Para este proyecto se parte de los datos de avistamientos cedidos por el proyecto Ninam. Estos datos son georeferenciados a partir de GPS, usando como localización del avistamiento las coordenadas del barco en el momento en el que éste se produce. Estas coordenadas tienen un error debido al propio método (GPS) y un error añadido debido a que los animales avistados no están en el propio barco sino alrededor de éste. Se considera que la precisión de la localización es suficiente ya que los animales observados están en continuo movimiento.

Un examen preliminar de la zona de estudio y de los datos de avistamientos del proyecto Ninam determinó que la obtención de un patrón de distribución espacial de los cetáceos presentes en la zona del Cap de Creus resultaba inviable. Era necesario ampliar la zona de estudio a toda Catalunya (ver figura 1) y obtener más datos debido a que la zona del Cap de Creus era relativamente pequeña en comparación con la resolución de los datos batimétricos, de las imágenes satélites y de los productos disponibles. En este sentido, la web 2.0 proporciona una herramienta muy útil para la obtención de estos nuevos datos.
La web 2.0 es un modelo de Internet basado en la construcción de webs orientadas a la interacción y a las redes sociales, en la que los usuarios del servicio, de manera individual, contribuyen con la aportación de información que queda a disposición de todos los otros usuarios; construyéndose de esta manera una gran base de datos de calidad y con un mínimo coste en recursos [14]. Este concepto se ha aplicado en diferentes ámbitos como la "Wikipedia", en el que los usuarios construyen una gran enciclopedia; los blogs o las wikis, u otros como "Flickr" en el que los usuarios comparten imágenes. En base a este concepto, se construyó un portal web colaborativo, que pretende implicar a toda la comunidad náutica, en el que es posible registrar los datos de los avistamientos realizados de forma casual por otras embarcaciones (de pesca, recreativas, etc.) presentes en la zona de estudio, conocidos como avistamientos "oportunistas". El aumento del volumen de datos y del ámbito permitirá en un futuro trabajar a escalas de menos detalle, en las que hay la información disponible de los factores que afectan a la distribución de las especies, realizar estudios geospaciales y obtener los patrones de distribución de los cetáceos presentes en Catalunya.

2 Área y especies de estudio

Catalunya es una región europea de la Mediterránea occidental situada en la costa nordeste de la Península Ibérica. El estudio realizado contempla todo el litoral catalán (aproximadamente unos 580 Km.) concentrándose sobretodo en la zona del Cap de Creus (situado en el Alt Empordà en el límite nordeste de Catalunya) (figura 1).

Figura 1: Localización de Catalunya (en verde) en Europa y en la Península Ibérica.

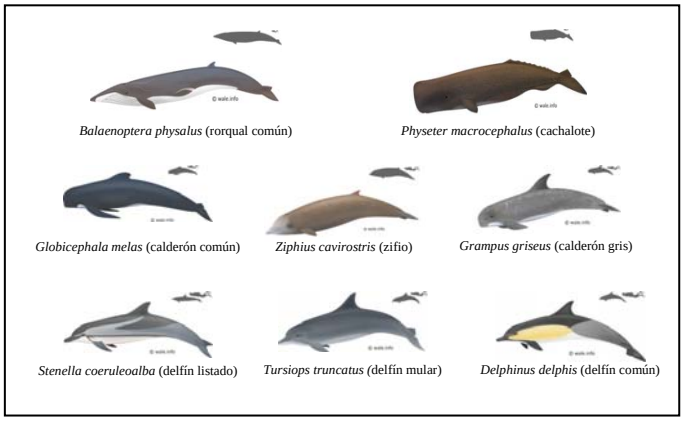

Figura 2. Principales especies de cetáceos presentes en el litoral catalán. (Dibujos extraídos de wale.info).

Se contemplan los avistamientos de cualquier especie de cetáceo (figura 2) que pueda estar presente en el litoral catalán pero sobretodo los de *Balaenoptera physalus* (rorcual común), *Physeter macrocephalus* (cachalote), *Globicephala melas* (calderón común o negro), *Grampus griseus* (calderón gris), *Stenella coeruleoalba* (delfín listado), *Delphinus delphis* (delfín común), *Tursiops truncatus* (delfín mular) y *Ziphius cavirostris* (Zifio o ballenato de Cuvier) [15,21].

3 Material, métodos y procedimientos realizados

3.1 Datos y material usado

Se parte de la información proporcionada por el proyecto Ninam sobre avistamientos de cetáceos en el Cap de Norfeu y Cap de Creus de les especies citadas anteriormente durante el período comprendido entre el 2004 y el 2008; de les bases cartográficas de referencia del Institut Cartogràfic de Catalunya (ICC), los datos del SIG de la Direcció General de Pesca i Acció Marítima del Departament d'Agricultura, Alimentació i Acció Rural (DAR) de la Generalitat de Catalunya y de los datos del satélite MODIS y Landsat de temperatura superficial y clorofila *a* del servidor *Thalassa* desarrollado por el Centre de Recerca Ecològica i Aplicacions Forestals (CREAF) [16].

Se ha usado el SIG *MiraMon Professional* [12] y sus herramientas de publicación de cartografía en Internet basadas en los protocolos de comunicación del *Open Geospatial Consortium* (OGC), herramientas gratuitas en el ámbito universitario de Catalunya, algunas de código abierto y que siguen de forma rigurosa varios estándares dotando al portal de interoperabilitat y facilitando el intercambio de la información con otros usuarios o servidores.

3.2 Publicación de les datos

Se ha unificado y uniformado los datos de avistamientos del proyecto Ninam de varios años, construyendo una sola base de datos relacional (figura 3), en la que se ha unificado conceptos y creado diccionarios para aquellos campos con valores repetitivos, como los nombres de las especies observadas, el estado de la mar [17], la dirección del viento, etc.

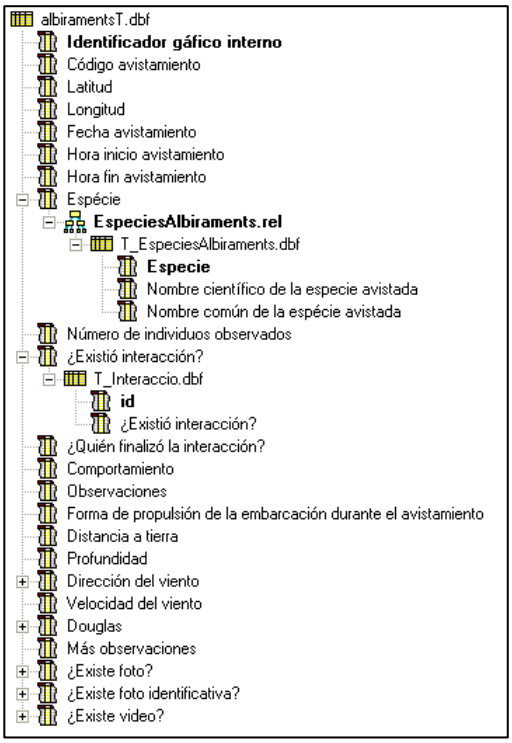

Figura 3: Estructura de la base de datos de avistamientos

A partir del *MiraMon* y su herramienta de creación de capas a partir de bases de datos (*bdpnt*) se ha dado un formato SIG a los datos de avistamientos. Se ha diseñado una simbología, basada en los dibujos de *whale.info* [21], que permite diferenciar las especies observadas. Finalmente, a partir de las herramientas de *MiraMon* de publicación de cartografía en Internet, se ha construido un servidor y navegador de mapas, en el que se ha publicado los datos de los avistamientos junto con la cartografía de referencia. Este portal es visible en http://www.opengis.uab.cat/wms/cetcat.

3.3 Portal de recogida de datos de los avistamientos oportunistas

Se ha diseñado e implementado un portal web colaborativo (web 2.0) (figura 4) integrado con el navegador de mapas y basado en estándares de servicios web interoperables de OGC, que además de permitir la visualización y consulta de los datos, permite recoger de manera integrada y automatizada los datos de avistamientos de cetáceos producidos en el litoral catalán, enriqueciendo y ampliando la base de datos del proyecto Ninam. Este portal incluye los datos de los avistamientos pero también contiene un conjunto de cartografía de referencia, cartografía marina, ambiental y pesquera y imágenes satélite que ayudan al usuario a situarse, además de proporcionar información adicional y útil para la realización de estudios geospaciales.

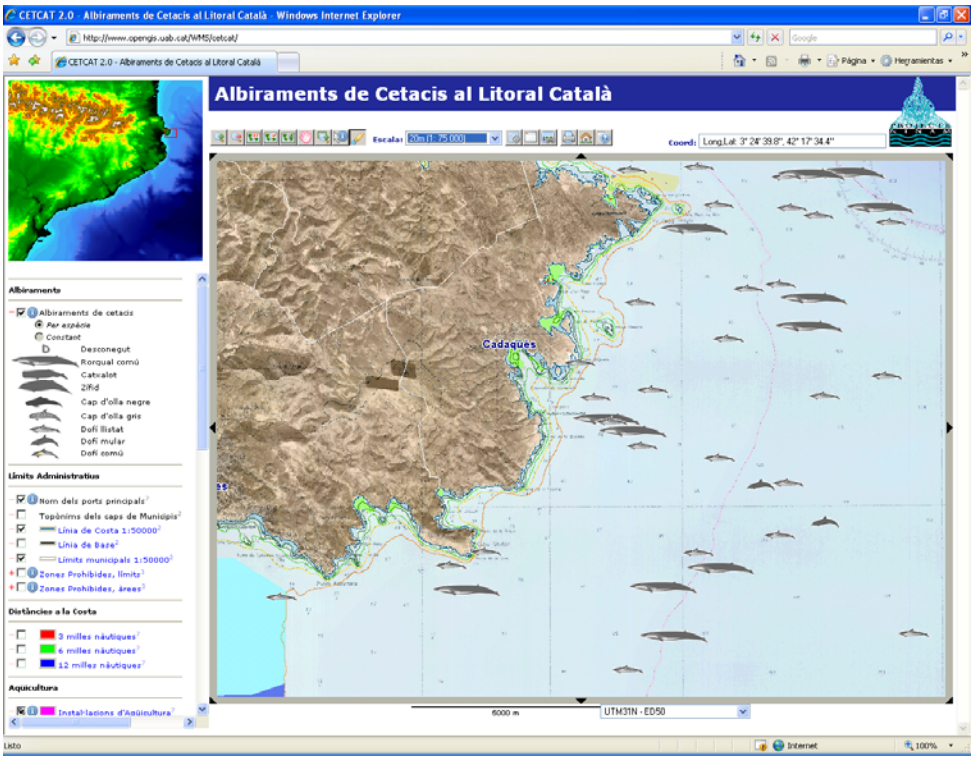

Figura 4: Portal CETCAT 2.0 para el registro de avistamientos de cetáceos del litoral catalán.

Este portal se basa en una estructura cliente-servidor y en los protocolos estándares Web Map Service (WMS) [2] y Web Feature Service (WFS) [18,19] de OGC; y ha sido diseñado y programado como una extensión del servidor y navegador de mapas de MiraMon (figura 5). Se ha realizado una implementación totalmente basada en estándares cuando se podía haber hecho una implementación con una sintaxis propia, seguramente más simple de diseñar y más fácil de llevar a la practica, pero se optó por hacerlo siguiendo los estándares para dotar al portal de interoperabilidad y hacerlo más accesible a todos.

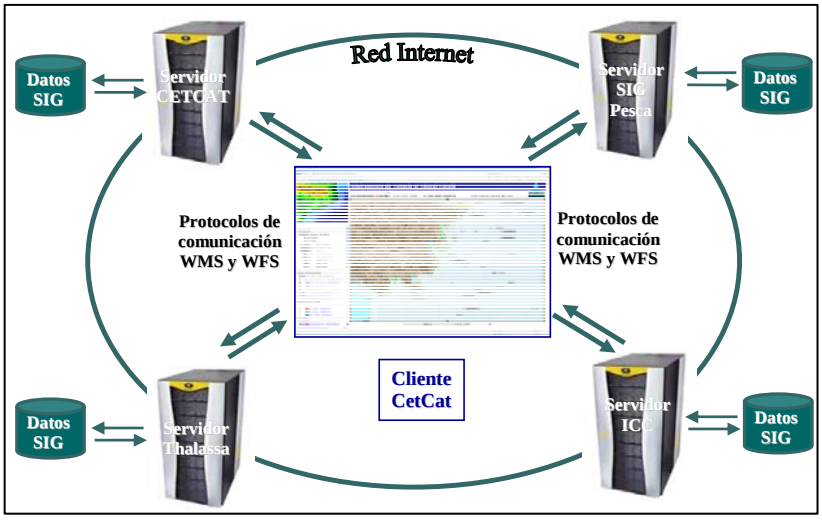

Figura 5: Estructura del Portal. Arquitectura cliente-servidor

Se ha diseñado e implementado un formulario interactivo de entrada de datos, con campos y selectores de valores prefijados que ayudan al usuario a determinar la información más relevante de los avistamientos. El punto o coordenada en el que se ha producido el avistamiento se establece interactivamente navegando sobre la cartografía de referencia y ayudándose de las diferentes herramientas de navegación disponibles en el portal, como acercar, alejar, zoom por ventana, pam, ir a coordenada, etc. El usuario puede escoger la proyección cartográfica sobre la que quiere navegar; las coordenadas del avistamiento se muestran y se envían al servidor en la proyección escogida. El servidor será el encargado de realizar las transformaciones de proyección cartográfica necesarias para que los datos sean almacenados en la proyección Universal Transversa de Mercator (UTM) huso 31 hemisferio Norte con Datum ED50 (parámetros ICC), proyección escogida ya que es la más usada y una de las oficiales en el ámbito de Catalunya.

Con este sistema se ha conseguido una recogida de les datos de forma coordinada y automatizada que no necesita de ninguna transformación posterior, los datos son creados directamente en un formato SIG, facilitando que sean inmediatamente visibles a los demás usuarios y que estén a punto para realizar cualquier tipo de análisis.

Se ha diseñado e implementado un sistema que permite simbolizar los objetos digitalizados en función del valor de uno de sus campos o atributos temáticos, de esta manera es posible simbolizar los avistamientos en función de la especie. Además cada capa de objetos digitalizados puede ser simbolizada en más de un estilo de visualización. En el caso de los avistamientos se ha implementado una simbolización con iconos grandes que intentan representar la forma y colores de cada especie, y una simbolización constante para facilitar la visualización en escalas de menos detalle.

3.4 Interfaz cliente-servidor

La interfaz o protocolo de comunicación entre el cliente y el servidor está basada en el servicio *WFS-Transactional* del OGC. Este protocolo es un extensión del WFS que define tres transacciones básicas para el mantenimiento de una estructura de datos: *insert*, *update* y *delete* [19]; es decir, creación de nuevos elementos, entidades, fenómenos (*features*) o objetos de una colección o capa, edición o modificación de algunas de les características de los objetos existentes y eliminación de los objetos.

Los elementos geográficos siguen un modelo de datos basado en objetos, definidos a partir de GML (*Geography Mark-up Language*), lenguaje que establece un conjunto de reglas basadas en el XML (*Extensible Markup Language*) que permite definir las características geográficas, topológicas y temáticas de las entidades o objetos [3,9]. Es un modelo de datos muy flexible, que permite definir cualquier capa geográfica, pero precisamente debido a esta gran flexibilidad puede ser muy complejo y difícil de implementar. Por este motivo, el OGC definió tres perfiles de GML en un documento llamado *GML simple features profile*, el más simple de todos, llamado *Level 0* e identificado con el subtipo MIME “gml/3.1.1/profiles/gmlsf/1.0.0/0” [20], define un modelo de datos muy simple, pautado, y por lo tanto, más fácil de usar. Este perfil, más interoperable, ha sido escogido para realizar las transacciones, pero no ha sido suficiente para la visualización y consulta de los objetos en el portal web debido a que los atributos temáticos sólo se pueden especificar indicando su nombre y valor.
Se ha definido un perfil propio, llamado *MiraMon attributes profile* e identificado con el subtipo MIME “gml/3.1.1/profiles/miramon/1.0.0/attributes” basado en los estándares WFS [18,19], GML [3] y los MIME-types [6]. Este perfil, más complejo que el anterior, permite definir los atributos temáticos de los objetos de una forma similar a la que se hace en una base de datos y con todos sus metadatos. Los atributos de este perfil siguen un esquema que se puede consultar en: http://www.miramon.uab.cat/ogc/schemas/atribut/1.0.0/Atribut.xsd y permite especificar características como la descripción del campo, el tipo, el tamaño, o si es o no visible entre otros; todas ellas características básicas para poder tratar los datos en un SIG.

3.5 El cliente

El navegador de Mapas de *MiraMon* ha sido ampliado para actuar como un cliente web 2.0 colaborativo en HTML, *JavaScript* y AJAX *(Asynchronous JavaScript And XML)* [7].

Los datos de avistamiento se almacenan en el servidor, en una capa vectorial de objetos de tipo punto de *MiraMon*. El navegador o cliente obtiene estos datos haciendo peticiones de tipo *WFS-GetFeature* a un servicio web que reside en http://www.opengis.uab.cat/cgi-bin/cetcat/MiraMon5_0.cgi. Éste transforma los datos y los envía al cliente en formato WFS-GML.

Se ha definido una clase en *JavaScript* [22] que permite agrupar todas las propiedades de una capa de tipo WFS: *CreaObjecteDigitalitzat*. Esta clase contiene información general de la capa como el servidor donde reside, el nombre y la descripción de la capa, sus características de visualización, sus estilos de simbolización y consulta, y los objetos que la conforman con sus datos geográficos y temáticos. De esta manera, una capa digitalizable se

define, en un documento de configuración del navegador, separado del código de la aplicación (config.js), como un objeto de la clase *CreaObjecteDigitalitzat*.

Las peticiones *WFS-GetFeature* al servidor se realizan asincrónicamente usando tecnología AJAX. Para optimizar las peticiones desde el cliente, se ha diseñado un sistema basado en dividir el espacio de navegación en teselas regulares de manera que los objetos de la capa se solicitan cuando se necesitan mientras el usuario navega sobre el espacio delimitado por cada una de les teselas. Los datos recibidos del servidor son cargados sobre una estructura XML-DOM (*Document Object Model*) [22], procesadas y almacenadas a memoria en objetos de la clase *CreaObjecteDigitalitzat*.

Con el fin de que el usuario pueda introducir o modificar avistamientos, se ha diseñado un formulario de entrada de datos en *JavaScript*, siguiendo un esquema de *layers* o divisiones al igual que el resto del portal, basado en la estructura interna de la base de datos, en el que el usuario puede indicar: las coordenadas del punto en el que se ha realizado el avistamiento, la fecha y hora del avistamiento, la especie observada, el comportamiento de los animales, características ambientales y físicas en el momento del avistamiento como las condiciones de viento, el estado de la mar, la profundidad, la distancia a la costa y otros.

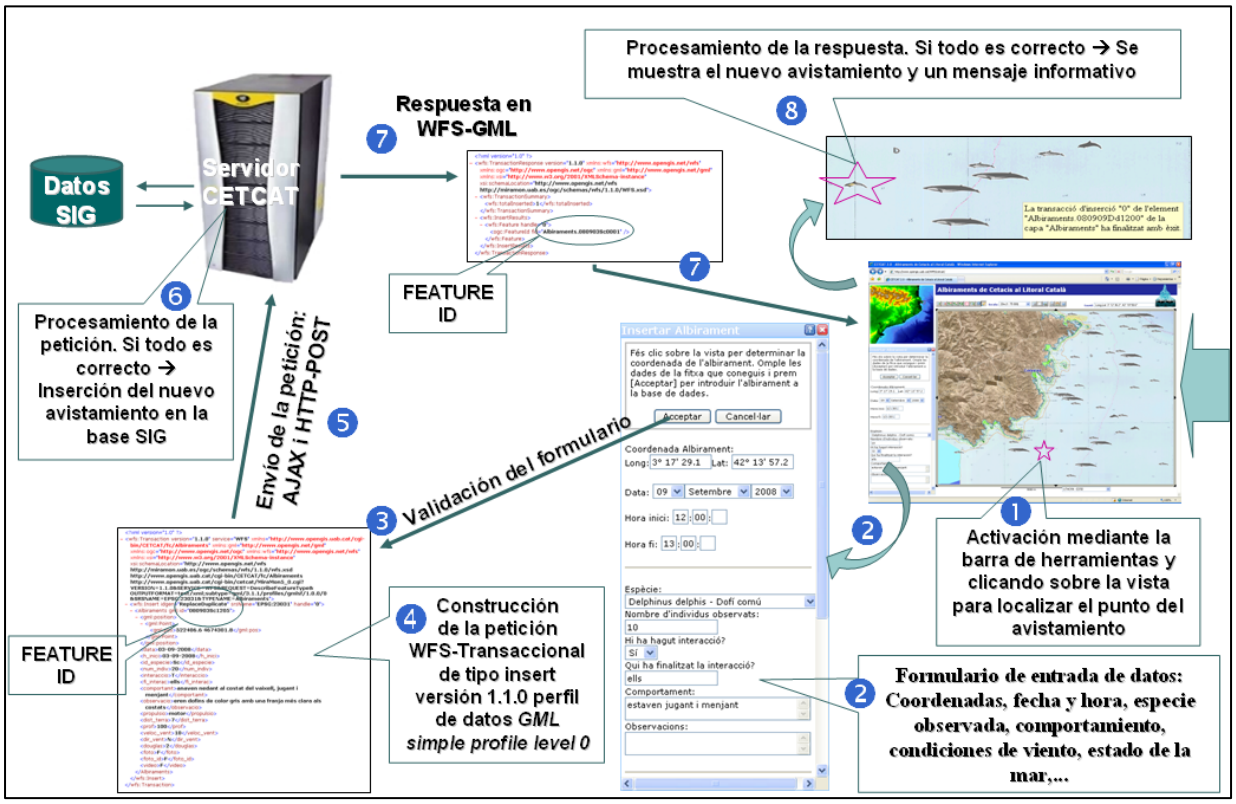

Figura 6: Flujo de datos durante la inserción de un nuevo avistamiento.

El formulario de entrada de datos se activa seleccionando la opción de edición de la barra de herramientas del navegador y haciendo clic sobre la vista. Al realizar el clic, las coordenadas del avistamiento son actualizadas y mostradas en el formulario en el sistema de referencia seleccionado para la navegación. Además de los campos de la base de datos, el formulario incluye los valores de aquellos campos que vienen definidos por diccionarios. Al aceptar, se valida el contenido del formulario y se construye una petición de inserción *WFS-transactional* de tipo *insert*. Esta petición es enviada asíncronamente al servidor usando tecnología AJAX y siguiendo el protocolo HTTP (*Hypertext Transfer Protocol*) mediante el método POST.

El navegador almacena la información de la transacción necesaria para su identificación: *handle* o identificador de la transacción, estado (finalizada o no, con o sin éxito), identificador del objeto (construido a partir de la fecha y hora del avistamiento y de la especie avistada), en objetos de la clase *CreaTransaccio* definida en *JavaScript*. Cuando se recibe la respuesta del servidor y se identifica la transacción, se muestra un mensaje informativo, en segundo plano, indicando el estado de la transacción, y si es necesario y todo ha funcionado correctamente, se muestra el punto con el nuevo avistamiento insertado (figura 6).

3.6 El servidor

El servidor es una aplicación de tipo CGI (*Common Gateway Interface*) programada en C, como una extensión del servidor de mapas de *MiraMon* que sabe entender y contestar peticiones *WFS-Transactional* de la versión 1.1.0 del estándar de tipo *insert* de objetos vectoriales de tipo punto.

Las transacciones llegan al servidor mediante el método POST y en formato WFS-GML. Son cargadas sobre una estructura XML-DOM, y validadas, parseadas, ejecutadas y respondidas según el esquema definido por el estándar WFS (figuras 7 y 8; en las que se muestra un ejemplo de una petición y una respuesta de una transacción).

Figura 7: Ejemplo de una petición de transacción de inserción de un avistamiento en XML subtipo GML perfil simple gmlsf/0

Figura 8: Ejemplo de una respuesta a una petición de transacción de inserción de un avistamiento finalizado con éxito en XML subtipo GML perfil simple gmlsf nivel 0

Según el estándar WFS [18,19], una capa es una colección de fenómenos (*features*) del mismo tipo que tiene un identificador único llamado *TypeName* que se puede asimilar con el nombre de la capa. Cuando el servidor recibe la petición, identifica la capa por este nombre y a partir de él, localiza en el catálogo de capas sus ficheros vectoriales.

El *FeatureId* es un identificador único de un objeto de una capa (lo que en *MiraMon* se conoce como identificador identidad). Este puede ser indicado o no en la petición de inserción. Según el estándar existen 3 modos posibles: *GenerateNew*, *UseExisting* y *ReplaceDuplicate* [19]. Todos ellos han sido implementados en este servidor.
Cuando en una petición de inserción se indica *GenerateNew*, se genera un nuevo identificador de forma aleatoria. Cuando se indica *UseExisting*, se valida que el identificador no exista y en caso de que exista, se envía una respuesta con un mensaje de error, indicando que la transacción no se ha realizado. Por último, cuando se indica *ReplaceDuplicate*, se valida el identificador, y en caso de que exista se intenta generar uno de nuevo, primero a partir del identificador indicado añadiéndole una dígito y si continua siendo no válido, generando una de nuevo de forma aleatoria.

Los objetos son validados y parseados según su esquema de definición en XML indicado en la petición y el perfil simple nivel 0 de GML de la versión 3.1.1. Se considera que el perfil *MiraMon*, muy útil para las peticiones de

WFS-GetFeature, contiene información innecesaria para la transacción que la haría más pesada y larga de procesar.

4 Discusión

Durante el diseño y la implementación se han detectado diversos problemas, que se detallan a continuación en los siguientes párrafos, algunos de ellos se han resuelto, otros se está estudiando la mejor manera de hacerlo; un vez resueltos se prevé que en poco tiempo se pueda implementar el resto de transacciones como mínimo para capas vectoriales de tipo punto.

Se ha visto que es necesario desarrollar un mecanismo de registro de usuarios. Cada usuario registrado debe poder entrar nuevos avistamientos, y si es necesario, posteriormente con una identificación previa, modificarlos o eliminarlos. Este registro e identificación de usuarios puede realizarse de manera automática, por ejemplo mediante direcciones IP (*Internet Protocol)*, o a través de un formulario, dónde además de registrarse para obtener un nombre de usuario y contraseña, se pueda indicar otros datos útiles para la realización de análisis posteriores, como por ejemplo el tipo de embarcación, la actividad a la que se dedica, horas de navegación, etc. Se han previsto 3 perfiles de usuario, el *usuario anónimo* que sólo podrá visualizar y consultar los datos navegando por el portal, el *usuario registrador de avistamientos* que podrá insertar, modificar y eliminar los avistamientos de los cuáles sea propietario, es decir, que haya insertado previamente, y el *usuario administrador o moderador*, con derecho a todo, incluso a modificar o eliminar avistamientos de otros usuarios, que será el encargado de controlar el buen funcionamiento del portal, los usuarios, etc., y que enviará las notificaciones que sean necesarias. Una vez realizado este control de usuarios tendrá sentido dotar al portal del resto de transacciones.

Otro punto importante ha resolver es el sistema de copias de seguridad y de versiones. Es necesario implementar algún sistema, que además de permitir la recuperación de los datos en caso de pérdida o caída del sistema, permita, si es necesario, deshacer algunas modificaciones no deseadas, lo que se conoce como *versioning*. Una posibilidad es implementar un sistema similar al que usan los grandes gestores de bases de datos, en los que se hacen copias enteras de toda la base, y además se crean tablas en las que se registran todas las transacciones realizadas, sean del tipo que sean, en orden cronológico; permitiendo recuperar los datos en un estado concreto.

Se ha realizado un esfuerzo para hacer un diseño flexible, que permite usar las herramientas implementadas para otras finalidades sin requerir mucho trabajo. No se ha implementado un servicio *WFS-Transactional* específico para avistamientos de cetáceos, sino que se ha realizado un servicio que puede ser usado para realizar transacciones con cualquier capa de puntos. Por ejemplo se podría construir un portal para el registro de avistamientos de pájaros, o uno para el registro de hallazgos arqueológicos. Esta flexibilidad esta totalmente resuelta en el servidor pero no en el navegador; los principales problemas son debidos por los atributos temáticos.
Las capas podrían tener una estructura de datos temáticos muy compleja, con enlaces con diferentes bases de datos relacionales. Los perfiles de GML usados, tanto el simple de nivel 0 (*gmlsf level 0*) como el perfil de *MiraMon*, permiten especificar qué atributos geográficos tienen los objetos y sus relaciones topológicas, pero no contemplan toda la complejidad de los datos temáticos, no contemplan las relaciones entre los diferentes atributos.
En los perfiles usados, los atributos temáticos son características con un nombre y un valor ligado a un identificador, características que pueden ser repetidas o no aparecer, que se obtienen siguiendo todas las tablas ligadas al identificador del objeto.
Esta estructura compleja de los datos no es un problema para responder a peticiones de tipo *WFS-GetFeature*; ya que se realizan *left joins* con el identificador del objeto, de manera que se construye una tabla única con todos los atributos en la que hay uno o varios registros (si alguno de los campos tiene más de un valor) para cada identificador de objeto.
La tabla principal es la tabla que nos relaciona cada uno de los objetos gráficos con sus datos temáticos organizados en una estructura de base de datos relacional. Para implementar la transacción de inserción, se supone que todos los atributos temáticos indicados en ella, pertenecen a la tabla principal y que el resto de atributos temáticos se puede obtener a partir de estos, ya que forman parte de diccionarios de datos.
Esta solución adoptada ha permitido resolver el problema de los atributos, al menos en parte, de del punto de vista del servidor pero no del navegador.

El navegador es flexible en lo que se refiere a estructuras internas de datos, funciones de construcción de peticiones y de parseado de las respuestas del servidor, funcionamiento de activación de la inserción de nuevos elementos, etc., pero no es flexible en la construcción de los formularios, es decir, su contenido es específico para insertar los avistamientos de cetáceos en la base. Con los protocolos WFS y GML no hay suficiente para especificar toda la información necesaria para la construcción de estos formularios de forma totalmente automatizada. Se ha pensado en desarrollar peticiones nativas o propias del servidor de *MiraMon*, basadas en estándares de datos, como ISO-19110 [4] u otros que están en estudio. La petición, que llamaremos, *DescribeThematicAttributteType* debe permitir solicitar todas las características de un atributo de una capa, características similares a las descritas en el perfil *gml* de *MiraMon* y otras como la posición del atributos en la

estructura de la base de datos, por ejemplo, a partir de direcciones únicas, y también, la relaciones del atributo con otros atributos, ya que en muchos el valor de un atributo condiciona el de otros; por ejemplo, el echo de que se produzca interacción o no con los animales hace que tenga sentido o no conocer como ha sido está interacción. También se debe definir una petición que permita obtener los diccionarios de datos de aquellos campos que tienen un conjunto de valores finitos y conocido. Nos consta que el grupo de trabajo del WFS se está planteando esté y otros problemas, y esperamos que en la nueva versión existan soluciones.

Todo esto permitirá automatizar y hacer totalmente flexible la construcción de los formularios de transacciones del navegador; pero también permitirá usar el perfil *gmlsf level 0* en les peticiones de *WFS-GetFeature,* ya que dispondremos de otro mecanismo para obtener los atributos de los atributos, y disminuir el volumen de información transmitida del servidor hacia el navegador.

5 Conclusiones

El portal web colaborativo permite obtener datos de calidad, en el que cada usuario, de manera individual, aporta nuevos datos y realiza su mantenimiento, construyendo de esta manera una gran base al alcance de todos, con un gasto mínimo en recursos humanos y económicos, ya que se trata de aprovechar la actividad normal de las embarcaciones presentes en el litoral catalán.

El hecho de que la recogida de los datos sea totalmente informatizada, automatizada y directamente en un formato SIG, permite que los datos estén siempre a punto para ser tratados y analizados. Este hecho también disminuye los recursos necesarios y el tiempo de procesamiento; hasta el momento los datos de los avistamientos oportunistas eran recogidos en formularios de papel que se debían procesar e introducir en un SIG.

El sistema de muestreo y recogida de los datos del proyecto Ninam es adecuado y suficientemente detallado, pero son necesarios muchos recursos ya que con un solo barco no se puede cubrir de forma exhaustiva todo el área de estudio. Por esto, los datos recogidos hasta el momento no se han demostrado suficientes para hacer análisis geospaciales. La web 2.0 es una idea para involucrar a mucha gente y obtener datos sin necesidad de invertir más recursos propios, aprovechando los destinados a otras actividades.

El portal oculta la complejidad tecnológica al usuario final. El desarrollo del proceso se ha llevado a cabo usando estándares OGC, aunque se ha comprobado que un solo estándar de datos y de comunicaciones no es suficiente para realizar un proyecto de estas características de forma completa. Es necesario realizar una búsqueda exhaustiva y combinar varios estándares y metodologías para realizar un entorno atractivo y dinámico que permita resultados satisfactorios. En este portal se ha combinado los estándares de servicios WMS, WFS, WFS-Transactional, los estándares de datos GML, XML, GML-Simple Feature, los protocolos HTTP con los métodos GET y POST, y otros. Se ha usado varios lenguajes y técnicas de programación, el C, el JavaScript, el AJAX, el HTML, el SQL, etc.

En el lado del servidor, las estructuras de datos de *MiraMon*, y en concreto la de capas vectoriales de objetos de tipo punto, se han demostrado suficientes para el volumen de datos tratado.

En el lado del cliente, los navegadores de mapas de código abierto de *MiraMon* permiten que cualquier persona, con suficientes conocimientos en JavaScript, pueda implementar extensiones y proponer que éstas pasen a formar parte de las distribuciones del código del navegador [8].
El AJAX, el JavaScript, el HTML y los formularios web han permitido realizar un entorno colaborativo fácil de entender y de utilizar, donde publicar los datos en tiempo real sobre una cartografía de referencia.

Este portal es una aproximación simplista, de un SIG verdadero que funciona a través de la web con arquitectura cliente-servidor y que no requiere de la instalación de ninguna aplicación. El portal desarrollado ofrece las funcionalidades de un pequeño SIG en el que se puede descargar objetos, interpretarlos, visualizarlos, mostrar sus datos, editarlos e imprimirlos en mapas.

Referencias

[1] Azzellino, A., Gaspari, S., Airoldi, S. and Nani, B. 2008. Habitat use and preferences of cetaceans along the continental slope and the adjacent pelagic waters in the western Ligurian Sea. Deep Sea Research Part I, Vol. 55, pp. 296-323.

[2] de la Beaujardiere, Jeff. 2004. OpenGIS Implementation Specification. OGC 03-109r1. OGC Web Map Service Interface Version 1.3.0.

[3] Cox, S., Daisey, P., Lake, Portele, R.C. and Whiteside, A. 2004. OpenGIS Implementation Specification. OGC 03-105r1. Geography Markup Language (GML) Implementation Specification Version 3.1.1.

[4] Draft International Standard: Geographic information – Methodology for feature cataloguing. ISO/DIS 19110. 2001. Technical Committee 211. International Organization for Standardization.

[5] Eenisse, Matthew. 2006. Build Your Own AJAX Web Applications. SitePoint Pty. Ltd.

[6] Freed, N. and Borenstein, N. 1996. Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies: RFC 2045. Innosoft, First Virtual Holdings.

[7] Gutiérrez, J.M., Palacios, F., Gutiérrez de Mesa, J.A. 2003. Estándar XML y Tecnologías Asociadas. Danypress.

[8] La Farga Web Site: Portal del navegador de mapas de código abierto de MiraMon. Agosto 2008: https://projectes.lafarga.cat/projects/navmapesmiramon

[9] Lake, R., Burggraf, D.S., Trninic, M. and Rae. L. 1994. Geography Mark-Up Language: foundation for the geo-web. John Wiley & Sons, Ltd.

[10] MODIS Web Site. Agosto 2008: http://modis.gsfc.nasa.gov

[11] Monestiez, P. et al. 2006. Geostatistical modelling of spatial distribution of *Balaenoptera physalus* in the Northwestern Mediterranean Sea from sparse count data and heterogeneous observation efforts. Ecological Modellin, Vol. 193, pp. 615-628.

[12] Pons, X. 2004 MiraMon. Sistema d'Informació Geogràfica i software de Teledetecció Centre de Recerca Ecològica i Aplicacions Forestals, CREAF. Bellaterra. ISBN: 84-931323-4-9

[13] Proyecto Ninam Web Site. Agosto 2008: http://www.projecteninam.org

[14] Scharl, A. and Tochtermann, K. 2007.The Geospatial Web. How Geobrowsers, Social Software and the Web 2.0 are Shaping the Network Society. Springer.

[15] SEC: Sociedad Española de Cetáceos Web Site. Agosto 2008: http://www.cetaceos.com

[16] Serral, I. 2007. Implementació d'un servidor de mapes per a l'ús marítim català (Thalassa): concentració de clorofil·la i temperatura superficial del mar. Màster en Teledetecció i SIG: De les imatges de satèl·lit a la publicació de cartografia a Internet: 8ena edició: 2006-2007

[17] Títulos náuticos Web Site: http://www.titulosnauticos.net/meteorologia/index.htm?/meteorologia/douglas.htm

[18] Vretanos, Panagiotis A. (ed). 2002. OpenGIS Implementation Specification.OGC 02-058. Web Feature Service Implementation Specification Version 1.0.0.

[19] Vretanos, Panagiotis A. (ed). 2005. OpenGIS Implementation Specification.OGC 04-094. Web Feature Service Implementation Specification Version 1.1.0.

[20] Vretanos, Panagiotis A. (ed). 2006. OpenGIS Implementation Specification Profile.OGC 06-049r1. Geography Markup Language (GML) simple features profile Version 1.0.

[21] Wale.Info. Die Welt der Wale und Delfine Web Site. Agosto 2008: http://www.wale.info

[22] W3 Schools Web Site. Agosto 2008: http://www.w3schools.com