

MIGRACIÓ I INTEGRACIÓ D'INFORMACIÓ GIS AL WEB CORPORATIU DEL PARC
DE COLLSEROLA MITJANÇANT APIS DE GOOGLE

Projecte Final

Màster en Tecnologies de la Informació Geogràfica

Departament de Geografia

Universitat Autònoma de Barcelona



Autor: Eduard Cuscó i Puigdellívol

Tutors: Laura Sala i Martín (UAB)

Raimon Reventós i Rovira (CPC)

Febrer 2011

Organitzador:



Entitat col·laboradora:



*Al Raimon i a la Laura per l'esforç i recolzament
constants de principi a final de la col·laboració.*

Paraules Clau: API, Google Maps, Earth, Visual Web Developer, ASP, Collserola, kml.

Resum

El present document exposa el projecte final del dotzè Màster en Tecnologies de la Informació Geogràfica, fruit de la col·laboració entre el Departament de Geografia de la Universitat Autònoma de Barcelona i el Consorci del Parc de Collserola.

Amb la finalitat de posar fàcils les coses al personal tècnic del Parc alhora de gestionar la informació espacial de l'ara ja Parc Natural, i amb l'objectiu divulgatiu de difondre al màxim llur patrimoni natural i humà, férem ús de les interfícies de Google Earth i Google Maps com a estàndards d'informació cartogràfica actuals i emergents per a programar un sistema d'informació geogràfica web. Mitjançant els llenguatges de programació ASP i JavaScript es dissenyen i programen eines i funcions per a la posada en marxa d'estructures de menús per permetre seleccionar i destriar àgilment la informació cartogràfica que es disposa sobre les interfícies.

Per tal de dur-ho a terme, cal primer elaborar informació cartogràfica en format kml de Google, a través del qual establim procediments per permetre la vinculació entre elements gràfics i dades alfanumèriques, imatges, enllaços i d'altres, a través de la programació de vinyetes dinàmiques emergents.

El resultat final que se n'ha derivat representa una sèrie d'aplicacions magnífica per tal de complementar el nou web corporatiu del Parc de Collserola. Es presenten doncs així a continuació, els procediments i tècniques per la elaboració del sistema d'informació geogràfica web.

Palabras clave: API, Google Maps, Earth, Visual Web Developer, ASP, Collserola, kml.

Resumen

El presente documento expone el proyecto final del duodécimo Máster en Tecnologías de la Información Geográfica (Master en Tecnologías de la Información Geográfica), fruto de la colaboración entre el Departament de Geografia de la Univsersitat Autònoma de Barcelona i el Consorci del Parc de Collserola.

Con la finalidad de poner fáciles las cosas al personal técnico del parque para gestionar la información espacial del ya parque Natural de Collserola, y con el objetivo de difundir al máximo su patrimonio natural y humano, hacemos uso de las interfaces de Google Earth y Google Maps como estándares de información cartográfica actuales y emergentes para programar sistemas de información geográfica web. Mediante lenguajes de programación ASP y JavaScript, se diseñan y programan herramientas y funciones para la creación de estructuras de menús que permiten seleccionar y discernir ágilmente la información cartográfica que se dispone sobre las interfaces.

Para llevarlo a cabo, hizo falta no obstante elaborar información cartográfica en formato kml de Google, a través de la cual establecemos procedimientos que permiten la vinculación entre elementos gráficos y datos alfanuméricos, imágenes, enlaces entre otros, a través de la programación de viñetas dinámicas emergentes.

El resultado final que obtenemos representa una serie de aplicaciones magnífica para complementar la nueva web corporativa del parque de Collserola. Se presentan pues a continuación, los procedimientos y técnicas para la elaboración del sistema de información geográfica web.

Keywords: API, Google Maps, Earth, Visual Web Developer, ASP, Collserola, kml.

Abstract

This document presents the final draft of the twelfth Máster en Tecnologies de la Informació Geogràfica (Master in Geographic Information Technologies), a collaborative internship between the Department of Geography of the Universitat Autònoma de Barcelona and the Collserola Park Consortium.

In order to make things easier to park staff to manage spatial information of the Collserola Natural Park, and with the aim of disseminating natural and human heritage, we use the interfaces of Google Earth and Google Maps as standards of mapping information current and emerging GIS program web. Programming languages ASP and JavaScript, we design tools and functions for creating menu structures to select and nimbly cartographic information available on the interfaces.

To accomplish this, however it take to develop mapping information in kml format for Google, through which would establish procedures that would allow the connection between graphics and alphanumeric data, images, links, among others, through programming dynamic balloons.

The final result we get represent different applications to complement the new corporate Collserola website. Now, we show procedures and techniques for the development of the geographical information system web.

ÍNDIX	Pàgina
1. Introducció.....	9
Marc Institucional i de treball: El Consorci del Parc de Collserola.....	9
1.2 Origen i antecedents del projecte.....	10
2. Objectius.....	12
3. Anàlisi de requeriments.....	14
3.1 Requeriments funcionals.....	14
3.2 Requeriments de software i llenguatges.....	15
3.3 Solució metodològica i criteris de selecció.....	17
3.4 Usuaris.....	19
4. Planificació inicial i Arquitectura.....	20
4.1 Planificació inicial dels treballs.....	21
4.2 L'Arquitectura.....	23
5. Tractament i estructuració de les dades.....	24
5.1 Edició i elaboració d'informació cartogràfica.....	24
5.2 Transformació i adaptació d'informació geogràfica a Google.....	26
5.2.1 Transformació a kml mitjançant ArcMap 10.....	26
5.2.2 Transformació a kml mitjançant programari lliure.....	27
5.2.3 WGS84 el sistema de coordenades de Google.....	32
5.2.4 Edició de cartografia mitjançant Google Earth, i els kmz.....	37
5.3 Protocols de transferència d'arxius al servidor públic d'informació.....	39
5.3.1 Client FileZilla.....	39
6. Implementació de l'aplicació, migració i integració de la informació GIS.....	43
6.1 Aproximació tecnològica.....	43
6.1.1 Interfícies de programació d'aplicacions Google.....	43

6.1.1.1 La interfície Earth.....	44
6.1.1.2 Interfície Maps.....	48
6.1.2 Entorn, disseny i tecnologia Visual Web Developer.....	50
6.1.2.1 El Disseny ASP.....	50
6.1.2.2 El Disseny d'un l'arbre.....	50
6.1.2.3 Disseny de l'opció esborrar capes.....	52
6.2.1 Programació javascript de les funcions de l'arbre.....	52
6.2.1.1 La càrrega.....	52
6.2.1.2 La invisibilització de les capes.....	54
6.2.1.3 La funció d'esborrar.....	55
6.2.1.4 La clau final: La funció OnTreeNodeChecked.....	55
6.2.2 La funció de càrrega de cartografia a Google Maps.....	58
6.3 Programació de vinyetes emergents.....	59
6.3.1 El disseny i entrellat d'un KML.....	60
6.3.2 Mètodes per a l'obtenció de vinyetes emergents de qualitat.....	64
6.3.2.1 Programació de vinyetes mitjançant plantilles.....	65
6.3.2.2 Programació de vinyetes mitjançant variables.....	66
7. Explotació i presentació de resultats.....	68
7.1 L'apartat SIG i Google Earth.....	69
7.2 L'apartat de 'Guia de serveis' i Google Maps.....	72
8. Més enllà de la col·laboració.....	75
9. Conclusions.....	77
10. Bibliografia i webgrafia.....	79

ÍNDEX DE FIGURES

Pàgina

Fig 1. Cartografia en pdf de l'antic web encara per renovar. Font: www.parccollserola.net/	11
Fig 2. Mostra de l'estat en que es trobava l'aplicació. S'havia començat a migrar la informació SIG però no de forma ordenada. Font: Elaboració pròpia.....	13
Fig. 3. Quadre de planificació. Font. Elaboració pròpia.....	31
Fig 4. Proposta d'arquitectura. Font: Elaboració pròpia.....	22
Fig 5. Els atributs del restaurant de la Fàbrica Sànsen a Sant Feliu de Llobregat en format shapefile a ArcView. Font: Elaboració pròpia.....	25
Fig 6. La transformació i llurs múltiples opcions. Font: Elaboració pròpia.....	26
Fig 7. Visualització del shape "restaurants" dins de GPS TrackMaker, amb la DLL incorporada. Font: Elaboració pròpia.....	27
Fig 8. L'opció Keyhole Markup Language a l'hora de guardar la capa vectorial i el sistema de coordenades. Font: Elaboració pròpia.....	28
Fig 9. La capa restaurants ara en format kml, amb el marker i l'etiqueta per defecte. Font: Elaboració pròpia.....	29
Fig. 10 Les opcions de transformació de la DLL de Mapwindow, com s'observa es poden seleccionar els atributs escollits per a salvar. Font: Elaboració pròpia.....	30
Fig 11. Un exemple de la quantitat d'atributs que es poden emmagatzemar i que prèviament han estat elaborats mitjançant Arcview 3. Font: Elaboració pròpia.....	30
Fig 12. L'advertiment de shape2earth. Font: Elaboració pròpia.....	31
Fig 13. Exemple de desviament resultant de la georeferenciació del consorci. Com s'observa és d'uns centenars de metres. Font: Elaboració pròpia.....	32
Fig 14. Opció DLL d'ArcMap amb opció a condicionar l'offset. Font: Elaboració pròpia.....	33
Fig 15. Les opcions d'edició d'Earth. Font: Elaboració pròpia.....	34
Fig 16. Columnes resultants de la resta en excel. Font: Elaboració pròpia.....	34
Fig 17. Importació d'un full de càlcul delimitat per comes en Quantum GIS.	

Font: Elaboració pròpia.....	35
Fig 18. Kml després de la correcció -93, -203 mitjançant QuantumGIS.	
Font: Elaboració pròpia.....	36
Fig. 19: Opcions de simbolització a Google Earth i opció de cerca d'imatge per a simbolitzar. Font: Elaboració pròpia.....	36
Fig 20. La cartografia kml, amb la possibilitat d'ésser guardat en kmz a Google Earth. Font: Elaboració pròpia.....	37
Fig 21: Imatge obtinguda en plena transferència d'arxiu kml al servidor. Font: Elaboració pròpia.....	38
Fig 22. Exemple de connexió al servidor mitjançant FileZilla. Font: Elaboració pròpia.....	39
Fig 23. Exemple de predeterminació de directori local a FileZilla. Font: Elaboració pròpia.....	40
Fig 24: Esquema FTP. Font: Elaboració pròpia.....	40
Fig 25: Imatge que ofereix el web de codi de Google per tal de generar l'API. Font: http://code.google.com/apis/maps/signup.html	43
Fig 26. Imatge inicial API d'Earth. Font. Elaboració pròpia.....	45
Fig 27: Càrrega de l'API i altres eines. Font: elaboració pròpia.....	47
Fig 28: Esquema de l'arbre. Font: Elaboració pròpia.....	56
Figura 29: El consorci va posar-se com a fita poder elaborar vinyetes semblants a les de google/maps Font: http://maps.google.es/	58
Fig 30: En aquest cas la vinyeta per defecte mostra l'identificador dels centres d'informació. Font: Elaboració pròpia.....	63
Fig 31: A l'esquerra, un exemple de com una marca de posició és fàcilment editable enganxant el codi de google.code.com. A la dreta, el resultat. Preparat per a ser degudament personalitzat. Font: Elaboració. Pròpia.....	64
Fig: 32 L'entrellat variable i invariable d'un kml. Font: Elaboració pròpia.....	65
Fig 33: Web corporatiu en estat de proves. Font: Elaboració pròpia.....	67
Fig 34. Apartat SIG del web corporatiu. Font: elaboració pròpia.....	68
Fig 35. Apartat SIG del web corporatiu: polígons. Font: elaboració pròpia.....	69
Fig 36. Apartat SIG del web corporatiu: punts. Font: elaboració pròpia.....	69
Fig 37. Apartat SIG del web corporatiu: vinyeta emergent. Font: elaboració pròpia.....	70
Fig 38. Símbol amb contorn transparent a l'apartat Maps. Font: elaboració pròpia.....	71

Fig 39. Apartat Guia de serveis, restaurants. Font: elaboració pròpia.....71

Fig 40. Apartat Guia de serveis, centres educatius. Font: elaboració pròpia.....72

1. Introducció

El document que tenim entre mans, més enllà de ser la memòria del projecte de final de Màster en Tecnologies de la Informació Geogràfica, exposa la meua primera experiència com a professional de les tecnologies de la informació geogràfica lluny del Departament de Geografia. Aquest últim factor ha estat determinant per tal d'entendre la complexitat del món laboral, però també per reflexionar sobre la constant evolució que la geoinformàtica està vivint avui en dia, i que m'ha obligat a adaptar-me a les necessitats tant d'aquest món laboral com dels nous llenguatges informàtics, els quals em brindaran una vida professional dinàmica i rica en nous coneixements, si la sort em permet el luxe de poder-m'hi dedicar en un context de crisi.

Sense deixar de banda les dificultats, la revalorització paradigmàtica de la geografia dins el paradigma quantitatiu posa sobre la taula, des de fa anys, noves maneres d'entendre la visió de la geografia en un món en que les tecnologies informàtiques ens han dut com a societat cap a una nova concepció de l'espai. Ja des dels anys vuitanta, multitud d'autors debaten sobre l'impacte de l'anomenada "automatització de la geografia", fet que no tan sols sembla canviar el món de la geografia i els geògrafs sinó la relació amb moltes altres disciplines que ara fan servir la geografia, o bé aquesta permet respondre preguntes fins ara inqüestionablement dins d'altres ciències consolidades. Des del meu punt de vista els geògrafs hem de tenir un paper important en aquesta nova interpretació del món del segle XXI, i en aquest sentit vaig considerar de màxima transcendència en la meua trajectòria formativa posar-me fil a l'agulla cursant el Màster en Tecnologies de la Informació Geogràfica.

Deixant com a més que òbvia la relació entre informàtica i geografia, presento sintetitzats, en unes vuitanta pàgines, els tres mesos de conveni de col·laboració al Consorci del Parc de Collserola i la meua experiència a les oficines tècniques. En un entorn únic, i a un pas de la gran ciutat, he pogut immernir-me en nous llenguatges i programes que han ampliat els meus coneixements sobre els crèdits teòrics del primer any de Màster.

Marc Institucional i de treball: El Consorci del Parc de Collserola

La meua primera aportació l'he dut a terme en un ens públic de caràcter associatiu entre l'Àrea Metropolitana de Barcelona, la Diputació i, des de fa ben poc, la

Generalitat de Catalunya, que amb la seva entrada dins al Consorci, permetrà a l'espai natural deixar de ser un simple espai protegit per passar a ser un Parc Natural en tota regla.

Davant d'aquest nou ordre ja imminent, es fa del tot necessari renovar el web del Parc de Collserola, i amb llur renovació poder utilitzar recursos de programació web-SIG per tal de resoldre problemes de divulgació i de gestió de la principal joia natural metropolitana. En primer lloc es considerarà oportuna la meua participació en tal renovació per a difondre els recursos físics i humans del parc a partir d'un visualitzador on-line d'informació geogràfica, i així poder informar millor als usuaris de "què, com i on" hi ha dins l'àrea protegida.

Una de les principals missions del consorci és la de complir els objectius bàsics del Pla Especial, conservar els recursos naturals, l'equilibri ecològic, o bé donar resposta a la demanda de lleure i l'aprenentatge dels ciutadans. En aquest sentit, perquè les oficines tècniques gestionin amb èxit, gràcies al visualitzador, sectors com ara les incidències dels guardes forestals, els inventaris de flora i fauna, dades sobre incendis, etc, el visor també gaudirà de tota la informació que faciliti a la institució el sistema d'informació geogràfica.

El treball, per tant, pren un sentit no tan sols divulgatiu, sinó que també preten d'ésser una eina d'ús transversal entre totes les institucions que conformen l'ens associatiu, i així, un punt de trobada metropolitana dins un web corporatiu.

1.2 Origen i antecedents del projecte

Abans de la meua arribada a les oficines del parc, el servei tècnic ja disposava d'una aplicació web creada amb Visual Web Developer dins del mateix programa que no era pública, i d'altra banda, gaudia d'un projecte de web corporatiu, i un web en funcionament més antic on la màxima font cartogràfica era una imatge en pdf que hom podia descarregar-se. Com és lògic, aquesta no permetia interactuar ni visualitzar cap tipus d'informació que no formés part ja de la pròpia imatge, i és així com la primera necessitat del Consorci fou la de migrar l'aplicació i adequar-la al disseny del web, i així aixecar un pilar més per a poder-lo publicar a mitjà termini.

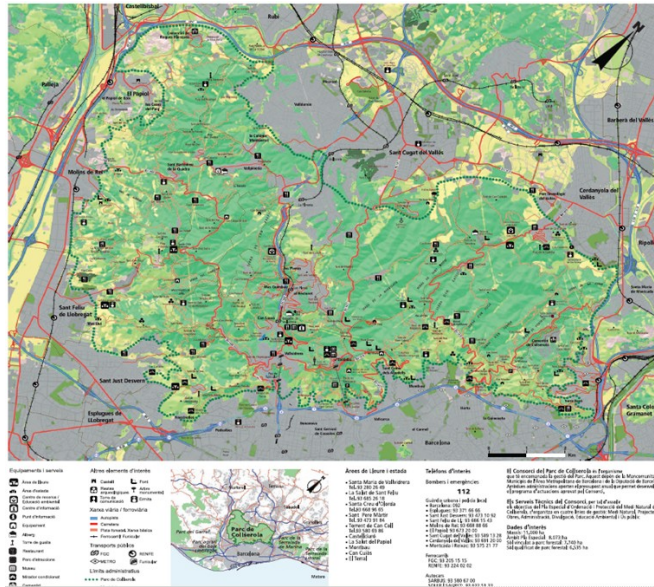


Fig 1. Cartografia en pdf de l'antic web encara per renovar. Font: [www.parcollserola.net/](http://www.parccollserola.net/)

La institució, que ha tingut col·laboradors del Departament de Geografia en la dotzena d'edicions que s'han dut a terme fins al moment, ja havia programat aplicacions web en les darreres tres col·laboracions. En la novena i desena edició aquestes foren desenvolupades a partir de MapServer, PHP i Progress, i en l'onzena s'avançà en els coneixements de l'API de Google Maps amb la programació del Web de consulta de l'arxiu fotogràfic. Tanmateix, aquestes no acabaren de reeixir per limitacions en la velocitat de resposta entre d'altres problemes, fet que motivà la petició de col·laboració 2010-11.

La familiarització en les API de Google pel desenvolupament de funcionalitats per part del consorci farà que conjuntament amb l'entorn de programació Javascript es canviï d'estratègia ateses les possibilitats d'aquestes interfícies, sobretot la d'Earth, i que s'investigui en aquest sentit mitjançant Visual Web Developer Express 2008.

2. Objectius

Per tal d'adequar-me bé a les exigències de divulgació i gestió marcades en el marc institucional i de treball, primer de tot havia de pensar quins eren els objectius generals que en un principi havia de tenir en compte per tal de satisfer les necessitats del Consorci i superar amb èxit l'experiència. D'aquesta manera, en un inici vaig considerar que m'enfrontava a cinc grans qüestions tècniques i objectius a resoldre:

1- Familiaritzar-me amb nous llenguatges i programes

Una de les primeres dificultats és la necessitat d'adequar-me a un nou llenguatge, ASP, i a un nou programa per a l'elaboració de pàgines web, Visual web Developer. En aquest sentit, com és lògic, el primer dels objectius és el d'aprofundir en altres formes de programació web-SIG, i així poder dur a terme un bon projecte adequat al codi d'ús de la feina començada pel mateix Consorci.

2- Dissenyar una nova estructura de menús i l'entorn web

És necessària l'elaboració d'una estructura de menús tipus arbre que permetin seleccionar i destriar àgilment la informació que es vol visualitzar. Tot i així el disseny en si de la resta de l'entorn web no pren més rellevància a petició del mateix consorci atès que el web corporatiu en si ja compta amb un especialista en la matèria.

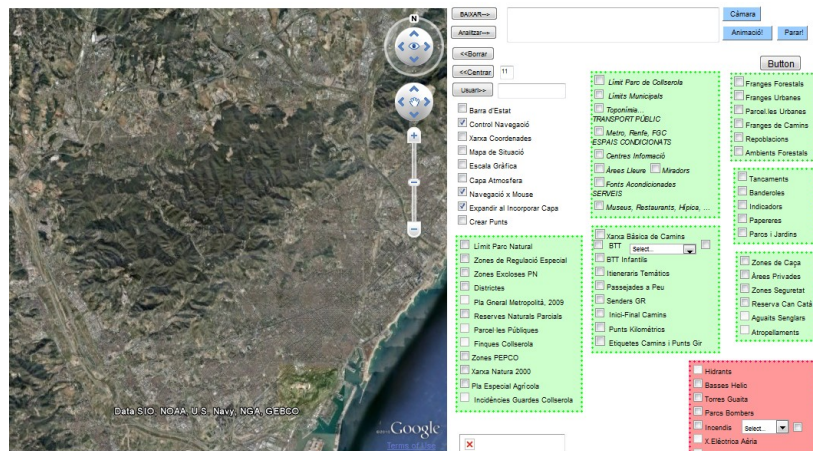


Fig2. Mostra de l'estat en que es trobava l'aplicació. S'havia començat a migrar la informació SIG però no de forma ordenada. Font: Elaboració pròpia.

3- Millorar l'aplicació accedint a dades

Establir procediments que permetin la vinculació entre els elements gràfics i dades alfanumèriques, imatges, enllaços, documents digitals, vídeos i la seva consulta mitjançant vinyetes emergents.

4- Establir diferents formes d'integració de l'aplicació al web corporatiu

Un cop enllestida la migració i la millora de l'aplicació, a petició del Consorci, també se'ns demana poder dissenyar altres aplicacions de caràcter sectorial per a complementar els diferents apartats de la guia de serveis. Així, per exemple, l'apartat del web on es descriu els restaurants hi ha d'haver un petit visualitzador que mostri en aquest cas la capa de restaurants, i el mateix en altres casos com els punts d'informació, centres educatius, etc.

3. Anàlisi de requeriments

Per tal d'acomplir els objectius cal analitzar els requeriments de l'aplicació i especificar-ne les funcionalitats, així com pensar en el programari a utilitzar, els llenguatges que en permeten la programació i els usuaris que faran ús del producte final. Per tal de fer-ho calia posar sobre la taula diferents mètodes, valorar èxits i fracassos d'altres projectes, especular sobre noves possibilitats i adaptar la programació en funció de la finalitat en què el servei tècnic vol dirigir els resultats que es van fent palesos a mesura que el sistema va esdevenint una realitat.

3.1 Requeriments funcionals

En un inici, els requeriments funcionals plantejats foren molts. Més endavant s'anaren prioritzant els requeriments de més pes, i també els que necessitaren més hores de dedicació.

A continuació exposo les funcionalitats que ja incorporava l'aplicació abans de l'inici de la col·laboració i les que es proposaren de tirar endavant de setembre a desembre. Finalment també incloc un apartat on especifico passos que vaig haver d'aprendre els primers dies per tal de dur a terme el projecte, encara que el consorci ja els hagués posat en pràctica abans del dia tretze.

1- Funcionalitats programades abans del 13 de setembre de 2010:

- Visualització de l'API d'Earth i la informació GIS en kml, kmz i altres formats que suportin l'API
- Reforç de la component intuïtiva de la web que permeti fer-la servir a un usuari no experimentat: controls, escala, navegació per mouse, zooms, extents, pan oferts per google, en els que la majoria de visitants poden estar familiaritzats.
- Incloure la barra d'estat, xarxa de coordenades, navegació per mouse i la possibilitat de crear punts entre els demés recursos de Google Earth.
- Possibilitat de filtrar la informació en funció dels usuaris que volen consultar-la, si són treballadors del Consorci o no.
- Incloure un convertidor de coordenades, de geogràfiques a UTM i una barra d'estat de càrrega.

2- Requeriments a programar a partir del 13 de setembre de 2010:

- Poder seleccionar la informació cartogràfica a consultar sobre l'Earth a partir d'un objecte tipus arbre amb checkboxes.
- Millorar la consulta interactiva a través d'objectes balloon o formularis que permetin vincular elements cartogràfics amb la càrrega d'imatges, enllaços i d'altra informació en les vinyetes emergents que respondran als clics a les capes kml i kmz.
- Pàgines sectorials replicant aquest model a demanda del gestor de la pàgina web.

3- Familiarització i aprenentatge de procediments i funcions ja implementades:

- Elaboració, transformació i transmissió d'informació cartogràfica a Google (kml).
- Visualització de l'API d'Earth i la informació GIS en kml, kmz i altres formats que suportin l'API
- Reforç de la component intuïtiva de la web que permeti fer-la servir a un usuari no experimentat: controls, escala, navegació per ratolí, zooms, extents, pan oferts per google, amb els que la majoria de visitants poden estar familiaritzats.

Per qüestions de temps i per poder exercir el màxim esforç i dedicació a complir tots aquests requeriments, quedaren exclosos de la meva adaptació a l'aplicació: el filtrador d'usuaris, el convertidor de coordenades i la barra d'estat de càrrega.

3.2 Requeriments de software i llenguatges

Els requisits tecnològics i de software representen tots aquells programes i llenguatges sobre els quals ens basarem per tal de dur a terme el projecte. En aquest sentit es feu palesa la necessitat d'adaptació a Visual Web Developer i sobretot a nous llenguatges i estils com ara ASP, que combinat amb Javascript conformarien el conjunt de codi que ens permetria donar a llum a l'aplicació.

A continuació faig una breu descripció dels softwares, llenguatges i servidors en els que suportàrem el projecte:

1-Requeriments de software

Visual Web Developer Express Edition 2008: Aquest software representa la versió express o gratuïta de Visual Studio, de Microsoft, està format per un conjunt d'eines i utilitats per a la creació de llocs web i les seves aplicacions, i representa avui en dia un potent entorn de programació dissenyat per ajudar –nos a crear aplicacions, facilitant el disseny i reduint la quantitat de codi necessari per a crear-les.

ArcGIS 10: Sota aquest nom hi ha el software més potent que avui dia pot capturar, editar, analitzar, dissenyar i imprimir informació geogràfica. A partir d'una de les seves eines transformarem la informació creada pel mateix ArcGIS al format adient per a poder-la carregar des del servidor a la nostra aplicació (kml). El consorci va haver de pagar llicència per gaudir d'aquest excel·lent programa.

ArcViewGIS 3.3: També software d'ESRI, l'última versió d'aquest SIG sortí al mercat l'any 2002, i ara mateix representa l'avantpassat més proper d'ArcGIS. El Consorci en manté l'ús per qüestions d'hàbits i per la seva senzillesa alhora d'editar informació geogràfica, encara que no pot transformar shapes a format kml.

Altres GIS de software lliure: QuantumGIS, Mapwindow GIS, GSP TraackMaker, seran altres programes, aquest cop gratuïts, que permetran la transformació de shape al format Google, i que em prendré el seu ús, en tant que programari lliure, com a quelcom personal i ètic. L'edició SIG tan sols serà possible en els dos primers casos.

Google Earth: Semblant a un SIG, aquest ens permet visualitzar la terra en tres dimensions i editar-hi informació cartogràfica. Ens serà molt útil per tal de verificar la informació kml acabada de transformar i fer-ne proves sense haver de penjar al servidor la cartografia cada vegada que hi duem a terme un canvi, o bé en programem les vinyetes emergents. També ens servirà per comprimir kml a kmz.

2- Llenguatges

ASP: Representa un framework per aplicacions web desenvolupat i comercialitzat per Microsoft. Permet la creació de webs dinàmiques, i en el mateix se suportarà gran part del codi emprat pel projecte com per exemple la part del disseny de l'aplicació que vindrà donada en gran part per Visual Web Developer.

Javascript: és un llenguatge d'scripts basat en objectes utilitzat per accedir -hi en aplicacions. Principalment s'utilitza integrat en HTML permetent el desenvolupament

d'interfícies d'usuari i pàgines web dinàmiques. Javascript ens permetrà complementar ASP i formular-hi funcions.

Kml-Xml: entendre el llenguatge en el que està composta la cartografia en format Keyhole Markup Language ens permetrà que aquesta, basada en xml (Extensible Markup Language), pugui accedir a dades i modificar l'aspecte de les vinyetes emergents de la cartografia. Kml entrà a formar part el 2008 dels estàndards Open Geospatial Consortium (OGC), amb la finalitat d'aconseguir l'estatus d'estàndard obert com a format d'intercanvi d'informació geogràfica.

3- Servidor

Perquè la nostra aplicació sigui visible per a tots els usuaris necessitem una plataforma on col·locar-la. D'aquesta manera el servidor públic del consorci permetrà que des del web, i per tant des de fora el parc, es puguin visualitzar les capes kml i kmz, penjar-hi la mateixa web, i fins in tot accedir a arxius determinats des de les vinyetes emergents. Per últim, l'FTP (veure apartat 5.3) FileZilla serà l'encarregat de traspasar la informació del nostre ordinador al servidor.

3.3 Solució metodològica i criteris de selecció

L'aplicació en la que ja havien treballat els serveis tècnics del parc havia estat programada fent ús dels llenguatges esmentats en l'apartat anterior i sobre Visual Web Developer.

En aquest sentit es va arribar a la conclusió de que sota aquests llenguatges i programari duríem a terme l'aplicació sobre els serveis de cartografia de Google, Earth i Maps. Així, es tingueren en comte la generalització d'aquests serveis de cartografia, que essent innovadors i emergents, resultaven ésser també força intuïtius tant pel personal del Consorci com per l'usuari extern.

Un any enrere, durant l'última col·laboració amb el Departament de Geografia, ja hi havia hagut un debat al respecte, que havia decantat el Consorci per fer ús de Visual Web Developer (en endavant VWD) i Google Maps en front de MapServer. D'una banda, MapServer no tenia tantes limitacions alhora de llegir dades i podia representar capes més complexes però Google Maps i VWD per la seva banda, tot i tenir

l'inconvenient de la limitació d'arxius kml segons la seva grandària, tenia els avantatges següents:

- Kits d'inici integrats plenament per arrancar d'immediat
- Depurador integrat per aplicacions Web.
- Un sol arxiu unificat en totes les parts del projecte
- Eines ja predefinides (zoom, escales..)
- Servidor web integral per fer proves a nivell local.
- Accés a informació alfanumèrica i a arxius fàcilment des de kml

Aquest fet també suposaria haver-me de familiaritzar amb un nou llenguatge, l'ASP, i el mateix VWD, però també poder assumir a curt termini la sèrie d'avantatges respectius.

El debat però de la col·laboració d'enguany aniria entorn a quina API fer servir, si la de Google Maps o la de Google Earth sota Visual Web Developer. I és que en gran part Google Earth suposaria gaudir dels citats avantatges, i a més a més suportaria kml i kmz d'una grandària molt superior a Maps, com ara per exemple: l'inventari de camins del parc sencer, sota un detall també força considerable i en tres dimensions. Alhora però tenia l'únic inconvenient que l'usuari havia d'instal·lar el plug-in per poder visualitzar l'API d'Earth, i això podria propiciar la desorientació d'usuaris no especialitzats o sense temps/motivació momentània.

La conclusió final doncs fou la de fer programar les aplicacions sota VWD, però la web corporativa tindria un apartat SIG en Google Earth per a usuaris més familiaritzats que instal·larien el plug-in i visualitzarien tota la informació SIG unificada, i d'altra banda, pels apartats sectorials de restaurants, centres d'informació, etc, Google Maps permetria una visualització ràpida de la localització dels elements sense que l'usuari hagués d'instal·lar cap plug-in i en forma de marc complementari, i no com a absolut atractiu únic i central del subapartat-web.

Arribats a aquest punt, dins una única solució metodològica, sota VWD, hem representat els apartats SIG i Guia de Serveis en tant que el primer seria programat sobre Earth i els segons sobre Maps.

3.4 Usuaris

El visor web es dirigeix a dos tipus d'usuaris finals, els visitants del web corporatiu convencionals i d'altra banda els treballadors del consorci. Els primers tindran accés a la informació de caràcter general com ara els límits dels municipis, els mitjans de transport, el perímetre del parc o bé les fonts. Els treballadors del servei tècnic, en canvi, a més a més de tota aquesta informació general tindran accés, a través de contrasenya, també a la informació com ara els incendis, les incidències, les batudes, les basses etc. L'accés mitjançant una contrasenya, tal i com s'especifica als requeriments, ja hauria estat programada pel Consorci abans del tretze de setembre.

4. Planificació inicial i Arquitectura

Donada la magnitud del projecte calia gestionar bé els tres mesos de col·laboració, marcant-me objectius plausibles però possibles, que sense deixar de ser ambiciosos fossin prudents, tenint en compte la limitació de quatre hores de dedicació diària. Alhora, calia plantejar en forma d'arquitectura el paper dels requeriments de software, així com llurs llenguatges respectius, per tal d'ordenar de forma esquemàtica el procés que seguiria la informació, un cop planificada, d'inici a final.

4.1 Planificació inicial dels treballs

El quadre de la pàgina següent exposa el que creguí que representaria la meua col·laboració al llarg dels mesos. Com s'observa, la millora de l'aplicació és l'etapa de més pes, i que s'estén més de mes i mig. La familiarització és relativament ràpida, mentre que la redacció es va repartint a mesura que s'avança en la resta de tasques específiques.

El fet de tenir els objectius força clars des d'un inici va fer que el calendari inicial no es mogués excessivament a la pràctica. L'aproximació amb el programari es va ajustar a la exposada, i la primera part d'octubre la varem dedicar a la programació de l'arbre i a l'ordenació de la informació, després d'acabar d'editar la informació cartogràfica. L'apartat "d'accés a dades" no va anar dirigit a crear una connexió a base de dades, sinó a l'estudi de l'entrellat dels kml i la programació de les seves variables. L'objectiu de no deixar mai de pensar en la memòria final es veuria reflectit en una posterior còmode i dilatada redacció, ja que aniria plantejant punt per punt des de la primera setmana com hauria d'anar estructurada i quina importància haurien de tenir les fases i les tasques específiques, més tard redactats en forma d'apartats i subapartats respectivament.

	FASE	TASCA ESPECÍFICA	Setembre		Octubre				Novembre				Desembre			
			III	IV	I	II	III	IV	I	II	III	IV	I	II		
		Anàlisi de requeriments i planificació inicial	■													
		Intal·lació del programari	■													
Fase 1	Familiarització	Familiarització amb els nous programes i llenguatges		■	■											
		Elaboració de cartografia		■	■											
Fase 2	Càrrega de cartografia i ordenació de l'aplicació	Creació d'un arbrat per a la càrrega de capes		■	■											
		Programació de l'accés a dades				■	■									
		Creació de vinjetes									■	■				
Fase 3	Millora de l'aplicació	Càrrega informació a les vinjetes									■	■				
		Introducció d'altres eines ja existents i complements														
Fase 4	Adaptació al web corporatiu	Creació d'aplicacions sectorials														■
Fase 5	Redacció del	-														■

Fig. 3. Quadre de planificació. Font. Elaboració pròpia.

4.2 L'Arquitectura

L'esquema que tenim a continuació ens exposa la proposta client - servidor, de l'arquitectura del projecte dins la que s'integra l'aplicació web.

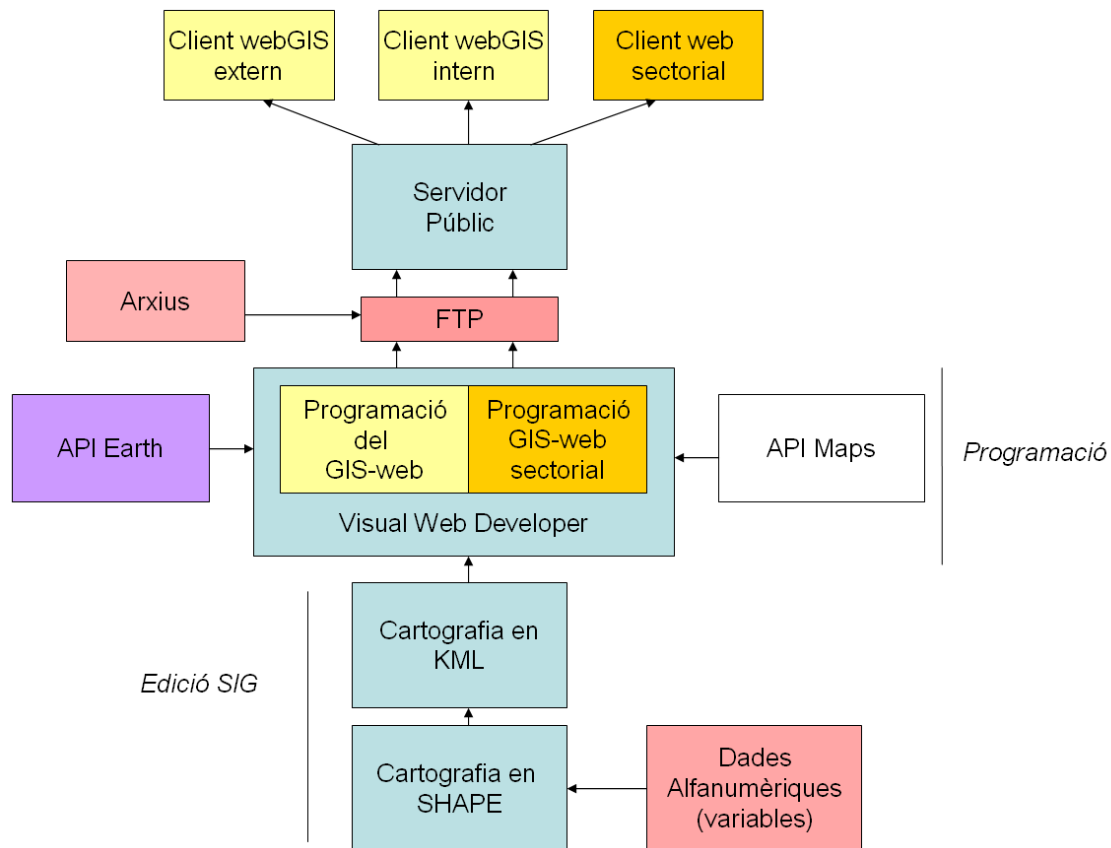


Fig 4. Proposta d'arquitectura. Font: Elaboració pròpia.

Bàsicament, l'esquema defineix de forma molt resumida i en forma de croquis el desenvolupament del projecte dels apartats que vénen a continuació. A grans trets tot comença amb l'edició de la cartografia, sota l'estàndard shape, que és transformada al format Google per tal d'ésser carregada a Visual Web Developer, des d'on es programaran les aplicacions i s'integraran les interfícies de Maps o Earth. A partir d'aquí, mitjançant un FTP, tota la informació és traslladada al servidor públic i els diferents usuaris faran ús del web des de la xarxa. La diferència d'aquesta i moltes altres arquitectures serà que la informació alfanumèrica vindrà incorporada des d'un inici amb la informació cartogràfica. Menys els arxius, que resten independents de la cartografia dins el servidor, la resta de variables alfanumèriques poden ésser programades dins el l'esmentat format kml.

5. Tractament i estructuració de les dades

Abans de començar a programar sobre les API de Google fou important pensar en com elaborar dades perquè aquestes fossin després representades sobre la base cartogràfica en la qual s'haurien de recolzar des del visor. La gran majoria de dades representen informació espacial que va acompanyada d'atributs alfanumèrics. De fet, encara que moltes ja foren elaborades en anterioritat a la meua arribada, calia també transformar-les a format Google i transferir-les al servidor públic per a poder ser carregades sobre les interfícies.

5.1 Edició i elaboració d'informació cartogràfica

Per tal de tirar endavant l'elaboració d'informació cartogràfica, el Consorci utilitzà ArcView GIS 3.3. Aquest, com a avantpassat proper d'ArcGIS, permet al servei tècnic seguir editant informació cartogràfica mitjançant el mateix mètode en que els treballadors ja estan familiaritzats des de principis de la dècada, a més a més d'oferir una menor complexitat i una major rapidesa a l'hora de dur a terme les funcionalitats bàsiques que necessita la nostra poc complexa cartografia.

La meua intervenció en aquest sentit ha estat més aviat de poca importància, però cal destacar la reedició de la simbolització de Restaurants, Espais de Lleure, Edificis PEPCO, límit del Parc Natural..., entre d'altres. En total però la informació cartogràfica a afegir i ordenar a l'apartat SIG del web corporatiu mitjançant la programació Visual Web Developer i sobre l'API d'Earth, ha estat la següent:

- Límit del Parc Natural
- Límits Municipals
- Zones de regulació especial
- Toponímia
- Zones excloses al PN
- Transport públic
- Districtes
- Centres d'Informació
- Pla General Metropolità, 2009
- Àrees de Lleure
- Reserves Naturals Parcials
- Miradors
- Parcel·les Públiques
- Fonts Condicionades
- Finques Collserola
- Restaurants
- Zones PEPCO
- Museus
- Xarxa Natura 2000
- Hípica
- Pla Especial Agrícola
- Xarxa Bàsica de Camins
- Incidències dels Guardes Forestals
- Itineraris BTT
- Itineraris BTT Infantils

- Itineraris temàtics
- Passejades a Peu
- Senders GR
- Inici - Final Camins
- Punts quilomètrics
- Etiquetes camins i punts Gir
- Franges forestals
- Franges Urbanes
- Parcel·les urbanes
- Franges de Camins
- Repoblacions
- Ambients forestals
- Tancaments
- Banderoles
- Indicadors
- Papereres
- Parcs i Jardins
- Zones de caça
- Àrees privades
- Zones de seguretat
- Reserva de Can Catà
- Aguaites Senglars
- Atropellaments
- Hidrants
- Basses Helicòpters
- Torres Guaita
- Parc de Bombers
- Incendis
- Xarxa elèctrica

L'apartat de la guia de serveis del web corporatiu consta de sis parts que han d'incorporar cartografia sobre l'API de Maps en els diferents marcs sectorials de cada apartat, també programats sobre Visual Web Developer:

- Punts d'Informació i perímetre del parc
- Restaurants
- Allotjaments
- Museus i equipaments
- Lleure i esport

El format sota el qual desem les dades cartogràfiques durant la seva edició és el format d'ESRI shapefile, un format d'arxiu informàtic que s'ha convertit actualment en estàndard de facto per a l'intercanvi d'informació geogràfica. “.SHP” és l'extensió de l'arxiu, que emmagatzema les entitats geomètriques. Aquest fet contrasta amb que l'API d'Earth i Maps no accepten aquest format i aquest ha d'ésser convertit a un format especial per tal d'ésser-hi representat.

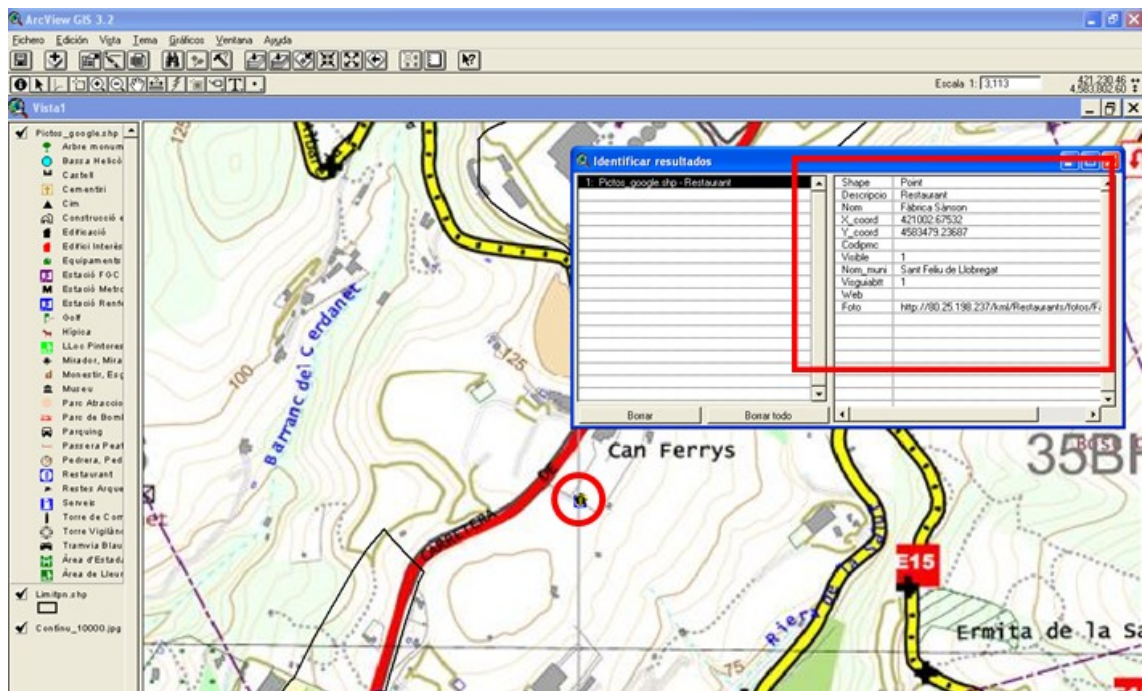


Fig 5. Els atributs del restaurant de la Fàbrica Sànsion a Sant Feliu de Llobregat en format shapefile a ArcView. Font: Elaboració pròpia.

5.2 Transformació i adaptació d'informació geogràfica a Google

Per a què les interfícies de Google puguin reconèixer la cartografia i carregar-la hem de transformar-la a l'anomenat format "kml", Keyhole Markup Language. Per a fer-ho, hem de recórrer a programari que llegeixi el format shapfile i tingui també la capacitat de transformar-lo a kml, retornant-ne el resultat final de la mateixa cartografia en un aquest últim format.

5.2.1 Transformació a kml mitjançant ArcMap 10

ArcMap 10 d' ArcGIS 10 també forma part del programari per tal d'elaborar dades cartogràfiques però pren menys rellevància al llarg de la col·laboració i sobretot representa un suport per a convertir els SHP a l'esmentat format de kml a partir d'una DLL. DLL, literalment de l'anglès "llibreria d'enllaç dinàmic", és el que ens permet en un sol clic retornar qualsevol cartografia que pugui obrir el mateix ArcMap en format Google (kml), salvar-ne llurs atributs i determinar-ne el sistema de coordenades i el 'diversió' o desviament. La DLL pren el nom de shp2kml i és de caràcter gratuït dins ArcGIS, un software no lliure.

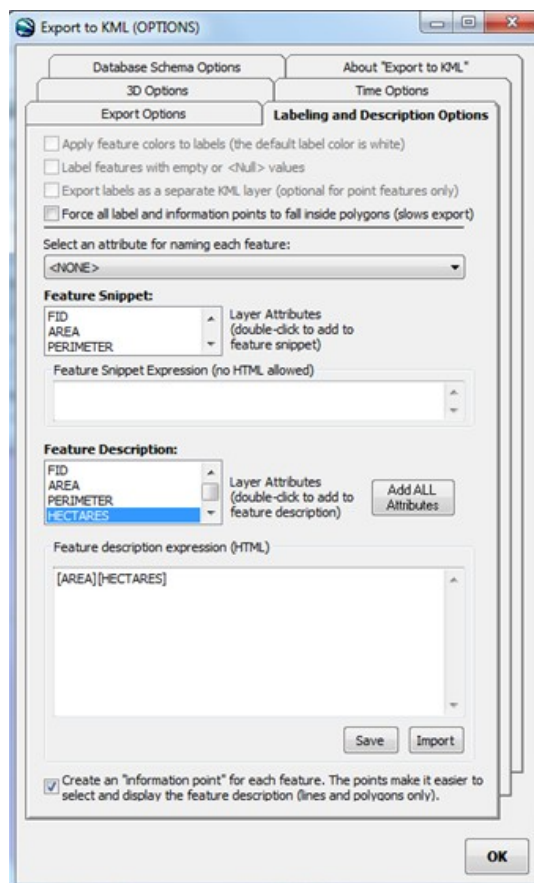


Fig 6. La transformació i llurs múltiples opcions. Font: Elaboració pròpia

ArcMap 10 representa el software més còmode a l'hora de transformar informació cartogràfica al format Google, i el que ens permet també determinar-ne més atributs, etiquetes i d'altres. Tanmateix, té l'inconvenient d'ésser software de pagament i per tant, lluny de les oficines del parc, el meu marge de maniobra a l'hora de fer ús de l'eina seria nul, sense abonar a l'empresa ESRI una quantitat important per tal d'adquirir el producte.

5.2.2 Transformació a kml mitjançant programari lliure

Per tal de poder dur a terme altres projectes sense haver de pagar llicències mitjançant les API de Google cal trobar altres programes de GIS que ens permetin dur a terme la conversió. Aquesta però és més una qüestió personal i moral, més que una necessitat en si. En aquest sentit he trobat a la xarxa molts programes, la majoria vinculats al món dels GPS, com ara GPS TrackMaker, que permeten convertir shapes a kml, això sí, sense salvar-ne els atributs, atès que no són un GIS.

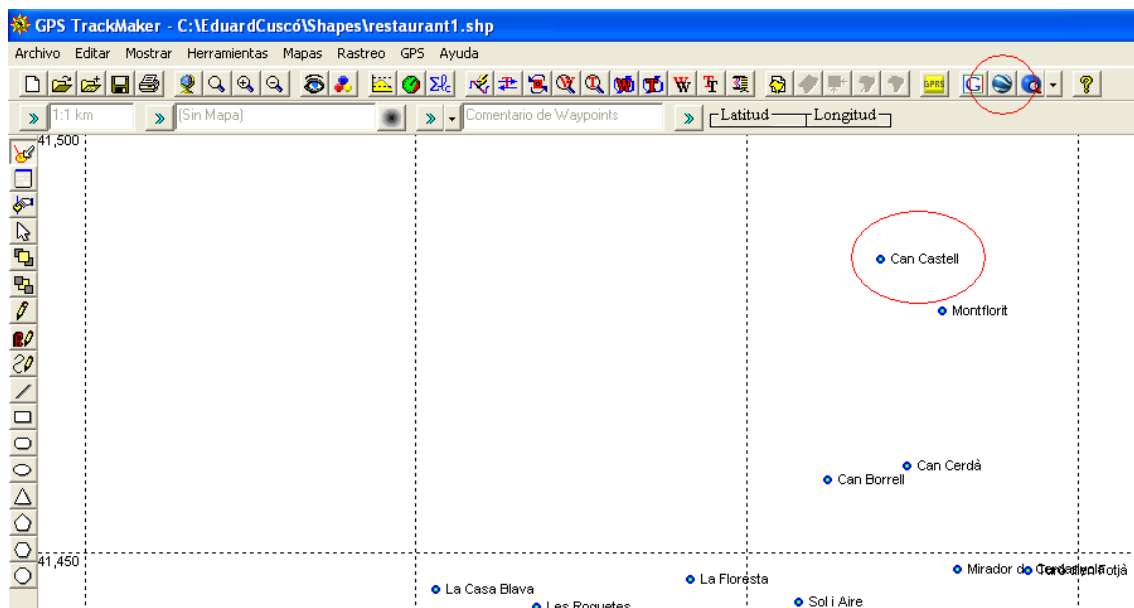


Fig 7. Visualització del shape “restaurants” dins de GPS TrackMaker, amb la DLL incorporada.

Font: Elaboració pròpia

Ens cal doncs un programa GIS que no tan sols elabori informació cartogràfica shape sinó que salvi la informació alfanumèrica del shape per tal de tenir accés a les seves dades a posteriori. En aquest sentit Quantum GIS ens permet elaborar informació cartogràfica de manera àgil i didàctica, i no obstant, convertir informació SHP a KML podent-ne escollir el nom o el sistema de coordenades en que es preten transformar la informació.

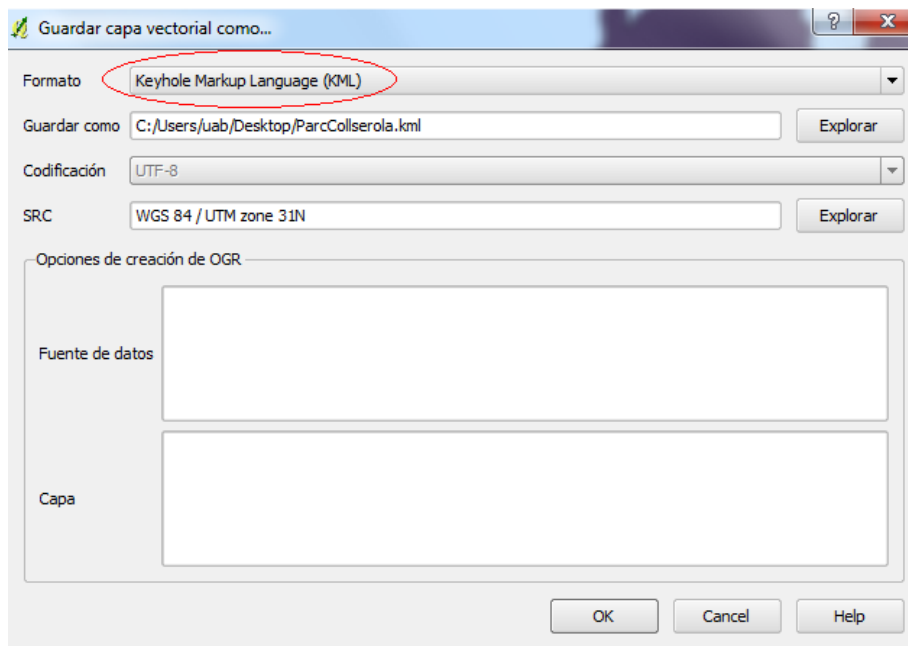


Fig 8. L'opció Keyhole Markup Language a l'hora de guardar la capa vectorial i el sistema de coordenades. Font: Elaboraci3n pr3pia.

Per3 tal conversi3 sol s guarda el primer atribut del shape, sense opcions a condicionar la columna de la informaci3 que cal mantenir, i a m3s a m3s ho fa en forma d'etiqueta, i no com a variable dins el codi kml. Aquesta opci3 no tan sols limita el posterior 3s de la informaci3 alfanum3rica per a l'elaboraci3 de posteriors vinyetes emergents, sin3 que, en cas d'3sser un shape amb gran quantitat de diferents entitats espacials, cal reeditar-ne les etiquetes a Earth una per una, aix3 com el s3mbol; que 3s per defecte el "macador" convencional de color groc d'Earth.

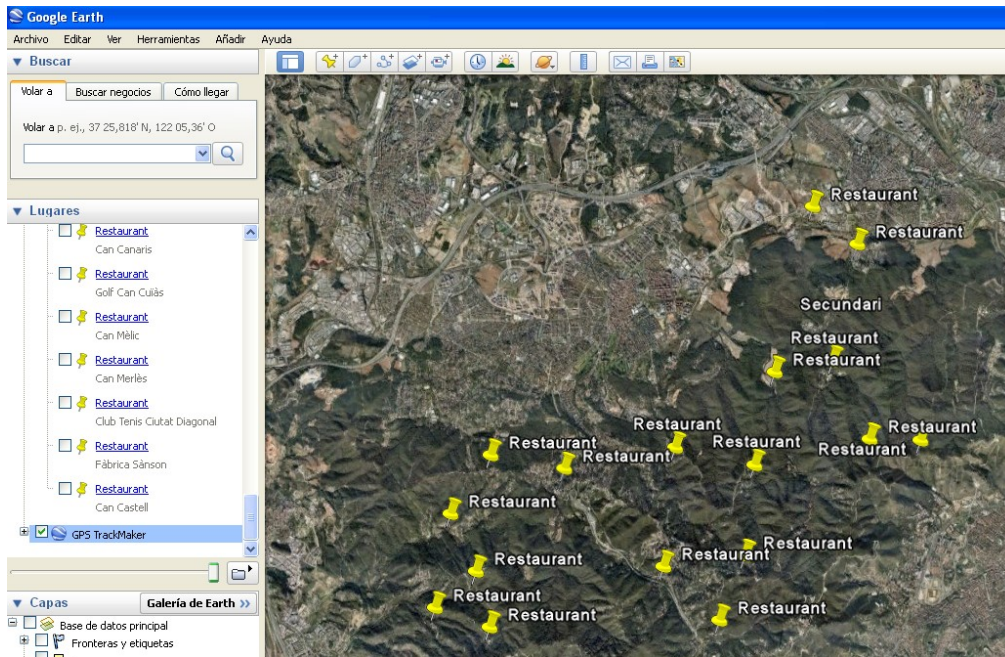


Fig 9. La capa restaurants ara en format kml, amb el marker i l'etiqueta per defecte.

Font: Elaboració pròpia.

D'aquesta manera ja estem en condicions de visualitzar la nostra informació mitjançant Google Earth com a mode de prova abans de carregar la capa a l'aplicació web. Però la informació alfanumèrica del shape ha d'ésser salvada, i en aquest sentit un altre GIS, MapWindow GIS, admet l'opció d'instal·lar una DLL anomenada "shp2earth", clarament inspirada en la "shp2kml" d'ESRI, permet l'opció de guardar-ne els atributs que desitgem com a variables, l'etiqueta, i fins i tot determinar-ne la vinyeta emergent.

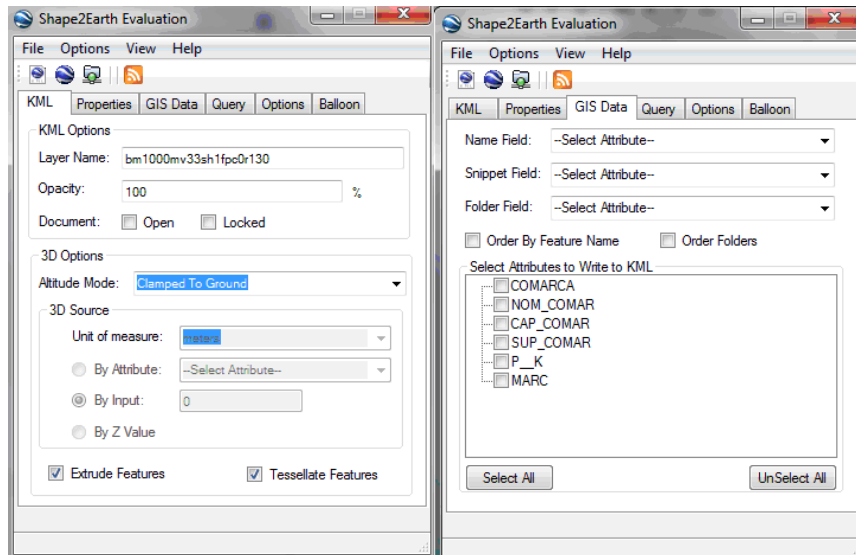


Fig. 10 Les opcions de transformació de la DLL de Mapwindow, com s'observa es poden seleccionar els atributs escollits per a salvar. Font: Elaboració pròpia.

Shape	Descripcio	Nom	X_coord	Y_coord	Codpmc	Visible	Nom_muni	Visguabil	Web	Foto
Point	Restaurant	Torreblanca	431576.47851	4589489.19631		1	Barcelona	1		http://80.25.198.237/kml/Res
Point	Restaurant	Mirador de Cerdanyola	427952.27353	4589105.62444		1	Cerdanyola del Valles	1		http://80.25.198.237/kml/Res
Point	Restaurant	Turó d'en Fojà	428541.53786	4589077.57044		1	Cerdanyola del Valles	1		http://80.25.198.237/kml/Res
Point	Restaurant	Can Cerdà	427026.65796	4590522.26285		1	Cerdanyola del Valles	1		http://80.25.198.237/kml/Res
Point	Restaurant	Montfloit	427430.83089	4592451.07873		1	Cerdanyola del Valles	1		http://80.25.198.237/kml/Res
Point	Restaurant	Sol i Aire	425594.44170	4588618.44482		1	Sant Cugat del Valles	1		http://80.25.198.237/kml/Res
Point	Restaurant	Can Borrell	426020.65415	4590273.83246		1	Sant Cugat del Valles	1		http://80.25.198.237/kml/Res
Point	Restaurant	Centre Informació CPC	424952.20028	4586032.46014		1	Barcelona	1		http://80.25.198.237/kml/Res
Point	Restaurant	Les Planes	423992.73594	4588879.97647		1	Barcelona	1		http://80.25.198.237/kml/Res
Point	Restaurant	La Casa Blava	421001.63395	4588842.31580		1	Sant Cugat del Valles	1		http://80.25.198.237/kml/Res
Point	Restaurant	La Rierada	420264.52062	4587832.28182		1	Molins de Rei	1		http://80.25.198.237/kml/Res
Point	Restaurant	Can Tintorer	420006.06213	4586204.43036		1	Molins de Rei	1		http://80.25.198.237/kml/Res
Point	Restaurant	Can Portell	420973.54552	4585866.51723		1	Molins de Rei	1		http://80.25.198.237/kml/Res
Point	Restaurant	Can Carbonell	423692.58250	4583797.22415		1	Sant Just Desvern	1		http://80.25.198.237/kml/Res
Point	Restaurant	Can Cortès	425413.64917	4587062.08385		1	Sant Cugat del Valles	1		http://80.25.198.237/kml/Res
Point	Restaurant	La Floresta	424217.58780	4588937.72330		1	Sant Cugat del Valles	1		http://80.25.198.237/kml/Res
Point	Restaurant	Les Roquetes	422281.04786	4588597.97945		1	Sant Cugat del Valles	1		http://80.25.198.237/kml/Res
Point	Restaurant	Can Campmany	420745.40566	4586831.31144		1	Molins de Rei	1		http://80.25.198.237/kml/Res
Point	Restaurant	Can Canaris	419923.22554	4584276.43769		1	Sant Feliu de Llobregat	1		http://80.25.198.237/kml/Res
Point	Restaurant	Golf Can Cuiàs	420296.94378	4583264.00102		1	Sant Feliu de Llobregat	1		http://80.25.198.237/kml/Res
Point	Restaurant	Can Mèlic	422481.49673	4582519.96199		1	Sant Just Desvern	1		http://80.25.198.237/kml/Res
Point	Restaurant	Can Merlès	423735.15153	4584126.96040		1	Sant Just Desvern	1		http://80.25.198.237/kml/Res
Point	Restaurant	Club Tennis Ciutat Diagonal	424522.33193	4583230.20730		1	Sant Just Desvern	1		http://80.25.198.237/kml/Res
Point	Restaurant	Fàbrica Sansón	421002.67532	4583479.23687		1	Sant Feliu de Llobregat	1		http://80.25.198.237/kml/Res
Point	Restaurant	Can Castell	426660.25228	4593136.47044		1	Cerdanyola del Valles	1		http://80.25.198.237/kml/Res

Fig 11. Un exemple de la quantitat d'atributs que es poden emmagatzemar i que prèviament han estat elaborats mitjançant Arcview 3. Font: Elaboració pròpia.

A la troballa sobre MapWindow GIS hi tenim però dos grans inconvenients. En primer lloc, el creador de shp2earth va limita la DLL a un màxim de 500 transformacions sense necessitat de registrar-se prèviament, i per tal de fer-ho, hom ha d'obtenir un codi pagant al creador 29,99 dollars americans. Per tant, l'aparent "software lliure" passa a poder-se considerar software de pagament.



Fig 12. L'advertiment de shape2earth. Font: Elaboració pròpia.

En segon lloc, l'inconvenient és el de no poder transformar la cartografia amb l'opció de condicionar-ne el desviament. En aquest sentit, fins la finalització de la col·laboració malauradament no trobem cap altre GIS que permeti transformar shapes a kml determinant-ne l'offset o compensació.

5.2.3 WGS84 el sistema de coordenades de Google

La cartografia georeferenciada i elaborada amb el sistema de coordenades Universal Transversa Mercator (UTM) o el mateix ED55 de l'Institut Cartogràfic de Catalunya causa més d'un problema si es té en comte que les coordenades que accepta Google, i per tant els kml, són les WGS84, un sistema de coordenades geogràfiques. En primer lloc, si s'intenta transformar la cartografia canviant el sistema a WGS84 el kml s'esguerra i per tant cal transformar-lo a partir de les UTM, i en segon lloc si així es fa, el desviament que es produeix és d'uns centenars de metres al nord-est.

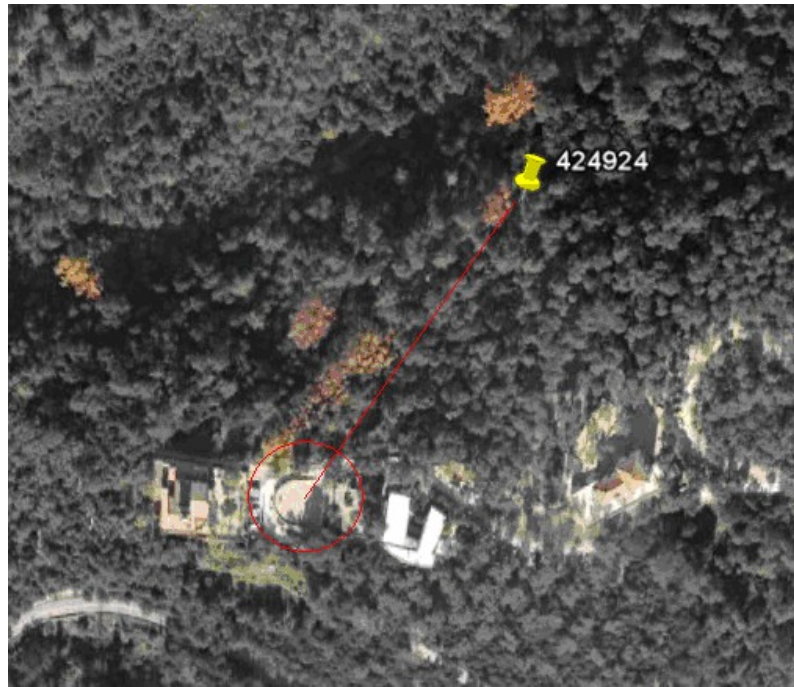


Fig 13. Exemple de desviament resultant de la georeferenciació del consorci.

Com s'observa és d'uns centenars de metres. Font: Elaboració pròpia.

Com s'apuntava anteriorment, el convertidor d'informació cartogràfica d'ArcMap 10 inclou l'opció de determinar l'offset o compensació sobre el desviament o 'diversion' que se'n deriva de transformar d'un sistema de coordenades a l'altre.

L'operació matemàtica que ArcMap 10 duu a terme és la de restar -93 a la coordenada UTM X i -203 a la coordenada Y, respectivament, i així corregir el desviament causat per la conversió.

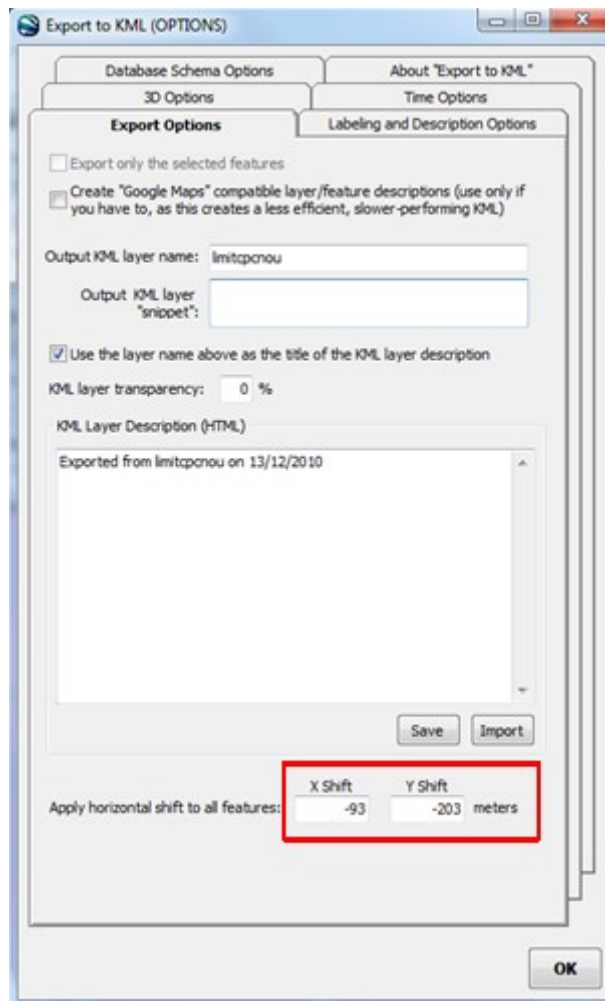


Fig 14. Opció DLL d'ArcMap amb opció a condicionar l'offset. Font: Elaboració pròpia.

El principal problema doncs pel que fa a la conversió mitjançant el software lliure és el de no poder determinar el desviament, atès que en aquest cas no podem transformar informació ni visualitzar-la correctament a una escala gran i detallada, en la qual es descobreix de seguida l'esviaix.

Per últim, podem concloure que en cas de que es vulgui fer ús de software lliure per a l'obtenció de cartografia kml podem fer ús de les següents opcions:

- 1- Elaborar informació mitjançant el programa de Google Earth: Com s'observa a la imatge, Google permet elaborar polígons, punts i línies directament i correcta sense desviament, atès que es treballa sobre les coordenades geogràfiques WGS84.

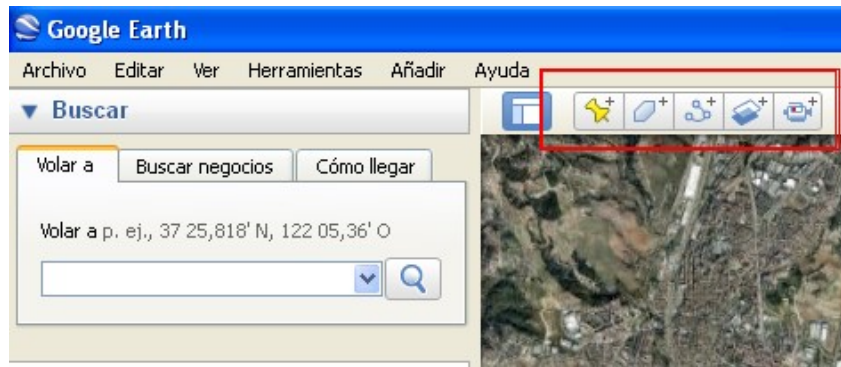


Fig 15. Les opcions d'edició d'Earth. Font: Elaboració pròpia

- 2- Corregir el desviament de forma manual mitjançant un full de càlcul: Això sí, en el cas de que siguin punts que es puguin georeferenciar a partir de les dades del shape i disposem de les respectives coordenades UTM.

	A	B	C	D
1	X Original	Y Original	X - 93	Y - 203
2	338384	4664841	338291	4664638
3	400716	4671832	400623	4671629
4	360949	4670761	360856	4670558
5	429843	4583245	429750	4583042
6	360949	4670761	360856	4670558
7	338384	4664841	338291	4664638
8	338384	4664841	338291	4664638

Fig 16. Columnes resultants de la resta en excel.

Font: Elaboració pròpia.

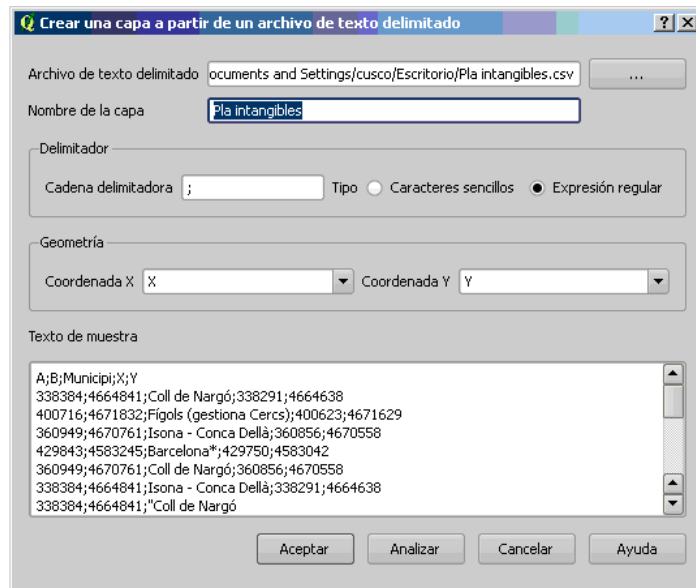


Fig 17. Importació d'un full de càlcul delimitat per comes en

Quantum GIS. Font: Elaboració pròpia.



Fig 18. Kml després de la correcció -93, -203 mitjançant

QuantumGIS. Font: Elaboració pròpia.

5.2.4 Edició de cartografia mitjançant Google Earth, i els kmz

El programa de Google Earth ens serveix, tal i com s'ha mostrat en l'últim apartat, per a verificar en cada cas si la conversió de shape a kml ha estat idònia i també poder visualitzar alguns aspectes abans de carregar la cartografia al servidor. Així mateix, també permet l'edició d'altres opcions com ara la simbolització, les etiquetes, etc, podent fer ús, tant d'estàndards de simbolització del mateix Google Earth, com d'altres imatges.

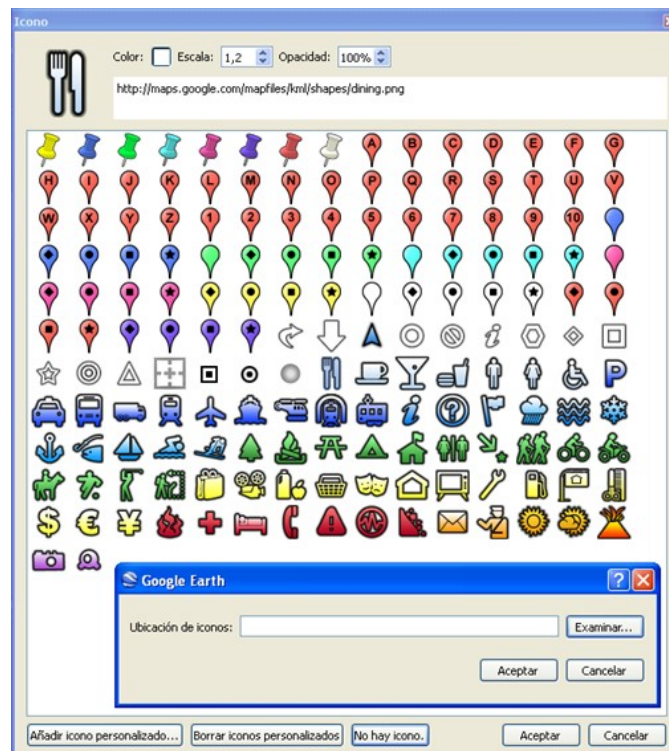


Fig. 19: Opcions de simbolització a Google Earth i opció de cerca

d'imatge per a simbolitzar. Font: Elaboració pròpia.

D'altra banda, earth ens representa la principal eina per a comprimir kml i transformar-los als anomenats kmz, i fer així que la cartografia sigui més lleugera, o bé pugui carregar-se amb condicions davant de possibles problemes amb les API. Les seves sigles fan referència també a Keyhole Markup però la L de la paraula "Language" en aquest cas és substituïda per la zeta de "Zip". Aquest, en tant que fitxer comprimit, permet la incorporació de molta més informació, per a fer-ho, Earth ens brinda l'opció de guardar en kmz.

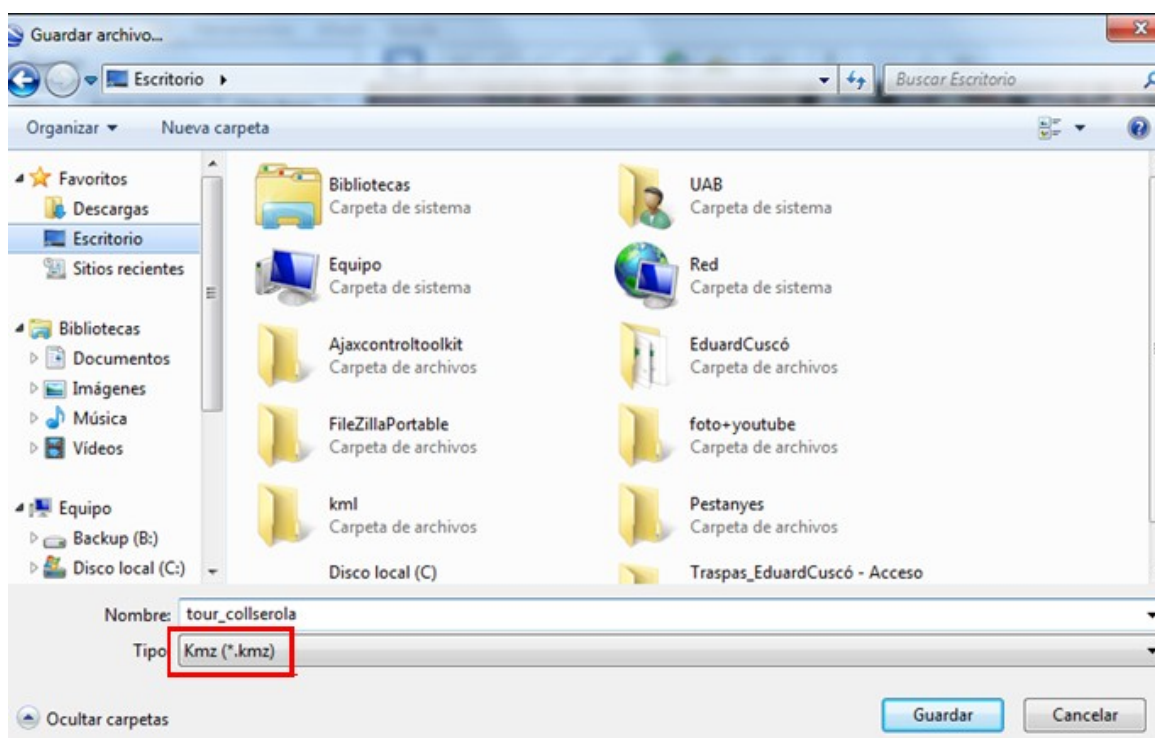


Fig 20. La cartografia kml, amb la possibilitat d'èsser guardat en kmz a Google Earth.

Font: Elaboració pròpia

5.3 Protocols de transferència d'arxius al servidor públic d'informació

Per tal de que la informació cartogràfica i el sistema d'arxius puguin ser objecte de treball durant la programació de la pàgina web que ha de publicar-los, hem de transferir tota la informació a un servidor públic a partir del qual hom pot accedir a tal informació mitjançant una adreça que fa referència al servidor en qüestió. Així, es presenten els anomenats protocols de transferència d'arxius com una bona eina per tal de traslladar, des de qualsevol ordinador personal del servei tècnic, tots els arxius de tot tipus, cartogràfics i no cartogràfics, al servidor sense haver-hi d'accedir de manera "física".

5.3.1 Client FileZilla

D'aquesta manera el Consorci del Parc de Collserola ja gaudia des de feia anys del protocol de transferència d'arxius "FileZilla", un client multiplataforma de software lliure, en anglès anomenat "File Transfer Protocol" i abreviat com a protocol de xarxa, FTP. Més enllà d'aquesta mera descripció, l'FTP FileZilla s'entén molt bé podent descriure la següent imatge:

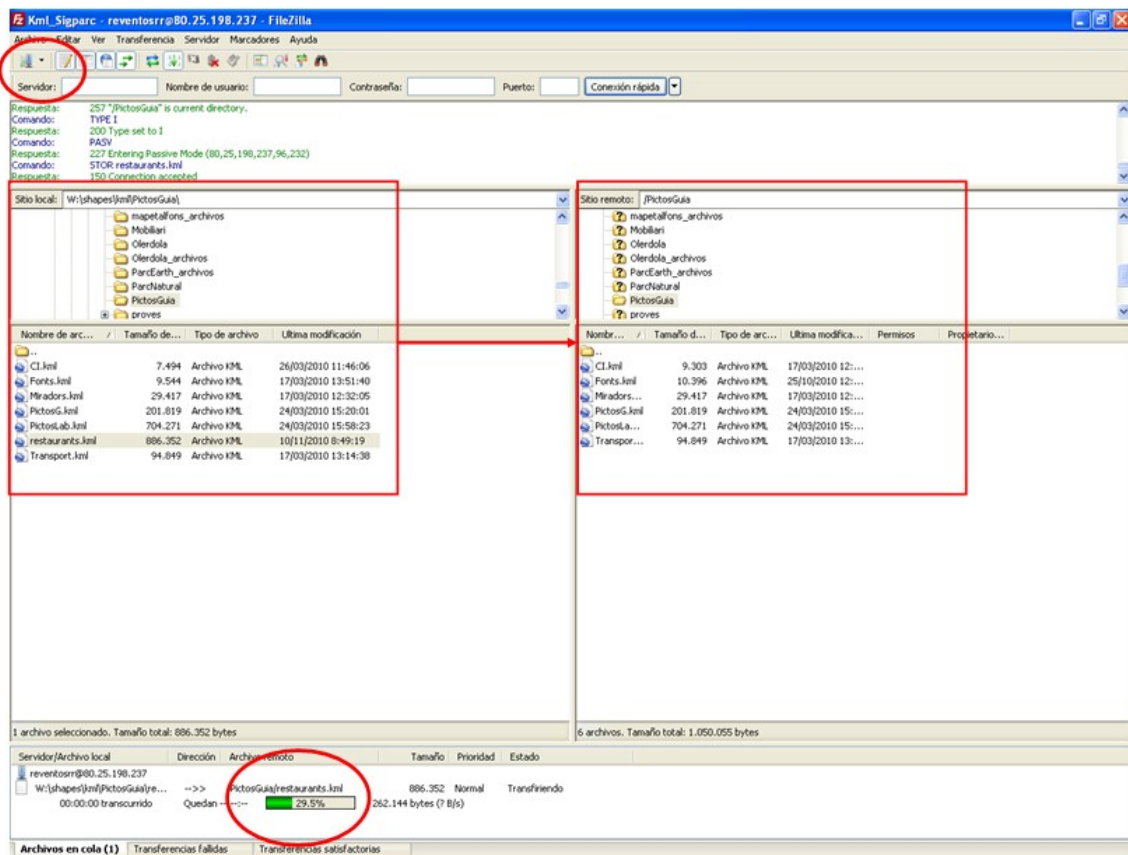


Fig 21: Imatge obtinguda en plena transferència d'arxiu kml al servidor. Font: Elaboració pròpia.

Com s'observa, el quadrat vermell de la dreta representa totes aquelles dades que han estat transferides al servidor públic, i a l'esquerra totes aquelles dades a partir de les quals s'han duplicat arxius al servidor o bé les que estan a disposició, dins el disc local, d'ésser transferides mitjançant FileZilla. La imatge, de fet, és tant oportuna que immortalitza el moment en que un kml, de "restaurants", del disc local W és traslladat dins el servidor públic a una carpeta del mateix nom per tal d'ésser duplicat i publicat. Allà s'especifica nom de l'arxiu, direcció, percentatge transferit (en verd), la mida, prioritat i finalment, l'estat en que es troba. Com és lògic a la dreta apareixen tots els arxius traspasats però no el kml de restaurants que ho està fent en el precís instant.

Per fer-ho però, cal una connexió al servidor mitjançant un usuari tal i com exposa la següent imatge:

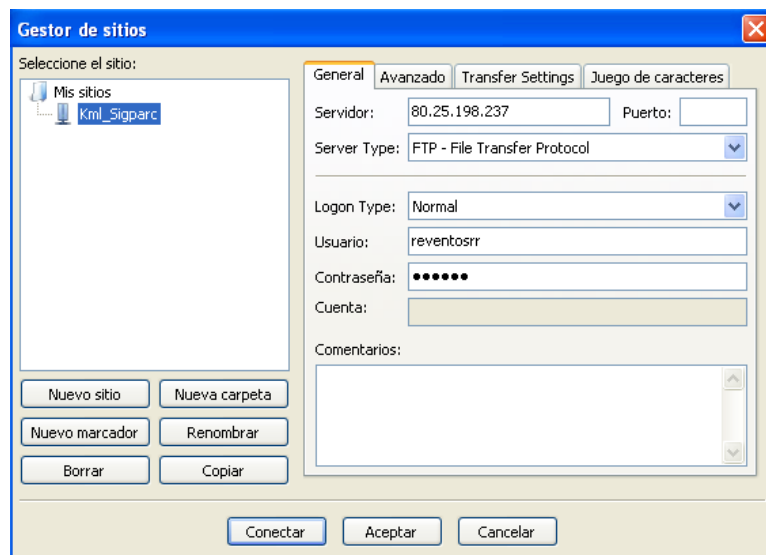


Fig 22. Exemple de connexió al servidor mitjançant FileZilla.

Font: Elaboració pròpia.

Després de crear un "lloc" (sitio), "kml_Sigparc", especifiquem el número del servidor, en el cas de Collserola "80.25.198.237", l'usuari, el del meu tutor **reventosrr**, i finalment la contrasenya i el directori local predeterminat "**W:\shapes\kml**" on FileZilla accedeixi per defecte alhora de transferir dades.

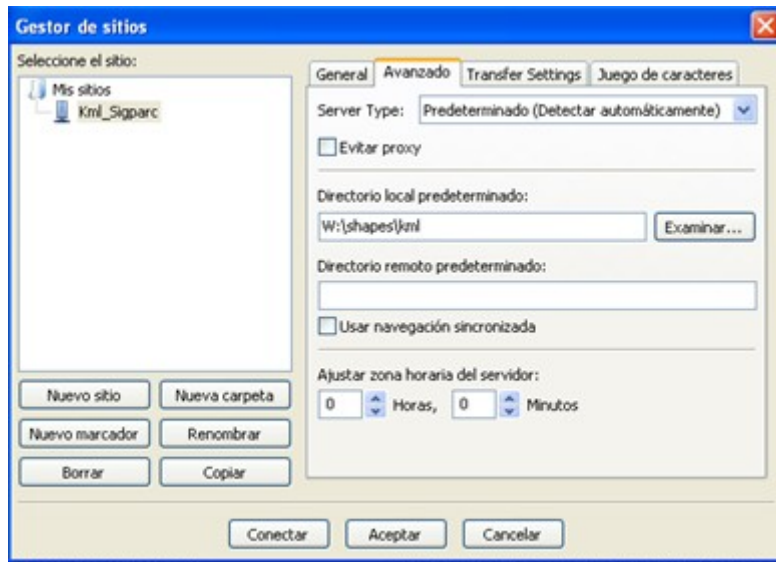


Fig 23. Exemple de predeterminació de directori local a FileZilla. Font: Elaboració pròpia

Tot plegat queda resumit en el següent esquema, de baix a dalt, el procés que segueixen els arxius per tal d'arribar del sistema d'arxius local de treball fins a qualsevol ordinador amb accés a la xarxa pública.

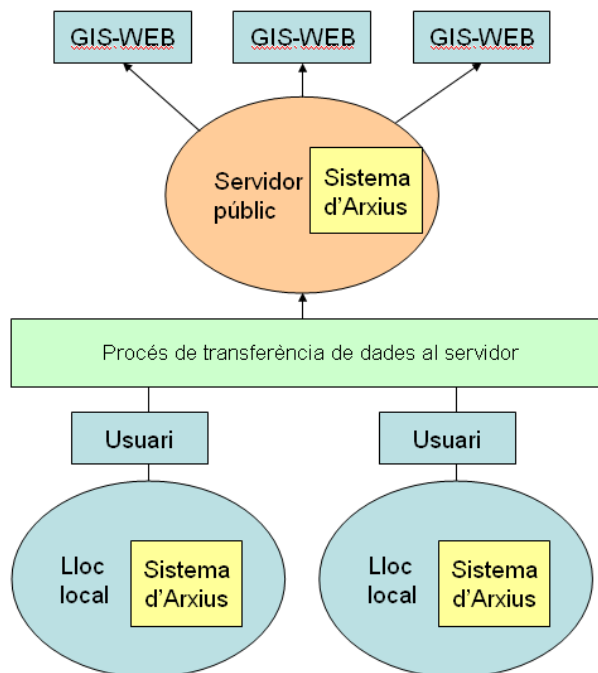


Fig 24: Esquema FTP. Font: Elaboració pròpia.

El sistema d'arxius, com es fa palès a l'esquema, rau dins un ordinador i un lloc local. A partir de l'usuari i els respectius passos esmentats anteriorment s'accedeix a FileZilla i s'engega el procés de transferència de dades al servidor, que un cop allà poden ésser carregats, sempre que la direcció sigui la correcta des de qualsevol codi de programació que vulgui fer ús de la cartografia i arxius del servidor. A continuació, l'exemple de crida d'un kml des del servidor, com s'observa del número del servidor s'arriba al kml passant per les carpetes corresponents en forma d'enllaç.

```
url = 'http://80.25.198.237/kml/Franques/FranquesForestals.kml'
```

6. Implementació de l'aplicació, migració i integració de la informació GIS

Fins ara, ja tenim la informació cartogràfica elaborada i transformada en els formats pertinents i ha arribat el moment de que l'aplicació carregui, de forma estructurada, totes les dades que el Consorci considera oportunes, sobre les interfícies oportunes. A més a més, tenim també la informació cartogràfica a un servidor on-line que ens la retorna a partir del seu correcte enllaç. En definitiva, deixant enrere el món de l'edició en programes de Sistemes d'Informació Geogràfica, cal posar-se a programar i a immersir-se dins de nous llenguatges, la part més difícil i alhora interessant del projecte, per tal d'elaborar una aplicació que representi, mitjançant un web dades cartogràfiques, i ens permeti fer consultes també alfanumèriques.

6.1 Aproximació tecnològica

Per a fer-ho, tenim a l'abast centenars d'exemples on-line que fan referència a possibles maneres de començar a programar sobre una sèrie de codi ja elaborat de forma prèvia. Hem doncs, de descobrir la manera de carregar interfícies primer, de migrar-hi la informació cartogràfica després, i finalment, programar-hi de forma estructurada les seves opcions de representació i informació de vinyetes emergents. A continuació exposo doncs, detall a detall, els passos seguits des de la càrrega d'una interfície fins a la programació de vinyetes amb informació complementària a la cartografia.

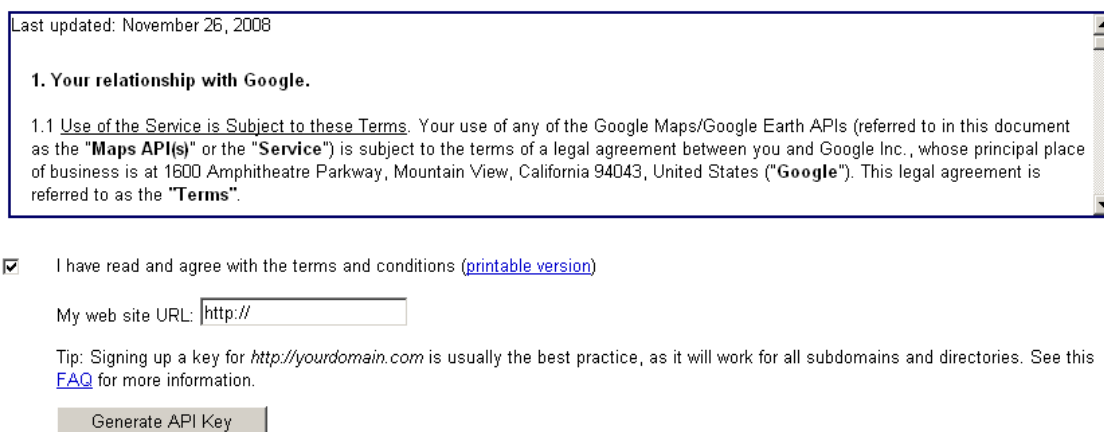
6.1.1 Interfícies de programació d'aplicacions Google

Una Interfície de Programació d'Aplicacions, de l'anglès *Application Programming Interface* i abreviat API, és, segons viquipèdia, un conjunt de declaracions que defineixen el contracte d'un component informàtic amb qui farà ús dels seus serveis.

Segons tal definició, l'API és un instrument que s'usa per a construir un sistema informàtic o llibreria de programació, així com per a donar suport a les invocacions a serveis fetes per un altre programa. L'API doncs, s'ofereix tant als programes externs, que podran usar els serveis oferts, com al programador, que ha de disposar de la informació indispensable per poder treure el màxim suc del component que ha adquirit.

En el cas de les API de Google, aquestes són unes biblioteques de Javascript que permeten la visualització de dades geogràfiques i cartogràfiques sobre les quals es pot representar informació cartogràfica dins una web. Les API que Google ofereix

actualment són dues, la de Google Earth i la de Google Maps, i tant l'una com l'altra cal sol·licitar-les a la web <http://code.google.com/apis/signup.html>, on s'accepten les condicions de la casa Google i es facilita l'adreça de la pàgina web interessada en carregar una de les dues API. Tal i com s'especifica a la següent imatge, un cop dut a terme aquest últim pas ja estem a disposició de generar la clau de l'API.



Last updated: November 26, 2008

1. Your relationship with Google.

1.1 Use of the Service is Subject to these Terms. Your use of any of the Google Maps/Google Earth APIs (referred to in this document as the "Maps API(s)" or the "Service") is subject to the terms of a legal agreement between you and Google Inc., whose principal place of business is at 1600 Amphitheatre Parkway, Mountain View, California 94043, United States ("Google"). This legal agreement is referred to as the "Terms".

I have read and agree with the terms and conditions ([printable version](#))

My web site URL:

Tip: Signing up a key for <http://yourdomain.com> is usually the best practice, as it will work for all subdomains and directories. See this [FAQ](#) for more information.

Fig 25: Imatge que ofereix el web de codi de Google per tal de generar l'API.

Font: <http://code.google.com/apis/maps/signup.html>

Un cop generada i rebuda la clau, aquesta ja pot ser inserida dins el codi de programació de Visual Web Developer, amb un aspecte similar al següent:

```
<script src="http://www.google.com/jsapi?key=ABQIAAAAwkbZLyhmTCWxbTcjbqBRsZs7K5SvaUdm8ua-Xxy_-2dYwMxQMnagaavTo7LiFE1-amhuQxI1Xw"></script>
```

Normalment l'espai de l'script on col·locar la clau ja ve donat pels exemples de Google. En molts casos aquests, ja vénen codificats amb una programació mínima i una clau de mostra que cal substituir per la que t'ha facilitat Google, fet que no complica obtenir-ne una, sinó haver de treballar i per tant personalitzar l'aplicació.

6.1.1.1 La interfície Earth

La filosofia de la primera de les API, la d'Earth, és poder oferir els avantatges del programa Google Earth dins un visor web. L'espectacularitat d'aquesta interfície la fa atractiva, entre altres coses, per les seves possibilitats en tres dimensions i la facilitat de càrrega de qualsevol cartografia en format .kml o .kmz.

L'anomenat "vol" d'Earth permet ja des del moment en que s'activa l'API, i si així es desitja, volar sobre imatges satèl·lit des del mateix univers fins al zoom-extent desitjat. Per defecte, el vol també s'adapta dins l'espectacular món en tres dimensions, a l'activació de qualsevol cartografia kml/z amb una rapidesa i realisme extrems.

Per a començar-nos a familiaritzar amb la interfície podem accedir a la pàgina d'exemples de Google, on les funcions per tal d'activar el món en tres dimensions sobre el que treballarem són ofertes des de bon principi:

```
google.load("earth", "1");

var ge = null;

function init() {
  google.earth.createInstance("map3d", initCallback, failureCallback);
}

function initCallback(object) {
  ge = object;
  ge.getWindow().setVisibility(true);
}

function failureCallback(object) {
}
```

Per si no n'hi hagués prou Google també ens facilita el "div" on rau el frame o marc on surt representada la cartografia en ASP:

```
</script>
</head>
<body onload='init()' id='body'>
  <center>
    <div>
      Hello, Earth!
    </div>

    <div id='map3d' style='border: 1px solid silver; height: 600px; width: 800px;'></div>
  </center>
</body>
</html>
```

Com observem, carrega la funció "init" i el respectiu mapa en tres dimensions dins el div que l'engrandim o enxiquim en funció dels nostres interessos. El marc resultant de l'anterior programació dins Visual Web Developer és la següent imatge:



Fig 26. Imatge inicial API d'Earth. Font. Elaboració pròpia.

També són interessants les facilitats que la casa Google posa per obtenir eines per tal d'aconseguir una millor navegació, escala gràfica, xarxa de coordenades, i fins i tot una capa atmosfèrica per a guarnir l'aplicació.

<input type="checkbox"/>	Barra d'Estat
<input checked="" type="checkbox"/>	Control Navegació
<input type="checkbox"/>	Xarxa Coordenades
<input type="checkbox"/>	Mapa de Situació
<input type="checkbox"/>	Escala Gràfica
<input type="checkbox"/>	Capa Atmosfera
<input checked="" type="checkbox"/>	Navegació x Mouse
<input checked="" type="checkbox"/>	Expandir Capa

A tall d'exemple, el control de navegació per ratolí s'afegeix a partir de la funció següent:

```
function updateOptions() {
    var options = ge.getOptions();
    var form = document.options;

    options.setMouseNavigationEnabled(form.mouseNav.checked);
}
```

I així successivament, fins a afegir la sèrie d'instruments i possibilitats del requadre anterior mitjançant el disseny amb Visual Web Developer i llenguatge ASP següent:

```

<form name="options" action='javascript:updateGeOptions();'

    style="border-color: #0000FF; border-style: dotted; z-index: 1; left: 827px; top: 22px; position: absolute; height: 160px;
    title="Controls Google">
<input type="checkbox" onclick='updateOptions()' name="statusbar" />Barra
d&#39;Estat
<br />

```

D'aquesta manera, afegim la funció al disseny del requadre anterior i del checkbox respectivament. A partir d'aquí tan sols ens queda centrar el mapa a l'extent per defecte que ens interessa emprant la següent funció:

```

function center_map() {

    var lookAt = ge.createLookAt('');
    lookAt.set(41.43, 2.1, 19000, ge.ALTITUDE_RELATIVE_TO_GROUND, 0, 0, 0);
    ge.getView().setAbstractView(lookAt);
    map.setCenter(new GLatLng(41.43, 2.1), 11);
}

```

Que la vinculem al següent codi ASP de disseny del botó:

```

<input id="Button_Centra"
type="button" value="<<Centrar Mapa"
onclick="center_map()"
style="font-family: Arial; font-size: xx-small; left: 731px; position: absolute; height: 20px; width:
title="Centrar Mapa a Collserola" />

```

Finalment, ja estem a disposició de començar a pensar en el disseny de l'arbre que classificarà la informació cartogràfica a carregar a l'aplicació, i que representarà l'apartat SIG de la nova web corporativa.

Fins aquí l'aplicació ofereix l'aspecte de la següent pàgina.

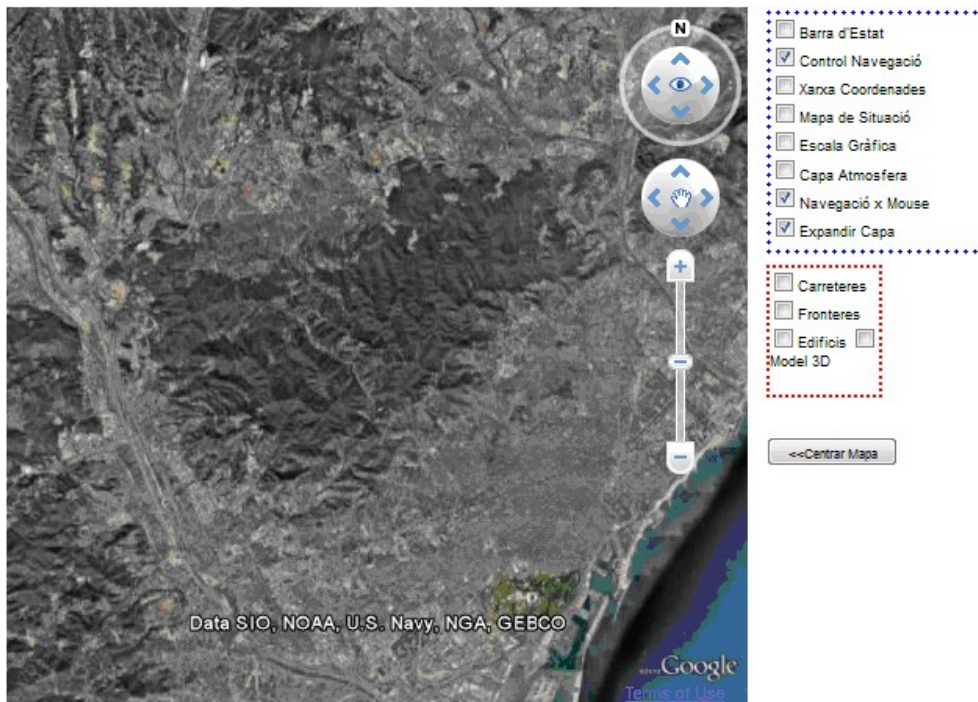


Fig 27: Càrrega de l'API i eines. Font: elaboració pròpia.

Com veiem, tenim l'API en marxa, els controls, la funció de centrar el mapa sobre Collserola i també la possibilitat que oferia Google d'afegir capes de carreteres bàsiques, fronteres o els edificis en tres dimensions, talment com ho fa amb l'escala, les coordenades o els controls.

6.1.1.2 Interfície Maps

L'API de Google Maps, en canvi, es basa en el web de la mateixa casa Google i probablement en el que representa avui en dia el buscador (i buscador geogràfic) de referència i més sol·licitat a nivell mundial, <http://maps.google.es/maps>. La virtut de la cartografia Maps és que a diferència d'Earth deixa de banda la tridimensionalitat per adoptar un format pràctic, de colors neutres i intuïtius que han aconseguit una base cartogràfica senzilla, homogènia i estàndard per tot el món. A més a més de calcular rutes en automòbil i a peu dins la gran majoria de vies catalogades, també ofereix poder alternar l'opció de la base cartogràfica amb imatges de satèl·lit, o bé combinar-les totes dues.

Com amb l'Earth, Google ens ofereix amb Maps marcs amb exemples pràctics de càrrega de controls, botons d'alternança de fons cartogràfic o imatge, escala i fins i tot exemples de funcions per a la càrrega de kml.

La funció següent facilitada des de la pàgina web d'exemples permet que en qüestió de poques línies de codi obtinguem un marc en condicions per a navegar-hi, determinar-ne l'escala inicial i les opcions de visualització.

```
var map;
var geoXml;
var toggleState = 1;

function initialize() {
  if (GBrowserIsCompatible()) {

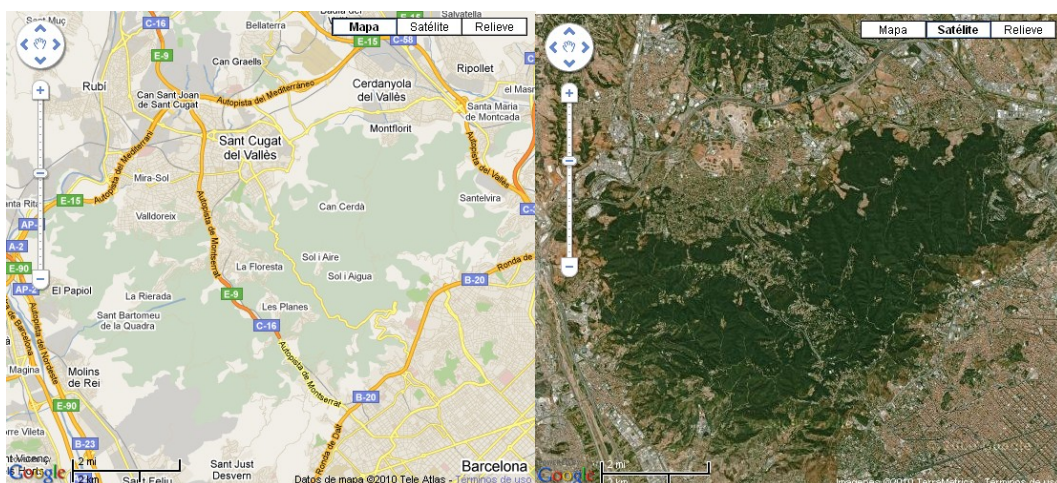
    map = new GMap2(document.getElementById("map_canvas"));
    map.setCenter(new GLatLng(41.875696,-87.624207), 11);
    map.setUIToDefault();
    map.addOverlay(geoXml);
  }
}
```

Els exemples de Maps també incorporen el disseny ASP del div i indiquen on es carreguen les funcions per defecte.

```
<body onload="initialize()">
  <div id="map_canvas" style="width: 640px; height: 480px; float:left; border: 1px solid black;"></div>
</div>
<br clear="all"/>
<br/>

</body>
```

Així doncs, també estarem a disposició de començar a carregar informació cartogràfica amb Maps, recordem, en apartats sectorials del web i sense necessitat de programar-hi cap arbre per a classificar-hi la informació. En aquest moment la nostra aplicació pilot prèvia a l'esmentada càrrega té un aspecte com el dels dos exemples següents:



6.1.2 Entorn, disseny i tecnologia Visual Web Developer

Tal i com ja havíem comprovat anteriorment, un dels avantatges de Visual Web Developer és el de crear automàticament el disseny de l'aplicació, així com de l'arbre des del qual classificarem les dades i donarem l'opció de seleccionar-les segons grup temàtic, tenint la possibilitat de precindir de mostrar totes les capes kml de cop, a més a més de poder-nos recolzar sobre botons com hem vist en aquest últim apartat.

6.1.2.1 El Disseny ASP

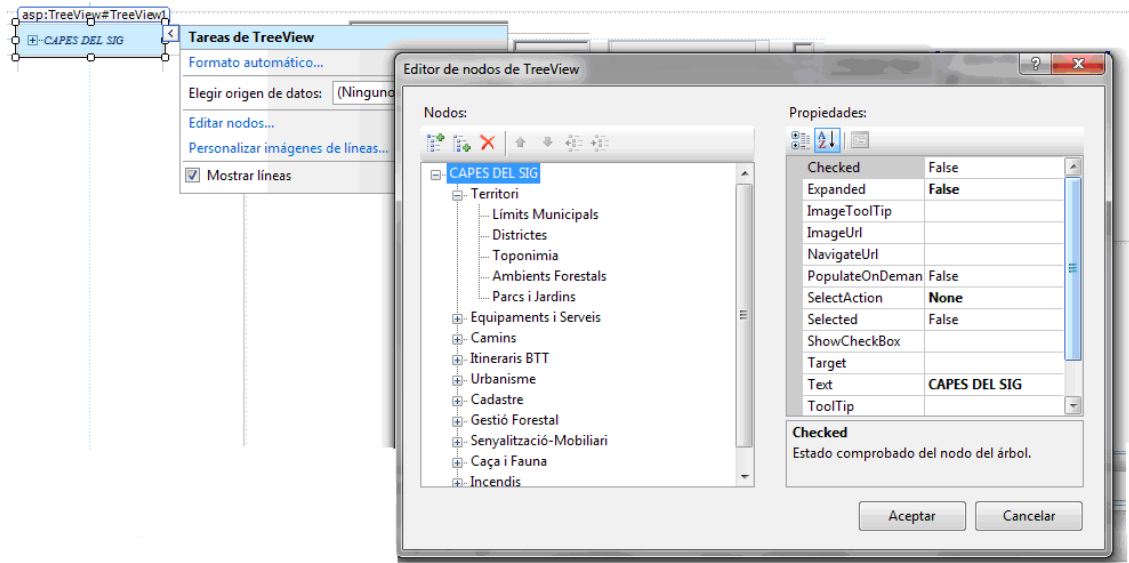
Tal i com succeeix amb el botó de centrar el mapa, a l'apartat de disseny tenim qualsevol element de control sobre la finestra on és representada la cartografia, incloent-hi la mateixa. Així, es crea paral·lelament un codi ASP en l'apartat de "codi" en el que també hi ha representades les funcions en llenguatge JavaScript. De fet, ASP dissenya l'aplicació però també resta al servei de la funció JavaScript que permet programar les funcions.

6.1.2.2 El Disseny d'un l'arbre

És d'aquesta manera com plantegem l'apartat SIG d'Earth al web corporatiu, en programar un arbre mitjançant el disseny ASP. La idea principal, com s'observa en la següent imatge, és la d'agrupar, ordenar i en definitiva estructurar els checkboxes de l'esquerra en un arbre on es pugui destriar la informació a visualitzar com el de la dreta, també amb checkboxes:



Per tal de començar a programar, dins la caixa d'eines de Visual Web Developer trobem l'opció **treenode**, la qual arrosseguem fins on ens interessi per tal de que des d'allà ja puguem començar a editar nodes:



Aquesta opció ens genera un codi que automàticament també programa en ASP, el nombre de "fills" de cada node, el valor, que ve donat des d'una funció, la seva mida i colors, així com si volem que els nodes siguin visibles, seleccionats per defecte, o bé duguin annexada una imatge que alhora pugui servir de llegenda.

A continuació veiem una mostra de part del codi del nostre arbre creat anteriorment a mà, i des del qual cridem funcions, donem nom a cada fill i on especifiquem totes les seves característiques, la gran majoria elaborades de forma automàtica i senzilla.

```

style="position: absolute; height: 500px; z-index: 1; left: 222px; top: 40px; width: 500px;"
&nbsp;&nbsp;&nbsp;<asp:TreeView ID="TreeView1" runat="server" onclick="OnTreeNodeChecked();"
style="z-index: 2; left: -206px; top: -30px; position: absolute; height: 24px; width:
ShowLines="True">
<Nodes>
<asp:TreeNode Text="CAPES DEL SIG" Value="CAPES DEL SIG" Expanded="False"
SelectAction="None">
<asp:TreeNode Text="Territori" Value="Territori" Expanded="False"
SelectAction="None">
<asp:TreeNode ShowCheckBox="True" Text="Limits Municipals" Value="1"
></asp:TreeNode>
<asp:TreeNode ShowCheckBox="True" Text="Districtes" Value="2"
SelectAction="None"></asp:TreeNode>
<asp:TreeNode ShowCheckBox="True" Text="Toponimia" Value="3"
SelectAction="None"> </asp:TreeNode>
<asp:TreeNode ShowCheckBox="True" Text="Ambients Forestals" Value="4"
SelectAction="None"> </asp:TreeNode>
<asp:TreeNode ShowCheckBox="True" Text="Parcs i Jardins" Value="5"
SelectAction="None"> </asp:TreeNode>
</asp:TreeNode>

```

6.1.2.3 Disseny de l'opció esborrar capes.

Una altra opció que considerem necessària, sobretot tenint en compte que els kml's amb opció a activar-se arriben pràcticament a la norantena, és la de poder invisibilitzar al complet totes les capes activades amb un sol clic. L'input button de la caixa d'eines del programa dissenya automàticament també aquest botó, del qual també penja una funció que s'ha de programar en llenguatge javascript.

```
<input id="Button2" type="button" value="Borrar Mapa"
onclick="return Esborrar()" style="font-size: x-small" /><br />
<div id='map3d' </div
```

En cas de saturació, confusió o senzillament per raons de comoditat, l'input button realitza la neteja de les capes resetejant la visualització i tornant l'usuari a la situació inicial de l'aplicació.

6.2.1 Programació javascript de les funcions de l'arbre

Però programar les funcions que ens permeten carregar els kml's del servidor i relacionar-los amb el disseny de l'arbre ASP no és tan fàcil. Javascript contrasta amb les facilitats d'aquest últim i hem de dur a terme un veritable treball de recerca al respecte per tal d'elaborar, o fins i tot senzillament trobar, estructurades mitjançant webs i fòrums especialitzats, les funcions definitives.

6.2.1.1 La càrrega

Aquesta representa la funció més bàsica ja que és la que ens permet carregar la informació a l'arbre. En un inici, abans de descobrir el bucle que relaciona l'arbre amb les funcions de càrrega i invisibilització, farem ús d'un input button que carrega les capes i d'un altre que les descarrega.

```

function Carrega() {
    var j = 0;
    var inputs = document.forms[0].elements;
    for (var i = 0; i < inputs.length; i++) {
        if (inputs[i].type == "checkbox") {
            j = j + 1
            if (inputs[i].checked) {
                num = j
                if (fileLoad[num] != 99) {
                    //alert("Primera Càrrega...");
                    exemplecapes(num);
                }
                if (fileLoad[num] = 99) {
                    //alert("Ja estic carregat. Em faig visible");
                    file[num].setVisibility(true);
                }
            }
        }
    }
}

```

Alhora de carregar doncs, fem ús de la variable fileload (declarada en anterioritat) la qual fa referència a qualsevol cas (número) igual o inferior a 99, que passant per la funció “exemplecapes” identifica el servidor mitjançant el número de la mateixa funció “exemplecapes”.

```

var fileLoad = (
    '1': 0,
    '2': 0,
    '3': 0,
    '4': 0,
    '5': 0,
    '6': 0,
    '7': 0,
    '8': 0,
    '9': 0,
    '10': 0,
    '11': 0,
    '12': 0
)

function exemplecapes(num) {
    var url;
    switch (num) {
        case 1:
            //numf = 1
            url = 'http://80.25.198.237/kml/Ambits/municipiscpc.kmz'
            break;
        case 2:
            url = 'http://80.25.198.237/kml/Ambits/districtes.kml'
            break;
        case 3:
            //...
            break;
        case 88:
            url = 'http://80.25.198.237/kml/Incendis/Incendis1990.kml'
            break;
        default:
            alert("No cap Arxiu Accosiat a la Viariable File" + num)
    }
    file[num] = getML(url);
}

```

```

function getNL(kmlURL) {
    //document.getElementById("Text1").value = kmlURL;
    var nl = ge.createNetworkLink("");
    var link = ge.createLink("");
    link.setHref(kmlURL);
    nl.setLink(link);
    if (document.getElementById("ExpandNav").checked) {
        nl.setFlyToView(true);
    }
    ge.getGlobe().getFeatures().appendChild(nl);
    return nl;
}

var file = {
    '1': null,
    '2': null,
    '3': null,
    '4': null,
    '5': null,
    '6': null,
    '7': null,
    '8': null,
    '9': null,
    '10': null,
    '11': null,
    '12': null
}

```

Per últim, la funció `getNetworkLink` ens ajuda a obtenir la url per poder acabar de carregar cada cas (número) a la primera funció de “càrrega”. Cada url va lligada a un cas que és carregat des del servidor. Posteriorment un altre bucle busca les capes carregades (números) que concorden amb la variable `file` i les fa visibles.

Classificant les capes amb la respectiva numeració i enllaç web al servidor, ja estem a disposició de carregar, i per tant, visualitzar les capes i migrar la informació GIS a l'aplicació.

6.2.1.2 La invisibilització de les capes

Seria absurd haver de carregar les capes una vegada i una altra cada vegada que les seleccionéssim des de l'aplicació. És per això que s'ha de buscar la manera de fer-les invisibles sense haver de tornar a fer el llarg procediment exposat a l'últim punt. Així doncs, per a què poguem fer-les invisibles, en cas de que les haguéssim seleccionat, dissenyem un botó especial el qual se serveix d'una funció d'“invisibilització”, abans de descobrir el bucle que lliga definitivament aquesta funció als “checks” de l'arbre.

```

function invisibilitza() {
    var num;
    var j = 0;
    var inputs = document.forms[0].elements;
    for (var i = 0; i < inputs.length; i++) {
        if (inputs[i].type == "checkbox") {
            j = j + 1
            if (!inputs[i].checked) {
                num = j
                if (file[num] != null) {
                    file[num].setVisibility(false)
                }
            }
        }
    }
}

```

Com s'exposa, la funció aquest cop no carrega a partir d' "exemplecapes" ni tampoc va a buscar la variable fileload. Senzillament detectant a partir de la variable file quin és el cas carregat, la funció fa invisible des del check de l'arbre la capa del cas seleccionat a priori.

6.2.1.3 La funció d'esborrar

Aquesta funció, que en gran mesura és la mateixa que la d'invisibilització, representa el mateix procediment que aquesta amb dues grans diferències. La primera que enlloc d'operar des del check de cada cas de l'arbre, ataca des del botó dissenyat anteriorment amb ASP a tots els casos seleccionats (i per tant carregats) de l'arbre. I la segona, que amb aquest fet, ens retorna, en qüestió de capes visualitzades, on érem a l'inici de l'aplicació. Notis que el bucle d'inputs seleccionats fa referència primer als elements seleccionats i en segon lloc falseja la seva visibilitat. Tan sols una línia diferencia una funció de l'altra.

```
function Esborra() {
    var num;
    var j = 0;
    var inputs = document.forms[0].elements;
    for (var i = 0; i < inputs.length; i++) {
        if (inputs[i].type == "checkbox") {
            inputs[i].checked = false;
            j = j + 1
            num = j
            //alert(num)
            if (file[num] != null) {
                //alert(num)
                file[num].setVisibility(false)
            }
        }
    }
}
```

6.2.1.4 La clau final: La funció OnTreeNodeChecked

Abans d'arribar a la funció OnTreeNodeChecked recordem que bàsicament gaudim de tres botons amb tres funcions principals diferents. El primer, un botó amb la funció "carrega" que carrega a través de diverses variables i amb l'ajuda d'altres funcions els checks seleccionats a l'arbre. En segon lloc un altre botó que desselecciona les capes carregades per la funció carrega, però tan sols ho fa en les de l'últim cas. I finalment, també partim d'un botó amb una funció "esborrar" que no tan sols invisibilitza les capes carregades durant l'última càrrega sinó que ho fa amb totes i cadascuna d'aquelles que resten actives.

La funció que més es fa difícil correspon a la que des del mateix arbre carrega, invisibilitza i pot tornar a visibilitzar cada check en particular sense necessitat de recórrer a la crossa d'un botó que remolqui les seleccions o desseleccions de cada cas particular de l'arbre.

En definitiva la funció final que ens permet obrir un arbre amb èxit és l'OnTreeNodeChecked que a partir d'un bucle decideix segons l'estat del "check", quina funció – de càrrega ,invisibilització o revisibilització"- entra en joc. I és així com finalment carreguem o invisibilitzem les capes, passant per totes les funcions i variables descrites.

```
function OnTreeNodeChecked() {  
  
    var obj = window.event.srcElement;  
    var treeNodeFound = false;  
    var checkedState;  
    if (obj.tagName == "INPUT" && obj.type == "checkbox") {  
        var treeNode = obj;  
        checkedState = treeNode.checked;  
        if (checkedState == true) {  
            //alert(checkedState);  
            Carrega();  
        }  
        else {  
            //alert(checkedState);  
            invisibilitza();  
        }  
    }  
}
```

Per tal de resumir tal garbuix de funcions i variables, a continuació mostro el recorregut que fa el "clic" a l'arbre i el "clic" al botó 'esborra' mitjançant un esquema força clarificador.

Com veiem, el botó 'esborra' sols pot anar a la funció esborra i a la variable file per fer un 'reset' a l'aplicació implicant a totes les capes de l'arbre en la invisibilització.

A l'arbre hi ha l'opció de carregar les capes una per una passant per fileload, i invisibilitzar-les individualment mitjançant la funció "invisibilitza". GetNetworklink és la funció de crida de capes de Google Earth, i per tant la que fa que l'arbre sigui idoni per a l'apartat 'SIG' en el qual ja es poden dur a terme consultes cartogràfiques de visualització, destriant-ne la informació desitjada.

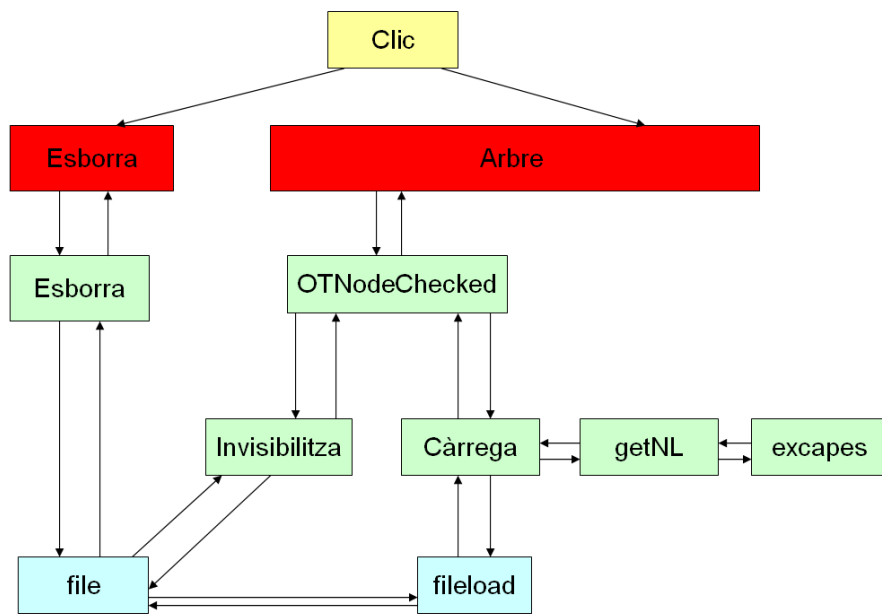


Fig 28: Esquema de l'arbre. Font: Elaboració pròpia

6.2.2 La funció de càrrega de cartografia a Google Maps

Tanmateix, per a les aplicacions sectorials no fa falta tal complexitat de funcions atès que l'apartat actua com a marc en el qual tan sols es pot interactuar amb els zooms i la informació de la cartografia.

Com avançàvem en un inici, l'API de Maps és l'encarregada de carregar la cartografia d'aquests marcs sectorials i la funció que permet carregar la cartografia per a sobreposar-la a la interfície és l'anomenada funció GGeoxml:

```
function GGeoxml() {  
  if (toggleState == 1) {  
    map.removeOverlay(geoXml);  
    toggleState = 0;  
  } else {  
    map.addOverlay(geoXml);  
    toggleState = 1;  
  }  
}
```

GGeoxml fa la funció que anteriorment en el cas de l'apartat SIG de Google Earth feia getNetworkLink, per tant si volguéssim fer ús del l'arbre amb Google Maps hauríem de canviar, com a mínim, aquesta última per GGeoxml, així com totes les referències d'Earth per les de Maps. En el nostre cas però aquesta ens és suficient per a carregar la cartografia sense el recolzament de botons i arbres dins les aplicacions sectorials.

6.3 Programació de vinyetes emergents

Per tal de complir un altre dels grans objectius del projecte fixats pel consorci hem de poder dotar d'informació diferencial a cada punt, línia o bé polígon més enllà de les diferències de posició o forma que ja es fan paleses de facto.

El fet de que amb un sol clic tinguem accés a la informació desitjada d'un restaurant, sender o reserva natural ens permet interaccionar de forma dinàmica amb el mateix objecte sense haver de reiniciar l'aplicació web i sense renunciar a la nostra escala i posicionament de visualització actuals, ja que la informació cartogràfica duu incorporada la alfanumèrica i els enllaços a altres webs, o bé informació publicada al servidor com ara fotografies o documents pdf. De fet, per aquest motiu és interessant la troballa de la DLL de Mapwindow GIS i el paper de la d'ArcMap.

Tal i com el programa de Google Earth o la mateixa web de Google maps ens ofereixen amb la seva vinyeta emergent, pretenem complementar l'espectacularitat espacial d'earth amb un bon accés a dades alfanumèriques.



Figura 29: El consorci va posar-se com a fita poder elaborar vinyetes semblants a les de google/maps

Font: <http://maps.google.es/>

6.3.1 El disseny i entrellat d'un KML

La programació però d'aquestes vinyetes va acompanyat, per sorpresa meua, de la programació del codi que conforma la cartografia i no pas d'una crida a una base de dades, tal i com s'havia descrit a la planificació. Deixant de banda explorar altres opcions menys senzilles, des de Vallvidrera optem per aprofitar al màxim els avantatges dels KML, que no són altres que els de poder accedir a dades alfanumèriques de l'antic shape o bé programar referències com a direccions dins d'un servidor públic entre el codi del KML.

Quan un shape és convertit a KML, aquest esdevé, si s'obra, en un entrellat de codi el qual una part és directament poc útil, o bé no influeix la seva presència o absència. Després de conèixer a fons l'estructura dels KML, aquest neix amb tres parts però, que en el meu parer sí són essencials:

1- Els atributs alfanumèrics variables del shape

Els atributs de l'antic shape queden reflectits dins l'anomenat "schema name", que integra el nom de l'atribut, en aquest cas amb els exemples de "VOLUM", "TIPUS" i "NOM". També veiem definits atributs, no de l'antic shape sinó del nou KML, com ara el nom del KML o bé la seva descripció, però que en aquest cas són generals. Els atributs variables identifiquen els valors de cada punt, línia o polígon i els mostren segons es programa cada entitat espacial dins el KML, anomenada pel codi com a "placemark".

```
<Document>
  <name>fbasshelpt</name>
  <open>1</open>
  <Snippet maxLines="0"></Snippet>
  <description>Exported from fbasshelpt on 15/11/10</description>
  <Schema name="fbasshelpt" id="fbasshelpt_schema">
    <SimpleField type="int" name="VOLUM"><displayName>VOLUM</displayName>
  </SimpleField>
    <SimpleField type="short" name="TIPUS"><displayName>TIPUS</displayName>
  </SimpleField>
    <SimpleField type="string" name="NOM"><displayName>NOM</displayName>
  </SimpleField>
```

2- Els atributs alfanumèrics invariables del shape i el disseny de la vinyeta

Aquesta segona estructura representa tots aquells atributs que són invariables dins la vinyeta de cada punt, polígon o línia. Cada “placemark”, per tant, comparteix aquest disseny i depèn de l’editor que es comparteixin més o menys variables. En el cas del consorci ens interessa en alguns casos no tan sols un model de vinyeta emergent que comparteixi un disseny homogeni, sinó que també logos, links o documents pdf puguin ser comuns a cada entitat espacial diferenciada dins un mateix KML.

A continuació exposo en primer lloc l’”style”, que recorda als placemarks quin és el seu disseny de vinyeta, la url d’una icona i el text que ha d’acompanyar les dades. La dada de volum, per exemple, la pot acompanyar un “m3” i demés text descriptiu comú. Les tres fotografies i el logo estan marcades per un “width” que indica la grandària de la imatge, i la direcció és la que correspon al servidor públic del consorci del parc. D’altra banda sota “icon” també s’especifica la direcció i la imatge que representa cada símbol puntual (placemark) del KML (en aquest cas cada bassa de la capa “basses”):

```
<Style id="ballon_basses">
<IconStyle>
<scale>0.583333</scale>
<Icon>
<href>http://80.25.198.237/kml/icons/png/bassahelicopter.PNG</href>
</Icon>
</IconStyle>
<BalloonStyle>
<text><![CDATA[<h2>Bassa: ${fbasshelpt/NOM} - Volum: ${fbasshelpt/VOLUM} m3 </h2>
<center>
Bassa Helicòpters Bombarders del Dispositiu de<br>
Prevenió d'Incendis de l'Àrea Metropolitana de Barcelona<br>
</center>
<center>

<center>
<center>

<center>
<center>

<center>
<center>
<br>
<a href="http://213.27.212.79/~parccoll/pages/en_bici/">NOVA WEB PARC COLLSEROLA!</a>
<br>
<br>
<hr>]]></text>
</BalloonStyle>
</Style>
```

3- Els elements cartogràfics o “placemarks”

En tercer lloc, cadascuna de les “entitats espacials”, com ara les basses, que van unides sota un KML i que per tant pertanyen a una classificació determinada, segueixen essent objectes semii independents gràcies als atributs de les variables de l'antic shape, i per suposat per les coordenades que especifiquen la referenciació espacial de l'element o georreferenciació.

Com veiem a continuació, l'anomenat placemark (cada bassa) fa referència primer a les variables de l'schema name (ara anomenat schema data), al disseny de la vinyeta i als atributs invariables (styleurl). Finalment apareixen també unes coordenades que són les que fan aparèixer la imatge de cada símbol puntual en el punt corresponent dins el mapa:

```
<Placemark>
<name>2</name>
|<Snippet maxLines="1">${fbasshelpt/VOLUM} ${fbasshelpt/NOM} ${fbasshelpt/TIPUS}</Snippet>
  <styleUrl>#ballon_basses</styleUrl>
  <ExtendedData>
    <SchemaData schemaUrl="#fbasshelpt_schema">
      <SimpleData name="VOLUM">12</SimpleData>
      <SimpleData name="TIPUS">3</SimpleData>
      <SimpleData name="NOM">Bombers Vallvidrera</SimpleData>
    </SchemaData>
  </ExtendedData>
  <Point>
    <coordinates>2.11771344772211,41.4166026820705,0</coordinates>
  </Point>
</Placemark>
```

L'estudi de cas exposat fa referència a un símbol puntual com les basses, ja que cada placemark, com es mostra, sols està descrit per una coordenada X i per una coordenada Y. És així com ens podem formular perfectament la pregunta de “ i com soluciona el problema de les coordenades un KML en forma de línies o polígons?” Doncs molt fàcil, el placemark integra cada desviament de la trajectòria de la línia o polígon mitjançant el respectiu origen i final de cada punt a tenir en compte i en descriu les coordenades.

I així successivament.

Com podem observar a continuació, l'espai “coordinates” del placemark s'eixampla fins a integrar totes les coordenades que permeten unir un polígon o línia fins a la seva totalitat.

```

<Placemark>
  <name><![CDATA[1]]></name>
  <Snippet maxLines="1"><![CDATA[ $[LimitCPC/HA] $[LimitCPC/PERIMETER]]></Snippet>
  <styleUrl>#FEATURES</styleUrl>
  <ExtendedData>
    <SchemaData schemaUrl="#LimitCPC_schema">
      <SimpleData name="PERIMETER">71335,87942</SimpleData>
      <SimpleData name="HA">8465,3579</SimpleData>
    </SchemaData>
  </ExtendedData>
  <Polygon>
    <extrude>0</extrude>
    <altitudeMode>clampedToGround</altitudeMode>
    <outerBoundaryIs>
      <LinearRing>
        <coordinates>
          2.17229873407849,41.4590896689721,0
          2.17212006136454,41.4590163239722,0
          2.17197684853032,41.4589792635572,0
          2.17173774065008,41.4589505199162,0
          2.17153386295045,41.458976073924,0
          2.17129360869999,41.4590373951096,0
          2.17112518962533,41.4590992334434,0
          2.17087273249663,41.459178480615,0
          2.1707886947155,41.4591958896412,0
          2.17053795828546,41.4591400374783,0
          2.17028768148924,41.4590481584838,0
          2.1701449289391,41.4589750695089,0
          2.16989453848968,41.4588921962392,0
          2.16971563933042,41.4588368606747,0
          2.1694647903259,41.4587900127614,0
          2.16926091230381,41.4588155627376,0
          2.16916398240499,41.4589049366717,0
          2.16904195737348,41.4590842032388,0
          2.16896839708349,41.459218782818,0
        </coordinates>
      </LinearRing>
    </outerBoundaryIs>
  </Polygon>

```

Així arribem a la conclusió que l'escala del KML no influeix en el seu "pes informàtic" sinó que realment ho fa la quantitat de vegades que aquest ha d'emmagatzemar un canvi de direcció de línia del KML. L'inconvenient que aquest mètode té en moltes ocasions és que en cas de tenir molts canvis de direcció (i per tant ser més detallat), pot arribar a retardar la seva càrrega uns segons i en alguns casos en l'API de Google Maps s'ha de comprimir a KMZ o bé simplificar, atès que és sensible a la grandària del KML, i per tant a la quantitat de coordenades i informació que ha d'emmagatzemar.

6.3.2 Mètodes per a l'obtenció de vinyetes emergents de qualitat

Els KML en si ja fan aparèixer una petita vinyeta per defecte un cop l'usuari ha fet 'clic'. Normalment sense una programació prèvia la vinyeta és buida o bé mostra qualsevol atribut de l'antic shape, com ara l'ID, el nom, o bé la superfície. L'inconvenient és que en la majoria d'ocasions l'atribut que mostra per atzar no és el desitjat, està mal dissenyat, o bé és una mera dada sense importància (p. Ex a la imatge mostra l'ID).



Fig 30: En aquest cas la vinyeta per defecte mostra l'identificador dels centres d'informació. Font: Elaboració pròpia

És així com a partir d'ara la nostra primera fita és la de poder canviar l'aspecte de les vinyetes, i per descomptat programar la informació que ha de mostrar.

6.3.2.1 Programació de vinyetes mitjançant plantilles

La guia de codi de la pàgina web de Google ofereix la possibilitat d'aprofitar unes plantilles programades de manera prèvia, i així guanyar fàcilment en bellesa i oferir la informació que creguem oportuna sobre cada objecte de manera ordenada. Bàsicament la plantilla consisteix en un "balloon style" (és a dir la part fixa del codi de la vinyeta del kml) que s'enganxa dins la descripció de cada placemark i que mostra, mitjançant l'editor de marca de posició del programa Google Earth, els llocs on van els enllaços, texts, fotografies etc.

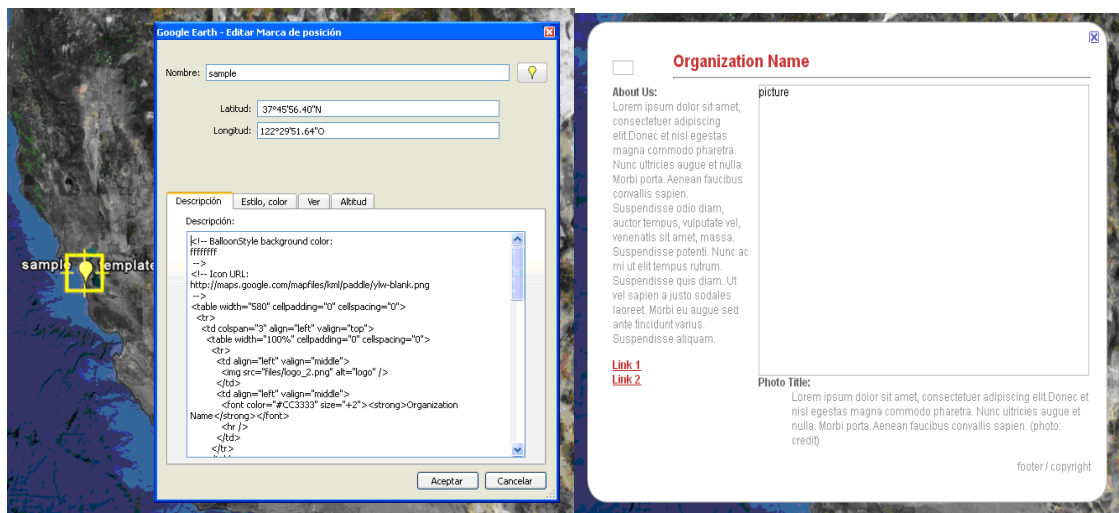


Fig 31: A l'esquerra, un exemple de com una marca de posició és fàcilment editable enganxant el codi de google.code.com. A la dreta, el resultat. Preparat per a ser degudament personalitzat. Font: Elab. Pròpia.

D'una banda doncs, les plantilles resolen aparentment el problema de les vinyetes i la informació adicional, la qual el consorci desitja disposar des del mateix visor. Els problemes però arriben quan, una vegada i una altra, hom ha de programar dins el programa d'Earth tots els kml, placemark a placemark. Si teníem en compte que la quantitat de kml/kmz frega la norantena i que la mitjana de placemarks per kml pot superar fàcilment la vintena, estariem parlant d'una laboriosa tasca que a més a més hipotecaria la posterior reedició de nova cartografia.

6.3.2.2 Programació de vinyetes mitjançant variables

Però com s'especifica anteriorment, l'estructura del kml permet moltes possibilitats les quals val la pena explotar. La qüestió és aconseguir crear una variable que memoritzi les propietats de cada placemark i les incorpori a les respectives vinyetes emergents.

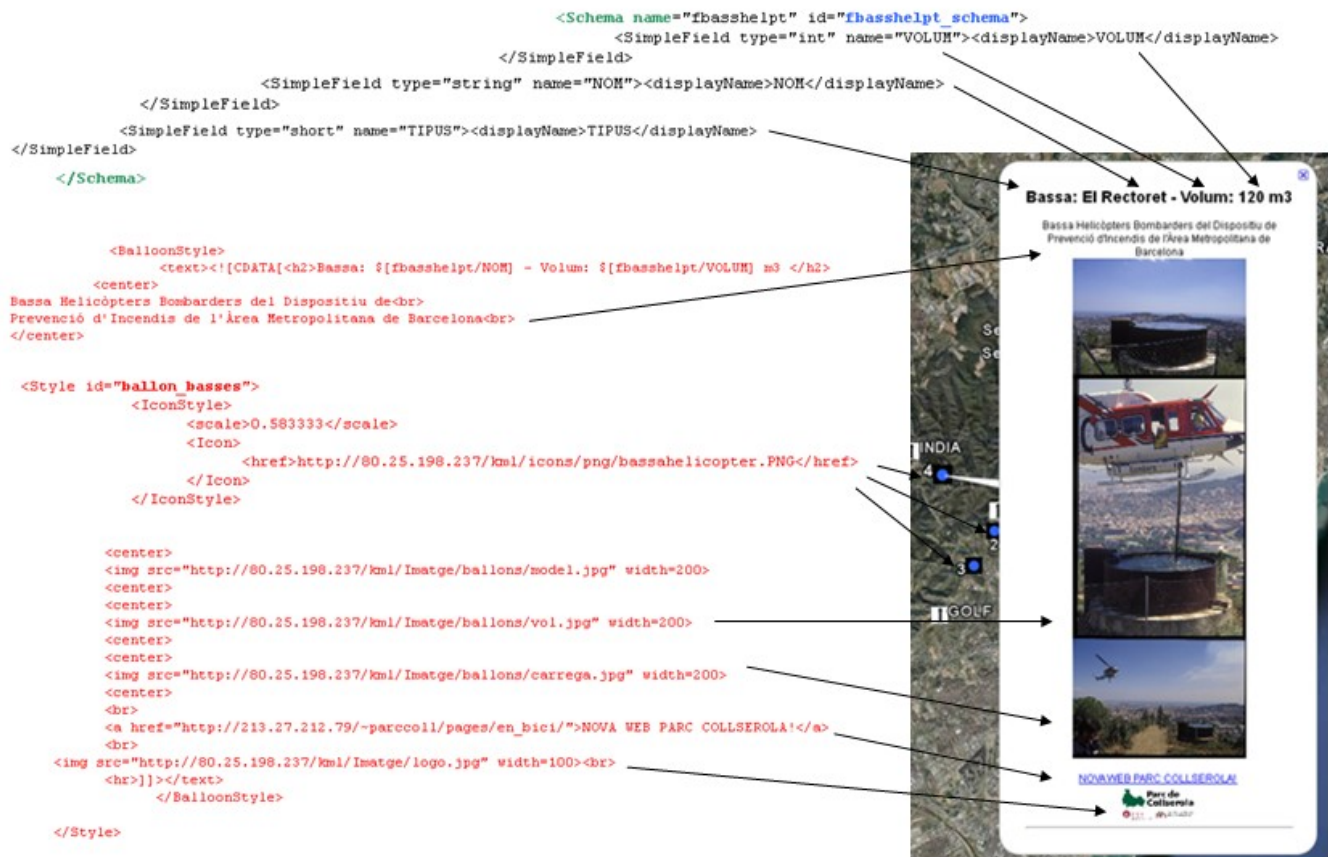


Fig: 32 L'entrellat variable i invariable d'un kml. Font: Elaboració pròpia.

Com s'observa a l'exemple, el placemark puntual de bassa antiincendis número 4, té com a vinyeta emergent una part variable (codi negre), i una part fixa (codi en vermell). D'aquesta manera el nom de la bassa, en aquest cas El Rector, i el seu volum, 120m3, representen la informació exclusiva que cal programar dins l'schema name per tal de que s'especifiqui dins de cada bassa (placemark). El Balloon style representa l'estil compartit per a cada vinyeta, i per tant un codi que tan sols ha estat programat una vegada i el qual també fa ús de variables per a fer aparèixer a partir de l'ID "balloon_basses", les tres fotografies, el text referent al dispositiu de prevenció

d'incendis i finalment l'enllaç directe a la web del Parc de Collserola i el logo del Consorci.

A continuació veiem el codi que conforma el placemark puntual de la bassa en qüestió. L'"style url" pren la variable de l'style ID de la part invariable de codi del kml. Així veiem en vermell com també no tan sols les fotografies i els enllaços es referencien al "balloon style" sinó que també, com a part fixa de programació, és comuna la "icon style" que cerca al servidor una imatge en format .PNG i que dona forma al símbol puntual sobre el qual hem de fer clic per a fer emergir la vinyeta.

```
<Style id="ballon_basses">
  <IconStyle>
    <scale>0.583333</scale>
    <Icon>
      <href>http://80.25.198.237/kml/icons/png/bassahelicopter.PNG</href>
    </Icon>
  </IconStyle>
</Style>

<Placemark>
  <name>3</name>
  <Snippet maxLines="1">${fbasshelpt/VOLUM} ${fbasshelpt/NOM} ${fbasshelpt/TIPUS}</Snippet>
  <styleUrl>#ballon_basses</styleUrl>
  <ExtendedData>
    <SchemaData schemaUri="#fbasshelpt_schema">
      <SimpleData name="VOLUM">12</SimpleData>
      <SimpleData name="TIPUS">3</SimpleData>
      <SimpleData name="NOM">Vila Paula</SimpleData>
    </SchemaData>
  </ExtendedData>
  <Point>
    <coordinates>2.11025726586921,41.4072217896154,0</coordinates>
  </Point>
</Placemark>

<Schema name="fbasshelpt" id="fbasshelpt_schema">
  <SimpleField type="int" name="VOLUM"><displayName>VOLUH</displayName>
</SimpleField>
  <SimpleField type="string" name="NOM"><displayName>NOM</displayName>
</SimpleField>
  <SimpleField type="short" name="TIPUS"><displayName>TIPUS</displayName>
</SimpleField>
</Schema>
```

El codi encapçalat en verd a sota fa referència a les variables de dins el placemark. Volum, tipus o nom són dades simples que fan referència a aquestes, i per tant a l'schema name, l'schema data del placemark.

7. Explotació i presentació de resultats

Per tal d'esvaïr qualsevol dubte del lector sobre el paper de les dues interfícies de Google dedico especialment la presentació dels resultats a mostrar la diferència de filosofia entre els dos apartats. I és que ja ho diuen que una imatge de vegades val més que mil paraules.

A continuació tenim el projecte de web corporatiu. Naturalment jo no he programat tal complexitat però em serveix per a exposar en quines pestanyes els usuaris aniran a l'apartat SIG on hi haurà Google Earth i l'apartat de guia de Serveis on hi haurà Google Maps, els quals sí he col·laborat a elaborar.

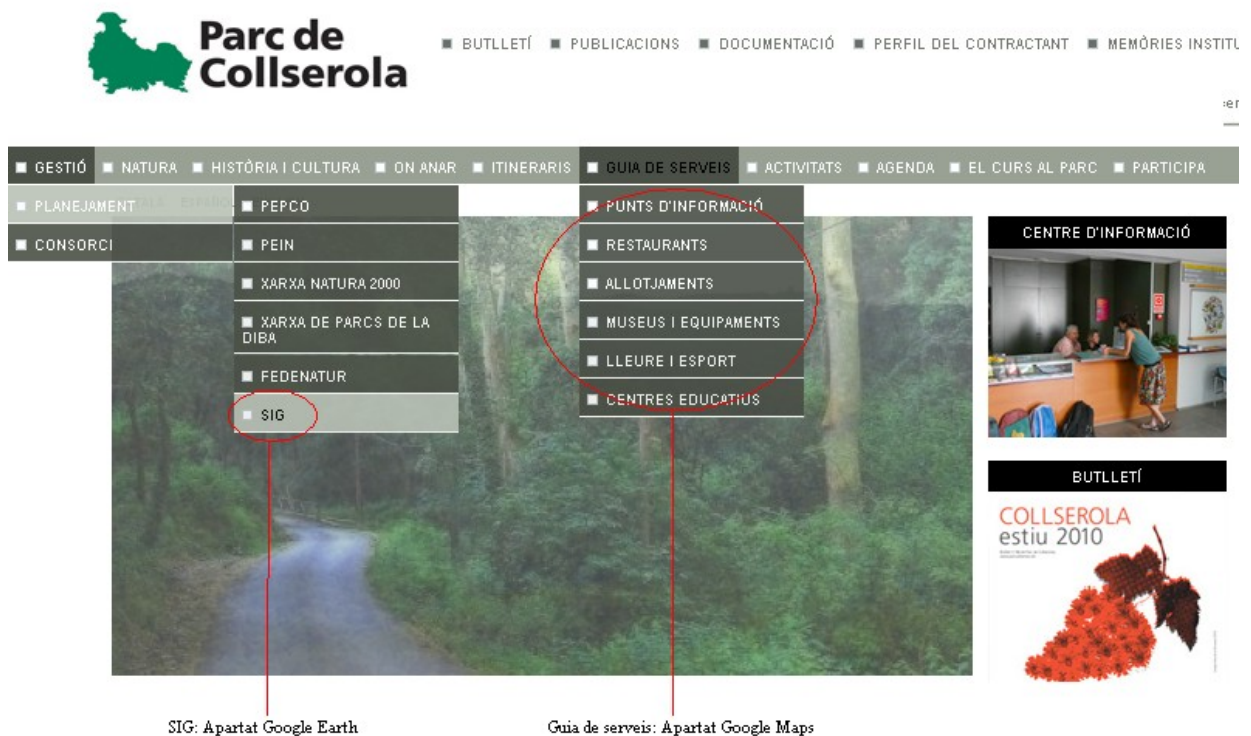


Fig 33: Web corporatiu en estat de proves. Font: Elaboració pròpia.

7.1 L'apartat SIG i Google Earth

Com veiem a continuació, un cop dins es pot observar l'arbre en ple rendiment i amb totes les capes classificades, activades i desactivades al gust de l'usuari. També podem observar els botons d'esborrar i centrar el mapa, així com els controls de navegació.

No obstant pot sobtar la presència de la barra d'estat dels kml –de color verd- que indica en quin estat es troben els kml que a petició de l'usuari s'han activat des de l'arbre, si s'estan carregant o bé si ja s'han carregat. També veiem l'opció de davallar el kml/z activat, així com les coordenades sobre les quals tenim el ratolí. Tots aquests, representen els requeriments que anunciàvem en l'apartat 3.2, que el Consorci ja tenia programats i que ha cregut oportú afegir, sense que la meua col·laboració hi interferís.

L'exemple representa la xarxa bàsica de camins en groc, els itineraris temàtics en blau cel, i finalment les passejades a peu en blau més fosc, totes en simbolització lineal.

The screenshot displays the web interface for the Parc de Collserola's SIG. At the top, there is a logo and a navigation menu with items like 'BUTLLETÍ', 'PUBLICACIONS', 'DOCUMENTACIÓ', 'PERFIL DEL CONTRACTANT', 'MEMÒRIES INSTITUCIONALS', and 'INTRANET'. Below this is a search bar and a secondary menu with 'GESTIÓ', 'NATURA', 'HISTÒRIA I CULTURA', 'ON ANAR', 'ITINERARIS', 'GUIA DE SERVEIS', 'ACTIVITATS', 'AGENDA', 'EL CURS AL PARC', and 'PARTICIPA'. The main area contains a map with a network of paths. On the left, a layer tree under 'CAPES DEL SIG' lists various layers, with 'Xarxa Bàsica de Camins', 'Itineraris Temàtics', and 'Passejades a Peu' checked. On the right, a control panel includes options for 'Barra d'Estat', 'Control Navegació', 'Xarxa Coordenades', 'Mapa de Situació', 'Escala Gràfica', 'Capa Atmosfera', 'Navegació x Mouse', and 'Expandir Capa'. Below the map, there are buttons for 'Borrar Mapa' and '<<Centrar Mapa'. At the bottom, a URL bar shows 'http://80.25.198.237/kml/Ambits/LimitCPC.kmz' and buttons for 'DOWNLOAD' and 'VIEWKML'.

Fig 34. Apartat SIG del web corporatiu. Font: elaboració pròpia.

Tanmateix, a continuació també m'agradaria posar dues demostracions més del funcionament de l'aplicació: les parcel·les urbanes com a polígons, i els restaurants i estacions de transport públic representats com a punts.

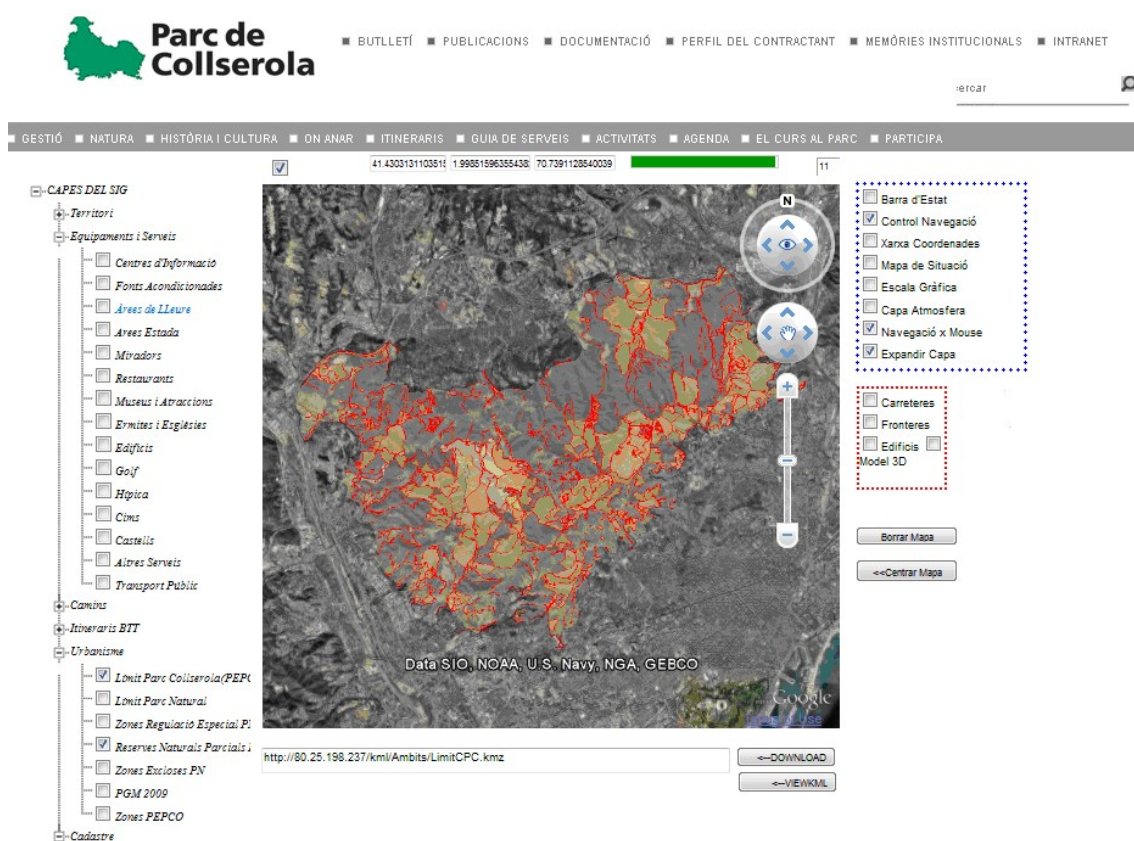


Fig 35. Apartat SIG del web corporatiu: polígons. Font: elaboració pròpia.

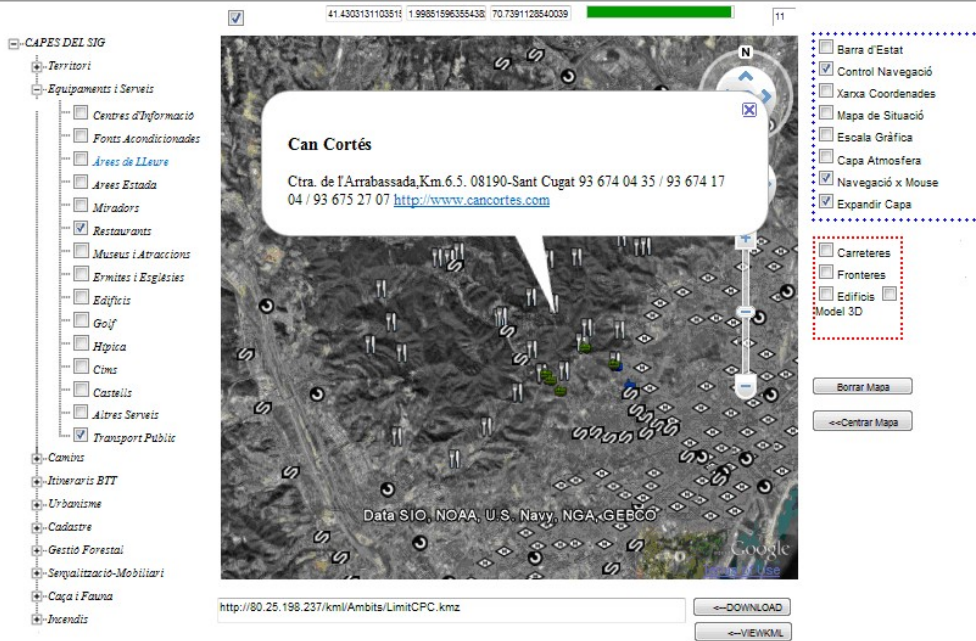


Fig 36. Apartat SIG del web corporatiu: punts. Font: elaboració pròpia.

Com veiem, al kml de restaurants s'hi pot extreure una vinyeta emergent que n'indica nom, adreça, web i telèfon.

A continuació tenim més exemples d'explotació de vinyetes, en aquest cas he retallat el marc de la cartografia, l'he enganxat i l'he comparat amb què se'n deriva de fer clic sobre les torres de vigilància. El resultat és una vinyeta amb el logo del parc, una foto d'una torre, les coordenades UTM exactes de localització de la mateixa, el nom, un text, i un enllaç.

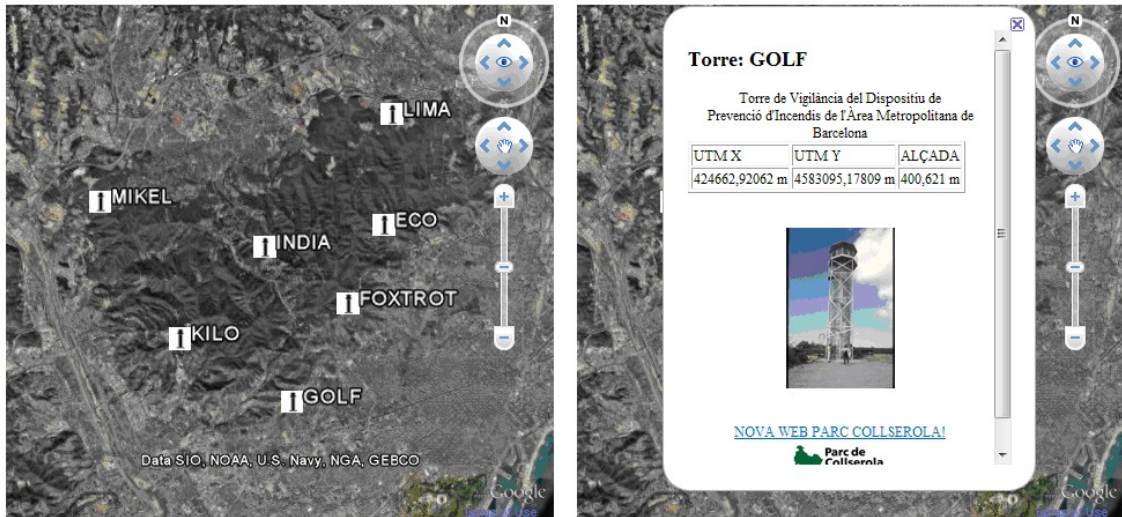


Fig 37. Apartat SIG del web corporatiu: vinyeta emergent. Font: elaboració pròpia.

Malgrat tot, l'apartat Earth està encara en període de proves i s'ha d'acabar de retocar per part dels informàtics, que estan desenvolupant el web corporatiu del Consorci. Probablement l'apartat incorpori bànners, adquireixi colors, o bé la lletra de l'arbre, i llurs formes canviïn després del "lifting" dels professionals de les webs. Això però ja és un altre debat lluny del SIG i de la tasca encarregada a la col·laboració.

7.2 L'apartat de 'Guia de serveis' i Google Maps

L'apartat que sí ja pot considerar-se una fita del tot aconseguida és el de Google Maps que apareix com un marc d'ajuda per a la guia de serveis. A la següent pàgina tenim un tast del que representen els serveis a nivell de restaurants i serveis educatius. Les vinyetes emergents que se'n deriven de cada restaurant/centre responen també doncs a l'adreça, web, nom o el telèfon del servei.

També m'ha semblat interessant mostrar l'opció d'alternar la foto aèria amb la cartografia de Maps; la primera distingeix perfectament la negror del símbol puntual que representa els restaurants i en destaca la seva distribució, i en segon lloc, l'híbrid permet distingir els centres educatius fent que destaquin per sobre el verd obscur de la imatge, tot, sense deixar de banda els principals carrers i les vies de comunicació.

Alhora d'explotar els resultats hi hagué, no obstant, un petit contratemps, i és que a Google Maps obligatòriament tots els símbols han de tenir obligatòriament una mida de 32x32 píxels fet que obliga a minimitzar manualment el símbol i fer-ne transparent

bona part del contorn, així hom s'assegura que fos estètic i no se solapi amb la cartografia o altres símbols.



Fig 38. Símbol amb contorn transparent a l'apartat Maps. Font: elaboració pròpia

A la següent imatge hi ha seleccionats el restaurant de Can Castellví i l'Escola Bressol Municipal de La Puput a Barcelona, respectivament, amb la simbologia adaptada. Observis que a l'esquerra de les imatges apareixen bànners o altres apartats als quals accedir des de la guia, també programats pels dissenyadors web del parc, i aliens al mèrit meu i del tutor que queden reduïts, en aquest cas, al marc de la cartografia.

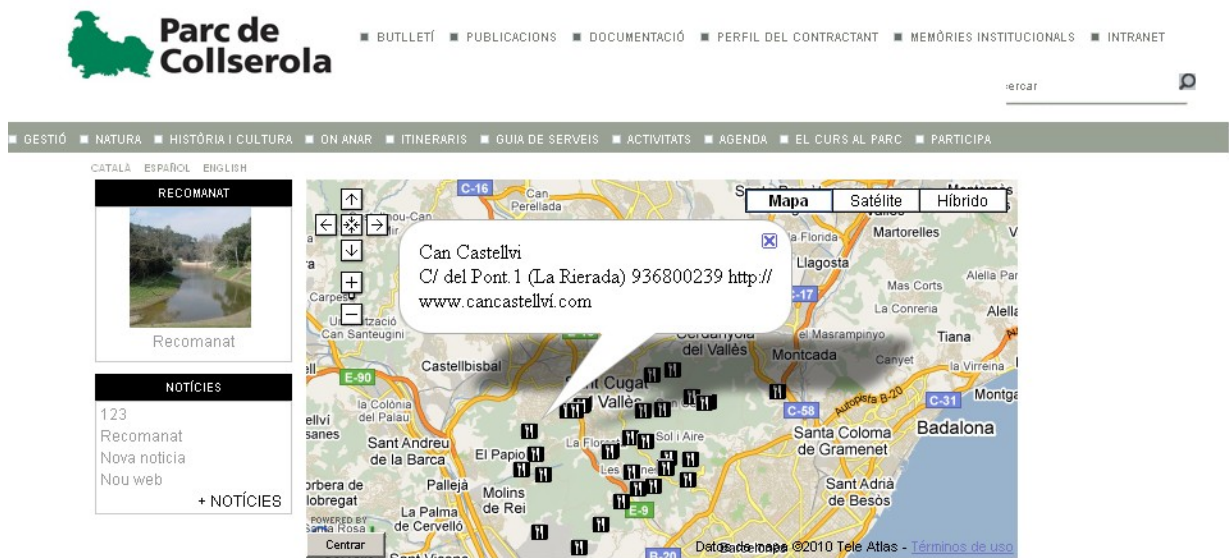



Fig 39. Apartat Guia de serveis, restaurants. Font: elaboració pròpia

RECOMANAT



Recomanat

NOTÍCIES

123
Recomanat
Nova notícia
Nou web
+ NOTÍCIES



Fig 40. Apartat Guia de serveis, centres educatius. Font: elaboració pròpia

Si finalment prenem com a antecedent més recent el projecte del mòdul de programació web, podem arribar a extreure ja, pel que fa a les explotacions, les primeres conclusions.

D'una banda les API carreguen la informació cartogràfica força més ràpid que les imatges WMS, i l'accés a dades alfanumèriques sembla ser força més senzill que el d'OpenLayers mitjançant php, encara que la diferència no es vegi reflectida en els resultats finals.

Cal destacar també que si haguéssim volgut construir un web partint des de zero hauríem trobat a faltar la llibreria GeoExt i els seus controls, els quals ens haurien estalviat la llarga recerca de funcions i ens haurien proporcionat una bona estructura i disseny web. Com esmento anteriorment, el Consorci tampoc necessita un disseny de sèrie com el de Geoext, ja que els programadors web se n'encarreguen i els resultats reeixen, com es fa palès a les imatges, deixant de banda qualsevol debat al respecte. A més a més, aquest format sembla adir-se amb la petició del servei tècnic de garantir frames sectorials, sobretot a l'apartat de serveis.

8. Més enllà de la col·laboració

En paraules del meu tutor a Collserola, Raimon Reventós, les opcions de Google Earth són infinites i tan sols n'hem explotat una petitíssima part. Malauradament la col·laboració s'acaba a mitjan Desembre, però no les idees i els esbossos de noves possibilitats per a la nostra aplicació. A continuació descriu algunes de les idees que no han reeixit i en les que podríem haver seguit treballant en el cas de que la col·laboració s'hagués estès:

1- *Cercar noves llibreries amb un arbrat amb més opcions.*

En aquest sentit llibreries com Dojo Digit permetrien desglossar placemark a placemark cada kml i per tant "volar a" (fly to) cada element vectorial dins el seu subconjunt. Així si ens interessés, amb el kml de restaurants veuríem disgregades les seves parts, així com els noms respectius a cada restaurant de manera que fent clic a l'arbre de Dojo Digit "volaríem" a l'element d'interès d'una manera similar a la que ho fa el Google Earth-programa, però dins un web. Sempre quedaria la grollera solució de carregar un kml per a cada restaurant.

2- *Afegir imatges a l'aplicació*

Altres opcions en les que ens hem quedat a mig camí ha estat amb la possibilitat d'afegir imatges a l'aplicació. Aparentment, mitjançant el programa de Google Earth és relativament senzill afegir una imatge i després guardar-la com a kml o kmz. El Consorci però malgrat marcar-s'ho com a idea mai va creure oportú dedicar-hi més temps més enllà d'aquest primer descobriment. L'Institut Cartogràfic de Catalunya permet al seu web de davallar-nos un mapa topogràfic en format kml que a l'ésser carregat al programa de Google Earth canvia d'escala en funció del zoom. No obstant l'activació i substitució dels diferents topogràfics no era gens àgil, i menys en la nostra aplicació, un cop carregat el kml al projecte web. A més a més, dins l'aplicació, el canvi d'escala o bé era encara més lent o bé no es duia a terme, fet que ens obliga a investigar en aquest sentit per tal de poder carregar un mapa topogràfic en condicions a diferents escales i en kml.

3- *Carregar kml més extensos a Google Maps.*

Fins ara, com ja hem descrit en l'apartat de desenvolupament de l'aplicació, Maps té una limitació de 10MB en els kml i de 3MB en els kmz. Aquesta limitació provoca, entre altres coses, que el consorci opti per Google Earth, fet que fa instal·lar un plug-in a cada visitant del web en el cas que sigui la primera vegada que fan ús de Google Earth, i aquest fet desmotiva a un usuari sense experiència a tancar l'apartat o el web. En alguns casos, Maps tampoc accepta grans kml que no arriben a 10MB. Aquestes limitacions cal estudiar-les per tal de que en un futur també el web corporatiu disposi de nous apartats sectorials amb camins i límit del parc, tots amb un detall important, i per tant, amb forces placemarks. Maps també mostra problemes alhora de carregar les variables de kml programats en variables, i malauradament s'han de programar a mà provisionalment fins que no es trobi una solució a com programar-les.

4- *Trobar noves DLL de software lliure per a transformar shapes a kml.*

També ens quedem a mig camí alhora de trobar una DLL lliure de llicència que corregeixi l'esviaix de coordenades, com ho fa ArcGIS, alhora de transformar a kml. Una opció aparentment senzilla, però que cap programari lliure dels que hem estat esbrinant ha pogut resoldre encara. Tampoc hem arribat a descobrir si hi ha algun sistema de coordenades que aplicat sobre els shapes ja editats en ED55 no ens retorni kml amb problemes de desviament. Alguns fòrums ofereixen opcions com plug-ins que permeten la correcció de desviaments des del mateix Google Earth, tanmateix els mateixos no han reeixit i cal seguir treballant en aquest sentit.

5- *Poder obrir l'aplicació amb tots els navegadors*

L'aplicació finalment ha pogut obrir-se amb Internet Explorer i Mozilla Firefox. I el problema no és d'exigència dins el codi sinó de tolerància de la funció final de l'arbre. En aquest sentit cada explorador necessita una funció final de càrrega a l'arbre diferent. Cal doncs dur a terme un bucle que obri l'aplicació en cas de que l'usuari sigui de Google Chrome, Safari o Opera.

9. Conclusions

Arribats a febrer de 2011, les conclusions són moltes: a nivell de treball, d' institució, el paper de les tecnologies... En general, cal dir que personalment l'experiència ha estat molt positiva, i considero també que ho ha estat pel servei tècnic del parc, que podrà treure'n un bon rendiment de tot plegat en tant que s'han complert en escriure els objectius bàsics que ens havíem proposat.

En primer lloc, a nivell personal i de treball, gairebé tot el què he après m'era una novetat el passat setembre, i m'és una eina útil de cara al futur avui en dia. I ho puc afegir a l'experiència de programació de Visual Basic i OpenLayers/GeoExt, els dos projectes de primer curs de màster que m'havien introduït a aquest complex món. Amb la col·laboració a Collserola obro així una altra nova esclatxa que espero anar eixamplant a mesura que pugui guanyar experiència.

A nivell institucional l'empenta, constància, implicació i polivalència dels treballadors del Consorci trenca qualsevol mite sobre l'eficiència i l'eficàcia del sector públic, i encara més a nivell de SIG, si tenim en compte que sempre han estat pioners en aquest sentit, programant aplicacions per a una millor gestió del Parc o elaborant cartografia fins uns anys enrere inexistent a nivell metropolità.

Treballar sobre els formats Google m'ha resultat una experiència innovadora en tots els sentits. No obstant cal destriar el gra de la palla i concloure que no és el mateix programar una funció que trobar un exemple a internet, gaudir d'una llibreria que haver de programar requeriment per requeriment, de la mateixa manera que no és el mateix programar consultes php que obtenir dades alfanumèriques d'un kml. Així, el nostre projecte ha combinat una cosa i l'altre, avaluant-ne peròs i contres i podent avançar amb més o menys velocitat en funció de la fase que ens proposàvem i la seva dificultat.

Amb en Raimon i companyia també he après que tot és possible en tant que hom s'hi posa, però que calen dies, hores, setmanes o a vegades mesos per tal d'assolir un objectiu. Poder arribar a un punt per pensar-ne dos més a treballar, sabent que són possibles si hom no s'arronsa. Aquesta també ha estat la meva altra gran formació; tot és a l'abast en tant que hom s'ho proposa.

A nivell de resultats del projecte també cal destacar l'espectacularitat i la importància emergent de les tecnologies GIS web dinàmiques i com aquestes s'adapten perfectament en la modernització i reestructuració dels webs, en aquest cas el projecte de web corporatiu. Així, s'ha aconseguit programar sobre informació flexible, i adaptar-

la al web mitjançant la personalització respectiva sobre les API, amb la integració total de frames o marcs amb informació geogràfica principal o complementària. D'aquesta manera, la web esdevindrà, a curt termini, un punt de trobada, transversal per a les administracions, i col·laboratiu pel que fa als individus que les componen des de qualsevol lloc de l'àrea metropolitana de Barcelona i del món. Així es fa palès com el Consorci intenta agafar-se als avenços de la geoinformàtica, lògicament, en la mesura del que és possible.

En definitiva és un orgull poder arribar a exposar els resultats davant dels companys dient-los-hi que l'experiència ha estat profitosa, que cal treballar de valent però que si ho fem tard o d'hora ens en sortirem, i que com a geògrafs no ens hem equivocat amb el camí que hem emprès.

10. Bibliografia i webgrafia

1- Bibliografia

BUZAI, Gustavo Daniel. Geografia Glob@l: El paradigma geotecnològic i el espai interdisciplinari en la interpretació del món del segle XXI, - 1 ed 1 reimpr. – Buenos Aires: Lugar Editorial, 2004.

2- Webgrafia: Obertes i en correcte funcionament fins a principis de febrer de 2011

Fonts d'apartats de Google Earth i KML:

<http://thematicmapping.org/>

<http://wholemap.com/>

<http://earth.google.com/>

<http://www.desarrolloweb.com/>

<http://code.google.com/>

<http://freegeographytools.com/>

Fonts de Google Maps:

<http://code.google.com/>

<http://econym.org.uk/>

<http://www.slideshare.net/>

<http://www.cbiwindstorm.com/>

<http://www.costaricabygps.com/>

<http://gmaps-samples.googlecode.com/>

Blogs i fòrums de discussió en xarxa: Funcions i altres

<http://forums.asp.net>

<http://www.forosdelweb.com/>

<http://foro.gabrielortiz.com/>

<http://riteshk.blogspot.com/>

<http://stackoverflow.com/>

<http://trixmontero.blogspot.com>

<http://vbcity.com>

Collserola

www.parccollserola.net/