

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

SCUOLA DI SCIENZE
Corso di Laurea in Informatica per il Management

**LA VISUALIZZAZIONE DI OPEN DATA
GEOGRAFICI:
WEB SHAPEFILE VIEWER**

Tesi di Laurea in Informatica Avanzata

**Relatore:
Chiar.mo Prof.
Fabio Vitali**

**Presentata da:
Michele Valente**

**Sessione II
Anno Accademico 2012–2013**

Indice

1	Introduzione	1
2	Accesso agli Shapefile in Open Data	9
2.1	Nascita e Importanza degli Open Data	9
2.2	Cosa sono gli Shapefile	14
2.3	Servizi WebGis	16
2.3.1	Shapefile-js	18
2.3.2	LsiViewer	19
2.3.3	Indiemapper	21
2.3.4	ArcGIS	21
2.4	Nuovi strumenti	21
3	Web ShapeFile Viewer.	23
3.1	Come funziona Web ShapeFile Viewer.	23
3.2	Casi d'uso.	29
3.3	La funzione del Web ShapeFile Viewer.	32
4	Descrizione della architettura del Web ShapeFile Viewer.	35
4.1	Composizione dell'architettura software.	35
4.2	Ruolo e struttura delle funzioni rilevanti	37
4.3	Tecnologie Utilizzate.	42
4.4	Informazioni per l'eventuale modifica da parte di terzi.	45
4.5	Difficoltà incontrate durante la progettazione.	47

5	Valutazione funzionale del Web ShapeFile Viewer	49
5.1	Efficienza e non efficienza	49
5.2	Pregi e difetti	54
6	Conclusioni	57
	Bibliografia	59

Elenco delle figure

2.1	Diffusione Open Data	10
2.2	L'utilizzo delle Licenze più Importanti	12
3.1	Pagina iniziale del Web ShapeFile Viewer	25
3.2	Dimostrazione delle Tab Bloccate	25
3.3	Struttura di un Shapefile	25
3.4	Finestra di Scelta Shapefile	26
3.5	Finestra di Successo	27
3.6	Tab Attive	27
3.7	Sviluppo OpenData Italia	32
4.1	Processo Software	36
5.1	Tempi di Upload e Visualizzazione	50
5.2	Tempi per la Visualizzazione	50

Capitolo 1

Introduzione

Lo scopo di questo progetto è realizzare un WebGIS, ovvero un servizio online basato su tecnologie web che permetta la visualizzazione di open data, contenenti dati geografici in formato ShapeFile, che la maggior parte delle nazioni, delle regioni e dei comuni rendono disponibili alla popolazione tramite internet.

Il servizio proposto è incentrato ed è stato testato maggiormente su ShapeFile contenenti informazioni riferite al territorio italiano, perché, come questa tesi vuole dimostrare, una delle caratteristiche di questo formato, e che i dati contenuti si diversificano considerevolmente in base alla geolocalizzazione del luogo a cui fanno riferimento.

Questa tesi è sostenuta dalla fruizione di un servizio che si presenta con un'interfaccia grafica semplice ed intuitiva e che garantisce all'utente inesperto, con limitate conoscenze in ambito informatico, di usufruire dell'operazione di upload dei file presi da siti che forniscono open data geografici, permettendone così una visualizzazione completa ed utile mediante la sovrapposizione delle informazioni da essi estrapolate, alle mappe geografiche fornite da Google e da OpenStreetMap.

La necessità di avere un servizio semplice ed efficace, per un principiante come per un utente esperto, pone le basi per lo sviluppo di questo servizio, la cui efficacia si concretizza nella possibilità di fare l'upload dei dati così

come ci vengono forniti, senza effettuare modifiche intermedie per garantire la compatibilità con il servizio. Inoltre questo sistema offre una disponibilità costante, essendo “a portata di mouse” sul web e quindi accessibile da ogni luogo e in ogni momento, senza avere l’obbligo di installare alcuna applicazione desktop.

L’utilità del WebGis, deriva dalle opportunità che offre il servizio, ovvero l’assenza di limitazioni di dimensione di upload dei dati, che permette di utilizzare ShapeFile di grandi dimensioni o con molti elementi all’interno, e la possibilità di rendere visibili gli attributi ad essi associati, se presenti, riguardanti un singolo elemento del file, dando la possibilità di conoscere in modo immediato le informazioni ricercate.

Il servizio di pubblicazione, con la possibilità di download di open data e quindi di dati geografici in formato ShapeFile, è offerto dai comuni e dalle regioni in modo efficiente ed è spesso accompagnato da un servizio separato che ne offre anche la rappresentazione, ma con la necessità di effettuare nuovamente una accurata ricerca.

Quindi, in questa tesi vengono messe in risalto le difficoltà riscontrate dagli utenti, relative alla visualizzazione dei dati appena scaricati dai siti che offrono open data, in quanto l’operazione di rappresentazione grafica risulta essere separata ed incoerente con i dati che l’utente ha precedentemente scaricato.

Sul web inoltre sono presenti diversi progetti dello stesso tipo di quello che questa tesi vuole proporre, che però presentano alcuni malfunzionamenti o sono ancora in via di sviluppo; in aggiunta è presente una moltitudine di applicazioni scaricabili ed installabili sul proprio personal computer che offrono potenzialità maggiori ma non effettivamente immediate e soprattutto semplici. Tutto ciò non è d’aiuto per l’utente preso in esame, ovvero colui che dopo aver trovato lo shapeFile contenente per esempio i dati riferiti ai marciapiedi di Bologna e dopo averli scaricati sul proprio pc non vede solo l’ora di visualizzarli per capire effettivamente se l’integrità dei dati da lui prescelti soddisfa le proprie necessità.

Per avvalorare questa tesi, viene effettuato uno studio approfondito degli elementi che permettono la creazione di un servizio capace di soddisfare i bisogni degli utenti; vengono inoltre presi in esame i casi in cui si è provato a creare un sistema di questo tipo, ma con dei risultati non soddisfacenti. Per poter affrontare e risolvere le difficoltà riscontrate dagli utenti, bisogna in primo luogo avere una descrizione chiara degli argomenti da cui prende spunto questa tesi, ovvero gli open data ed in particolare gli shapefile per la creazione di sistemi gis.

Gli open data (dati aperti) sono dei dati il cui copyright, eventualmente, si limita ad obbligare di citarne la fonte e nel caso la redistribuzione delle modifiche. I movimenti che hanno contribuito alla diffusione di questi dati sono, negli USA la direttiva sull'open government del 2009, ed in Italia la direttiva 2003/98/CE e il progetto sviluppato da OpenStreetMap citate nei capitoli successivi. La mancanza però di una legislazione che imponga alle pubbliche amministrazioni di diffondere i propri dati, definisce il chiaro fallimento di tali direttive; infatti le pubbliche amministrazioni come quelle private, hanno la tendenza a mantenere i propri dati nascosti, non garantendo così la partecipazione alle scelte amministrative da parte del pubblico cittadino.

Il formato Shapefile sviluppato nel 1990 da Esri per regolare l'interoperabilità con i sistemi GIS, consiste in un insieme di file con estensioni diverse, ma con la condivisione dello stesso nome; il contenuto è caratterizzato da elementi geometrici primitivi: punti, linee, poligoni e testi, cui viene attribuito il nome di "features", le quali vengono arricchite da informazioni aggiuntive chiamate "attributi". Una particolare caratteristica di questo formato è la presenza del file con estensione ".prj" che definisce la proiezione ovvero la trasformazione delle coordinate di un luogo in un piano.

Il GIS (acronimo di Geographic Information System, la cui nascita risale al 1963), è un insieme complesso di componenti software per l'acquisizione, l'immagazzinamento, l'analisi, la visualizzazione e la restituzione di informazioni derivanti da dati geografici. Un webGIS si distingue da un progetto GIS per le specifiche finalità di comunicazione e di condivisione delle informazioni con

altri utenti: in pratica si implementa un software GIS su di un web server, che consente l'interazione attraverso internet con la cartografia e con i dati ad essa associati. Alcuni esempi noti di webGIS sono le applicazioni web che gestiscono la localizzazione cartografica, gli stradari oppure gli atlanti online. Diverse applicazioni e progetti presenti sul web sono stati testati per verificarne la funzionalità e la stabilità dei requisiti necessari per soddisfare gli utenti presi in esame. Il risultato della ricerca di applicazioni che soddisfacessero i bisogni degli utenti descritti in precedenza, ha messo in risalto 4 applicazioni già presenti sul web con le caratteristiche richieste:

1. Shapefile-js:

Questa applicazione adopera un nuovo sistema per leggere ed accedere ai file presenti sul pc del utente del servizio, la cui velocità di processo delle features è impressionante, come viene citato nei capitoli successivi; ma presenta delle difficoltà nella trasformazione delle proiezioni di input in quella italiana; inoltre non presenta una sovrapposizione dei dati su di una mappa geografica come Openstreet o Google, fornendo così un'informazione incompleta.

2. LsiViewer:

Il progetto con le caratteristiche più simili a quelle ricercate ha mostrato, in seguito al testing, dei risultati scadenti, ovvero una limitazione sostanziale del numero di features accettate in input, ed inoltre uno sviluppo instabile delle funzioni di visualizzazione e gestione dei dati.

3. Indiemapper:

Questa applicazione dalle specifiche molto avanzate, dai testing, risulta la più funzionale, ma non offre un supporto completo per tutte proiezioni esistenti.

4. Online ArcGis:

Come si è citato nei capitoli successivi, questa applicazione offre un supporto per i file con un numero minore o uguale a 1000 features, limitandone considerevolmente l'utilizzo. La soluzione proposta dagli sviluppatori è la generalizzazione delle features che spesso genera imprecisioni nella visualizzazione.

Un'integrazione degli strumenti disponibili in rete quali le applicazioni desktop: ogr2ogr e proj4, e le librerie: openlayers, api google e geoExt2, permettono una risoluzione dei problemi sopracitati, garantendo la creazione di un software webGIS stabile ed efficiente.

Il Web ShapeFile Viewer è nato con lo scopo di soddisfare il bisogno degli utenti in difficoltà nel visualizzare i propri open data. È un'applicazione disponibile sul web che permette agli utenti di visualizzare, all'interno di Google Map e OpenstreetMap, i dati geografici contenuti in Shapefile.

La composizione di questo progetto e il suo funzionamento possono essere descritti brevemente nei seguenti punti:

1. Caricamento file:

L'applicazione permette esclusivamente la selezione e il caricamento di file con estensione “.zip”. Questa scelta è stata effettuata perché è stata riscontrata, nei siti open data, la diffusione degli Shapefile compressi in file con questa estensione. In questo modo si permette all'utente di effettuare l'upload del file così come è stato scaricato dal web.

2. Visualizzazione sulle mappe:

La visualizzazione dei dati caricati avviene sulle mappe OpenstreetMap e Google Map; inoltre viene mostrato in formato SVG su di una pagina html bianca, in modo da mostrare le informazioni di base dello Shapefile. Le prime due mappe sono il risultato della scelta dello sviluppatore di utilizzare un servizio dalle grandi funzionalità come google map e l'utilizzo di un servizio open source quale OpenStreetMap.

3. Attributi by Click:

L'applicazione, in aggiunta, offre la funzionalità che permette di visualizzare le informazioni riguardanti una sola feature cliccandoci semplicemente sopra. In questo modo si potranno avere sempre a portata di click gli attributi dello Shapefile caricato.

Il progetto Web ShapeFile Viewer presenta caratteristiche che lo rendono paragonabile o migliore delle applicazioni sopracitate e anche piccoli difetti che però non interferiscono con il funzionamento complessivo del sistema. I pregi di questo sistema sono:

- La possibilità di visualizzare Shapefile contenenti dati del tipo, linea, punto, poligono e testo, con qualsiasi tipo di proiezione predefinita.
- La possibilità di visualizzare Shapefile con un numero di features di media lunga grandezza;
- La possibilità di usufruire della funzione per visualizzare gli attributi della singola feature;
- La possibilità di visualizzare lo Shapefile in diverse mappe con differenti livelli di zoom.
- La sicurezza che i dati caricati non vengano conservati evitando così una fuga di informazioni.
- La sicurezza di un supporto completo ed ottimale con i diversi browser esistenti.

I difetti principali sono, nel caso di Google Map, un problema di sfarfallio dei dati caricati in caso di spostamento della mappa. Inoltre, la necessità di dover caricare il file sul server per effettuare le conversioni necessarie, spesso può aumentare i tempi di processo. Temporaneamente è stata gestito solo il caricamento dei file di tipo zip, escludendo altri formati. Il primo avvio da parte del client, non avendo nessun dato salvato in “cache”, può risultare lento.

La valutazione di questo progetto riporta dei risultati positivi, prendendo in considerazione i test che verranno descritti con più precisione nel capitolo 5, in cui è dimostrato un funzionamento del sistema con tutti i file testati, con associate le diverse proiezioni e grandezze.

In conclusione il sistema sviluppato ha le funzionalità e la stabilità adatte per risolvere le difficoltà presentate dagli utenti, consentendo così una visualizzazione impeccabile dei dati caricati, e fornendo una navigazione nel sito semplice e veloce. Per gli sviluppi futuri è in programma il rilascio di una nuova versione che consenta l'esportazione dei dati in altri formati o in immagine, con il supporto al caricamento di altri formati contenenti informazioni geografiche; inoltre si tenterà di rendere il sistema più veloce per caricamento dei file sul server.

Capitolo 2

Accesso agli Shapefile in Open Data

2.1 Nascita e Importanza degli Open Data

Gli open data come, si può leggere sul sito Dati Open.it [1], sono dei dati privi di brevetto e pubblicamente accessibili il cui copyright, eventualmente, si limita ad obbligare di citarne la fonte e di ridistribuirne le eventuali modifiche. Gli open data derivano dall’open government, che come descrive Fabio Deotto in “daily.wired.it” [2], è una disciplina più ampia basata su modelli, strumenti e tecnologie che consentono alle amministrazioni di essere aperte e trasparenti nei confronti dei cittadini, permettendo una maggiore partecipazione ai processi decisionali, offrendo l’opportunità d’accesso ai dati e alle informazioni, consentendo una analisi approfondita dei processi amministrativi e delle decisioni territoriali. Tuttavia, come afferma anche Ernesto Belisario nel suo blog [3], la mancanza di una legislazione chiara che definisca degli standard europei e mondiali limita le possibili applicazioni degli open data; di conseguenza le aziende e le amministrazioni continuano a tenere il proprio patrimonio informativo fuori portata del pubblico cittadino. I maggiori avvenimenti e sviluppi riguardanti il movimento open data si possono riassumere in una scala temporale di circa 10 anni, come si può leggere sul sito

“dati.emilia-romagna.it” [4]. Nel novembre 2003 viene approvata la direttiva 2003/98/CE da parte del parlamento e del consiglio europeo, tale direttiva, verrà poi considerata il primo passo in tema di riutilizzo dell’informazione del settore pubblico.

Nel 2009 alla conferenza denominata TED (Technology Entertainment Design) [5], l’inventore del World Wide Web, Tim Berners-Lee, con le parole “Raw data now”, incoraggia il pubblico e la popolazione a pretendere il rilascio dei dati che le pubbliche amministrazioni raccolgono per svolgere le loro funzioni. Nonostante ciò, negli anni successivi, in Europa, non si vide una grande divulgazione di dati grezzi e aggregati da parte delle aziende pubbliche e private. Un gran numero di amministrazioni pubbliche nazionali, regionali e comunali, hanno creato i propri servizi web, divulgando così i dati da loro in possesso, solo alla fine del 2011.

Il presidente Barack Obama ha sostenuto il movimento open data negli, Stati Uniti, attraverso la direttiva sull’open government del 2009 [6], nella quale si legge testualmente:

«Fin dove possibile e sottostando alle sole restrizioni valide, le agenzie devono pubblicare le informazioni online utilizzando un formato aperto (open) che possa cioè essere recuperato, soggetto ad azioni di download, indicizzato e ricercato attraverso le applicazioni di ricerca web più comunemente utilizzate. Per formato open si intende un formato indipendente rispetto alla piattaforma, leggibile dall’elaboratore e reso disponibile al pubblico senza che sia impedito il riuso dell’informazione veicolata.»

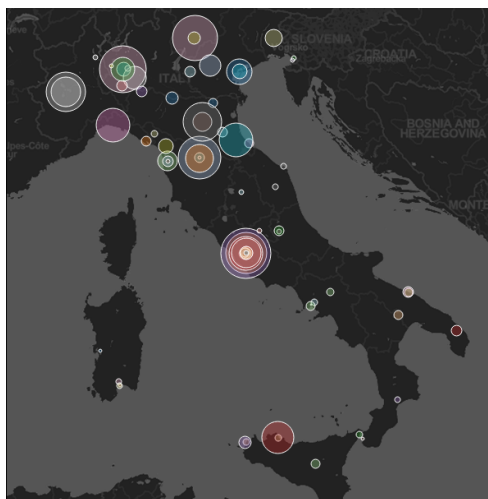


Figura 2.1: Diffusione Open Data.

Questa direttiva si concretizzò nello sviluppo del sito web Data.gov¹, che oggi ingloba tutte i dati aperti resi disponibili dalle amministrazioni e dagli enti statunitensi. La figura 2.1² rappresenta i maggiori centri di diffusione italiani di open data, in cui si può notare un'affluenza e una disponibilità maggiore di dati nelle regioni nordiche di questa nazione. In Italia, come si legge nel sito [7] della "GFOSS"³, si è diffuso il movimento riguardante i dati aperti solo nel 2007, grazie al progetto OpenStreetMap, che si poneva il fine di creare uno stradario utilizzabile liberalmente. Tra il 2007 e il 2010, molti comuni e regioni italiane hanno pubblicato, con licenza aperta, i propri stradari, insieme ad altri dati da loro posseduti. Successivamente, il 18 ottobre 2011, come si può leggere sul proprio sito [8], viene messo online il sito dati.gov.it, sul modello del servizio online statunitense. Gli avvenimenti più recenti riguardano la diffusione di nuove licenze che regolano l'utilizzo degli open data; come dimostra parzialmente la figura 2.2⁴, che rappresenta le licenze utilizzate maggiormente, dalle amministrazioni italiane, le cui più importanti sono:

- Creative Commons (CC0):

Usando CC0, si può rinunciare a tutti i diritti d'autore e diritti connessi o simili che si detiene sulla propria opera, quali i diritti morali (per quanto rinunciabili), i diritti all'immagine o alla riservatezza, diritti che proteggono contro la concorrenza sleale, e diritti sulle banche di dati che limitino l'estrazione, la disseminazione ed il riuso dei dati.

- Creative Commons (CC-by):

Usando questa licenza si è liberi, alla condizione di attribuire la paternità nei modi definiti dall'autore, di riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire e recitare, di modificare e di usare l'opera in questione per fini commerciali.

¹Url: <http://data.gov.com/>

²Url: <http://www.dati.gov.it/content/infografica>

³GFOSS: Associazione italiana per l'informazione geografica libera

⁴Url: <http://www.dati.gov.it/content/infografica>

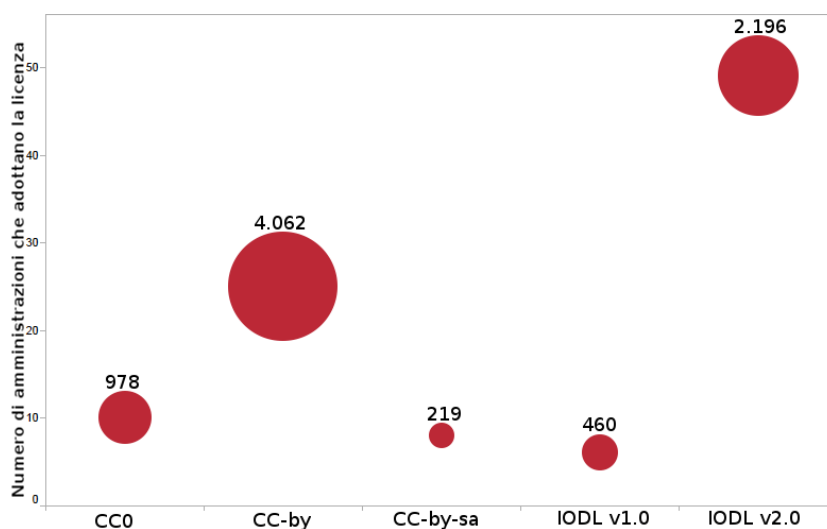


Figura 2.2: Licenze Utilizzate dalle Amministrazioni Italiane, i cui cerchi rappresentano il numero di dataset rilasciati.

- Creative Commons (CC-by-sa):
Usando questa licenza si è liberi, alle condizioni di attribuire la paternità nei modi definiti dall'autore e di ridistribuirne le modifiche con la stessa licenze, di riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire e recitare, di modificare e di usare l'opera in questione per fini commerciali.
- Italian Open Data Licenses V1.0 (IODL):
Usando questa licenze si può riprodurre, distribuire al pubblico, concedere in locazione, presentare e dimostrare in pubblico, comunicare al pubblico, messa a disposizione del pubblico inclusa, trasmettere e ritrasmettere in qualunque modo, eseguire, recitare, rappresentare, includere in opere collettive e/o composte pubblicare, estrarre e reimpiegare le Informazioni con le condizioni di indicare la fonte delle Informazioni e il nome del Licenziante, includendo, se possibile, una copia di questa licenza o un collegamento link; non riutilizzare le Informazioni in un modo che suggerisca che abbiano carattere di ufficialità o che il Licen-

ziente approvi l'uso che fai delle Informazioni; prendere ogni misura ragionevole affinché gli usi innanzi consentiti non traggano in inganno altri soggetti e le Informazioni medesime non vengano travisate.

- Italian Open Data Licenses V2.0 (IODL):
Questa licenza è la versione aggiornata di quella precedente, i diritti contenuti sono gli stessi ma viene aggiunta la condizione di pubblicare e condividere gli eventuali Lavori derivati con la stessa licenza o con una Licenza compatibile.

L'importanza di garantire una immediata diffusione dei dati aperti risiede nel utilizzare raw data (dati nudi e crudi) di aziende pubbliche ed eventualmente private, per offrire l'opportunità di creare nuovi tipi di informazioni. I raw data, spesso, non garantiscono un'informazione utile e chiara, ma, una volta aggregati e visualizzati come grafici o in siti web in modo utile, possono offrire un'informazione immediata e intuitiva. In questo caso, la domanda che ci si pone è perché diffondere raw data e non direttamente dati aggregati, grafici o informazioni utili? La risposta a questo quesito la offre Tim Berners-Lee al TED del 2009, con le seguenti parole tradotte dalla lingua inglese⁵

« Se voi stessi siete a conoscenza o siete responsabili di dati di un determinato dipartimento governativo, spesso scoprite che queste persone, sono molto tentate a tenersi stretti, Hans Rosling lo chiama "Tenersi stretto il database". Si tiene stretto il proprio database finché non si è fatto un bel sito web per l'accesso. Beh, io vorrei suggerire invece: sì, fate un bel sito web, chi sono io per dirvi di non fare un bel sito web? Fate un bel sito, ma prima dateci i dati non adulterati, vogliamo i dati.»

La popolazione dovrebbe poter sfruttare al massimo i raw data per fare statistiche, grafici e siti, invece di avere dati già aggregati, filtrati o non del tutto soddisfacenti, e quindi inutilizzabili.

⁵Tradotto da Andrea Fare, url:

http://www.ted.com/talks/lang/it/tim_bern timers_lee_on_the_next_web.html

2.2 Cosa sono gli Shapefile

Il formato Shapefile è un popolare formato vettoriale per sistemi informativi geografici. Questo formato è stato sviluppato e regolato da ESRI, per migliorare l'interoperabilità fra i sistemi GIS; di fatto, è diventato un standard per lo sviluppo di sistemi GIS di ogni tipo. Shapefile, come si può leggere nella descrizione tecnica [9] messa a disposizione dalla ESRI, consiste in un insieme di file con la condivisione dello stesso nome, ma con estensioni diverse, di cui alcune obbligatorie (.shp, .dbf, .shx) e altre 9 opzionali che conservano indici e dati accessori.

In questo progetto si è prestata una particolare attenzione al file opzionale con estensione “.prj” che contiene la proiezione, cioè le informazioni che specificano il sistema di coordinate geografiche dei dati geometrici contenuti nel file “.shp”. La proiezione, come è stata definita da Scott Hutchinson [10], è una trasformazione sistematica della latitudine e della longitudine di un luogo in un piano.

Le proiezioni sono necessarie per la costruzione delle mappe, ma spesso mostrano delle distorsioni in base alla mappa a cui si riferiscono; la correttezza della proiezione dipende dalla minore visibilità di queste distorsioni. Tra le tante proiezioni difficilmente è individuabile la “migliore”, perché è necessario considerare diversi fattori come il luogo geografico d'interesse e la scala che si vuole utilizzare. La proiezione più diffusa e supportata nei sistemi GIS è la WGS84 (Transverse Mercator) che utilizza una proiezione in larga scala. Le mappe utilizzate dai principali servizi di mappe, come Google Map e OpenStreetMap, usano la stessa proiezione ovvero EPSG:90013, molto spesso differente da quella utilizzata negli Shapefile.

Generalmente si tende a considerare il file con estensione “.shp” uno Shapefile completo; tuttavia, come precedentemente sottolineato, gli elementi obbligatori che lo compongono sono il file con estensione “.dbf” e quello con estensione “.shx”, senza i quali lo Shapefile sarebbe incompleto. È inoltre definito un formato vettoriale di registrazione di identità geometriche e dei dati a loro associate, senza il supporto per la registrazione di informazioni

topologiche. Questo formato è caratterizzato soltanto da elementi geometrici “primitivi”: punti, linee, poligoni e testi, a cui viene attribuito il nome di “features”, sono inoltre arricchiti da informazioni aggiuntive chiamate “attributi”, che specificano cosa questi elementi rappresentino.

Ogni file, per essere compatibile con la maggior parte delle applicazioni, è conforme alla convenzione per i nomi MS-DOS 8.3, in cui viene definito che il nome del file deve essere composto da 8 caratteri per il prefisso, un punto e tre caratteri per l'estensione. Tutti i file di uno Shapefile devono essere collocati nella stessa cartella. I file obbligatori:

- .shp il file che conserva le geometrie.
- .shx il file che conserva l'indice delle geometrie.
- .dbf il database degli attributi.

Gli Shapefile non possono contenere informazioni topologiche (adiacenza, connessione, prossimità, coincidenza) sulle loro feature; essi sono quindi la mera rappresentazione spaziale delle feature e degli attributi.

Quando due linee si incontrano formano un angolo, tali angoli sono definiti da un punto, perciò non sono possibili angoli arrotondati se non aggiungendo un gran numero di punti. Di fatto questo formato non supporta le curve di Bèzier.

Lo sviluppo del formato Shapefile, prende spunto da una branca della matematica chiamata oggi topologia, fondata dal matematico Leonhard Euler nel 1736. Molto più recentemente, nel 1970, il United States Censes Bureau, nel predisporre il censimento, fece uso di concetti topologici per mappare e ridurre gli errori dei dati del censimento. In seguito a questo utilizzo della topologia matematica, nel 1990, venne introdotto il formato Shapefile con la release di ArcView2. Il vantaggio primario di questo formato è caratterizzato dalla semplice struttura del file che permette una creazione più veloce delle features.

2.3 Servizi WebGis

GIS è l'acronimo di Geographic Information System che consiste in un insieme complesso di componenti hardware e software per l'acquisizione, l'immagazzinamento, l'analisi, la visualizzazione e la restituzione di informazioni derivanti da dati geografici. Secondo la definizione di Burrough (1986) [11] *«il GIS è composto da una serie di strumenti software per acquisire, memorizzare, estrarre, trasformare e visualizzare dati spaziali dal mondo reale»* I GIS sono l'intersezione di diverse discipline: cartografia, informatica e statistica, attraverso l'unione di queste materie si riescono a definire gli elementi base che li compongono:

- **Elementi Cartografici:**

Sono le informazioni che permettono la georeferenziazione delle mappe e quindi la corrispondenza fra gli elementi grafici e la loro reale posizione nello spazio, secondo un riferimento di tipo geografico. A questa categoria appartengono anche le rilevazioni geografiche e cartografiche precedenti alla digitalizzazione.

- **Elementi Grafici:**

Sono determinati dalla combinazione di sei primitive geometriche: il punto, la linea (segmento), l'area (poligono), il pixel, il simbolo e le annotazioni geografiche (etichette).

- **Database degli Attributi:**

Gli attributi sono tutte le informazioni utili alla rappresentazione degli elementi grafici, geografici e relativi alle loro caratteristiche.

La storia dello sviluppo dei sistemi GIS si può riassumere come segue: [12]

- **1963** - Nasce CGIS(Canada GIS), progetto che nasce con lo scopo di analizzare i dati raccolti dal Canada Land Inventory.
- **1974** - Si tiene a Reston (Virginia) AUTOCARTO, la prima conferenza internazionale sulla cartografia computerizzata.

- **1978** - ESRI sviluppa la prima versione di Arc/Info, primo software Gis.
- **1992** - ARC/INFO viene dotata di interfaccia grafica per rendere più facile l'utilizzo.
- **1993** - Nasce PARC il primo visualizzatore di mappe interattive basato sul web.
- **1994** - Viene fondato l' Open Gis Consortium con lo scopo di diffondere le tecniche di geoprocessing.
- **1995-2000** - Inizio della diffusione dei primi strumenti con alte funzionalità gis.
- **2000-2010** - Viene implementata e sviluppata la visualizzazione in 3D dei dati.

Un WebGis di, conseguenza, è un sistema informativo geografico (GIS) pubblicato sul web. Un progetto WebGis si distingue da un progetto GIS per le specifiche finalità di comunicazione e di condivisione delle informazioni con altri utenti; in pratica si implementa un software GIS su di un web server, anche detto map server, che consente l'interazione attraverso internet con la cartografia e con i dati ad essa associati.

Alcuni esempi noti di webGis sono le applicazioni web che gestiscono la localizzazione cartografica, gli stradari oppure gli atlanti online.

Sul web è presente una moltitudine di webGis, differenziati dalle scelte effettuate dallo sviluppatore durante la progettazione. Le differenze si possono notare facilmente prendendo in considerazione i webGis open-source; le diversità, emerse da un'accurata ricerca, sono:

- **Software e librerie:**

Scelte ed utilizzate in base ai parametri di gradimento dello sviluppatore, quali velocità di processing dei dati, riusabilità e manutenzione del codice, nuove tecnologie software e tutto ciò che potrebbe influire

sulla scelta della tecnologia da utilizzare e sulla velocità del sistema, come gli effetti grafici di alto livello.

- **Tecnologie serverside:**

Decisioni che portano lo sviluppatore a scegliere tra un semplice web server, un map-server con database o un geo-server.

- **Tecnologie per il data mining:**

L'utilizzo o meno di database per la gestione dei dati georeferenziali e degli attributi ad essi associati; spesso la scelta ricade sull'utilizzo di file appropriati per la conservazione di questi dati, come: Shapefile, geoJson, kml e gml.

- **Funzionalità:**

lo sviluppatore sceglie quali funzionalità implementare, tra le quali la possibilità di modificare le singole features o interi vettori e di conseguenza gli attributi, di esportare le mappe ed altre tante funzionalità che i software GIS stand-alone solitamente offrono.

Diversi progetti presenti sul web sono stati testati per verificarne le funzionalità e i sistemi utilizzati, e confrontarli infine con le potenzialità del Web Shapefile Viewer.

2.3.1 Shapefile-js

La libreria Shapefile-js, il cui autore è Tom Carden, è stata testata perché adopera un nuovo sistema per leggere ed accedere ai file presenti sul pc del client del servizio; perciò è stata varata la possibilità di un'integrazione con questo progetto, perché come affermano gli utenti che l'hanno testata ha una velocità impressionante di processo delle features. Gli utenti del sito Pro-Develop commentano così: [13] « *L'ho provato con uno Shapefile di 88 MB con 37000 fetures, è molto rapido, Ottimo Lavoro*», « *La velocità è impressionante! Con uno Shapefile di 6 MB con all'interno dei poligoni*»Dopo un accurato testing e diverse prove è stata riscontrata un'incompletezza delle

funzionalità ed una conseguente instabilità del sistema. I punti a favore di questa libreria sono l'utilizzo delle api di Html5, che consentono un accesso veloce ai file locali ed un rendering delle features in un elemento canvas.

D'altro canto, i punti a sfavore risultanti dai test sono: la difficoltà nella trasformazione della proiezione delle coordinate geografiche lette, prendendo in considerazione gli Shapefile con proiezione EPSG:23032 (sistema di riferimento incentrato sulla nazione italiana) a cui si aggiunge la mancata sovrapposizione del vettore su di un servizio di mappe come Google Maps e OpenStreetMap con le funzionalità ad esse associate; inoltre è stato difficile applicare a questa libreria la funzione di zoom a causa di un ritardo eccessivo e molto fastidioso nel ridimensionamento del vettore.

I servizi che offrono opendata pubblicano i propri Shapefile compressi in file zip per garantire un download completo, in questo caso la libreria presa in esame impone un'estrazione dei file di cui utilizza solamente l'estensioni .dbf e .shp, che non garantiscono una completo e corretto rendering dello Shapefile.

2.3.2 LsiViewer

Sviluppato dal laboratorio Spatial Informatics, IIIT Hyderabad, è un webGis che ha delle caratteristiche molto simili al progetto che questa tesi vuole descrivere, infatti uno degli sviluppatori nel proprio blog scrive [14] «*LsiViewer è stato sviluppato per gli utenti che non voglio accedere a una suite GIS ma vogliono semplicemente visualizza e stampare i dati che hanno. Inoltre LsiViewer non dipende da MapServer o GeoServer*» Queste caratteristiche sono le stesse su cui si basa il progetto Web ShapeFile Viewer, ma differentemente, sono sviluppate in modo da far funzionare il progetto con una maggioranza di dati presenti sul web. Dopo un accurato testing del LsiViewer si sono verificati i seguenti problemi:

- **Prima Prova:**

Upload dei marciapiedi⁶ di Bologna con circa 13000 features di tipo Linea, il software è andato in crash ed è stato necessario effettuare il refresh della pagina.

- **Seconda Prova:**

Upload delle siepi⁷ di Bologna con un numero di features poco minore con geometrie di tipo poligono il risultato è stato lo stesso, il sistema si è bloccato.

- **Terza Prova:**

Contatto diretto tramite mail con uno degli sviluppatori chiedendo informazioni riguardo i problemi riscontrati, con la specifica domanda: “LsiViewer è in grado di caricare uno Shapefile con 13000 features?” La risposta dello Sviluppatore ha caratterizzato la differenza maggiore con il progetto Web ShapeFile Viewer:

«13000 features è un numero troppo grande per il sistema LsiViewer.»

- **Quarta Prova:**

Upload dei marciapiedi di Bologna con la riduzione a 100 delle features. Il sistema risponde bene a questo input e riesce a caricare e a visualizzare i dati su di una pagina html bianca, togliendo però la possibilità di effettuare un confronto e una localizzazione delle features su di una mappa .

Dopo aver effettuato i test è stato possibile verificare le caratteristiche del LsiViewer che pur restando simili nella progettazione risultano incomplete dagli sviluppatori nel concreto.

⁶Marciapiedi Bologna url:

<http://dati.comune.bologna.it/node/186>

⁷Siepi Bologna url:

<http://dati.comune.bologna.it/node/215>

2.3.3 Indiemapper

Un webGis dalle specifiche molto avanzate e con molte funzionalità offerte dagli sviluppatori. Dopo averlo testato, però, si è verificato un problema che ha reso spesso inutilizzabile questo webGis. Conseguentemente all'upload dei file scaricati da diversi siti italiani che offrono open data di tipo Shapefile, il sistema si blocca e avvisa l'utente che Indiemapper non supporta il tipo di proiezione adottata dal file in upload. Quindi questo sistema circoscrive la sua utilità alla necessità di un file con la proiezione supportata o di una operazione intermedia di conversione, tramite un sistema Gis installato sul proprio personal computer.

2.3.4 ArcGIS

All'interno della guida di utilizzo di arcGis online afferma che: «*Se uno Shapefile contiene una quantità di dati ridotta, in genere meno di 1.000 feature, è possibile renderlo visibile agli altri utenti mediante un browser Web aggiungendolo come file .zip, contenente i file .shp, .shx, .dbf e .prj, a una mappa creata con ArcGIS.com Map Viewer.*» La soluzione che offre a questa limitazione è la generalizzazione delle features che spesso genera delle imprecisioni nella visualizzazione.

2.4 Nuovi strumenti

Nella progettazione del sistema ShapeFile Viewer sono state sperimentare diverse librerie Open Source e algoritmi, per ovviare ai problemi di cui si è parlato nel paragrafo 2.3. In seguito ad un lungo testing, riscontrando problematiche di conversione, di riproiezione, di visualizzazione e di velocità di processing, la lista dei sistemi stabili e utili si è ridotta alle seguenti:

- **Ogr2ogr (Applicazione Desktop):**

Un programma che permette di fare una riproiezione delle coordinate geografiche da un qualsiasi sistema di proiezione al EPSG:900913 utilizzato da Google Maps e Openstreet.

- **Proj4 (Applicazione Desktop):**

Un programma che permette di utilizzare il maggior numero di sistemi geografici esistenti.

- **Openlayers (Api Javascript):**

È una libreria javascript Open Source che consente di visualizzare mappe interattive, questa libreria fornisce delle api che permettono al programmatore di accedere a diverse fonti d'informazioni cartografiche come : Web Map Service, mappe di tipo commerciale come (bing, gMap, Osm...), Web Features Service e diversi formati vettoriali.

- **Api Google Javascript V3 (Api Javascript):**

Le api google permettono di utilizzare il servizio Google Map e tutti i servizi e le funzioni ad esso allegati.

- **GeoExt2 (Api Javascript):**

È una libreria che permette l'integrazione di elementi openlayer dentro una pagina creata con Extjs4, un puro framework javascript che consente di costruire applicazioni web interattive utilizzando tecnologie del tipo: Ajax, DHTML e lo scripting DOM.

Capitolo 3

Web ShapeFile Viewer.

3.1 Come funziona Web ShapeFile Viewer.

Il Web ShapeFile Viewer è un'applicazione pubblicata su Internet che offre la possibilità, agli utenti presenti sul web, di visualizzare i propri dati geografici contenuti in Shapefile, all'interno di alcuni servizi che offrono mappe geografiche. Le mappe introdotte in questo servizio sono OpenStreetMap e Google Map, le motivazioni che portano alla scelta di questi due servizi, al primo impatto simili, ma alla base molto diversi, sono, a mio avviso:

1. - Il bisogno di utilizzare uno dei servizi migliori presenti sul web, ovvero Google Maps che fornisce un'interoperabilità sempre crescente della propria applicazione, anche se, i termini di servizio e le licenze proprietarie, spesso, ne limitano l'utilizzo.
2. - Il desiderio di mettere a confronto il servizio proprietario offerto da google con un servizio open source, ovvero OpenStreetMap che è, come si può leggere sul proprio wiki, “ *un progetto che crea e fornisce dati cartografici liberi e gratuiti a chiunque ne abbia bisogno. Il progetto è stato avviato perché la maggior parte delle mappe che si credono liberamente utilizzabili, hanno invece restrizioni legali o tecniche per l'utilizzo e ciò ne impedisce l'uso per scopi produttivi, creativi o inattesi.*” [16].

A tal proposito citiamo la risposta data da Fabio Antimi alla domanda: “Qual’è brevemente la differenza fra Google Map e OpenStreetMap?” ,presa dall’intervista presente nel Webzine Sanità Pubblica Veterinaria [17]: “La differenza fra queste iniziative sta in due questioni fondamentali. Google Map è pensato principalmente per un utilizzo come stradario automobilistico; Inoltre, essendo curato da un’azienda privata che ne ha fatto un cospicuo business, risulta essere più completo sia come strade che per i nomi delle vie.

Il progetto OpenStreetMap è stato pensato per vari usi, non solo per la navigazione stradale ma anche per utilizzazioni geografiche più generali. In OSM si troveranno, ad esempio, sentieri montani, l’uso del terreno, il trasporto pubblico e molto altro.

Questo servizio essendo basato sull’apporto di volontari, in molte zone, specialmente quelle meno popolate, i dettagli del territorio e le strade risultano essere un po’ carenti, mentre nelle grandi città si hanno normalmente più informazioni, percorsi e spesso anche punti di interesse non presenti in altri contesti (GoogleMap e simili)”.

In aggiunta alla sovrapposizione del vettore sulle mappe, è presente la visualizzazione in SVG del suddetto vettore, risultante dai dati contenuti nello Shapefile. Come definito dal World Wide Web Consortium (W3C): “SVG è un linguaggio di markup per descrivere applicazioni grafiche e immagini bidimensionali, e i relativi script d’interfaccia grafica.” [18]

Con l’utilizzo del linguaggio SVG, il Web ShapeFile Viewer mostra, in una pagina HTML bianca, il vettore, fornendo così l’informazione di base, contenuta nello Shapefile. Di seguito verrà descritto, con l’aiuto di schermate e con infine una rappresentante i casi d’uso del funzionamento del Web ShapeFile Viewer.

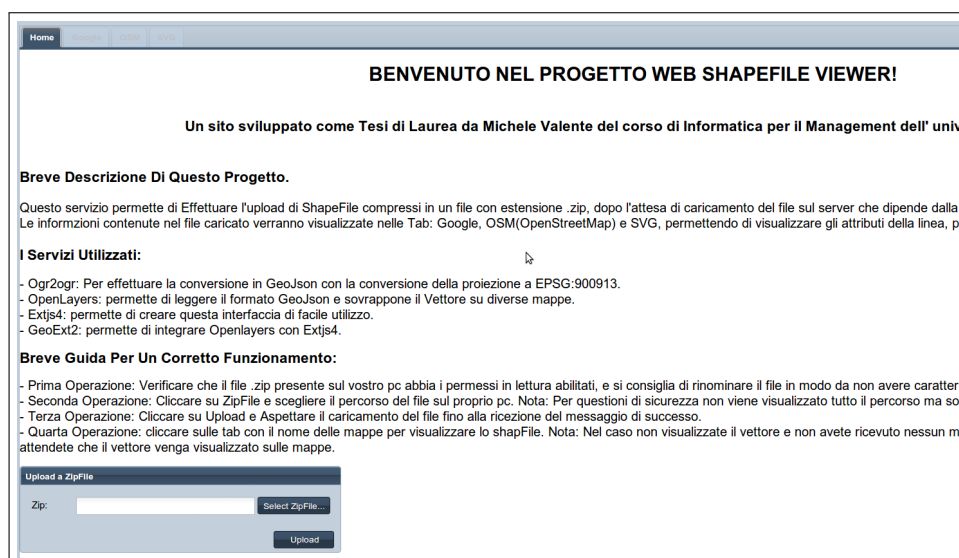


Figura 3.1: Pagina iniziale del Web ShapeFile Viewer

La Figura 3.1 rappresenta la “Home” del Progetto Web ShapeFile Viewer; all’interno di questa pagina è presente una breve descrizione del servizio offerto e degli strumenti utilizzati e subito in basso è presente una breve guida con i consigli utili per un corretto funzionamento del sistema. Questa Schermata, al primo avvio, si presenta con le tre tab Google, Osm e SVG bloccate, come si vede nella figura 3.2, una scelta implementativa che limita la visualizzazione delle mappe solo in successione all’upload di uno Shapefile. La guida presente nella home ci indica che, per procedere, bisogna caricare uno Shapefile in formato corretto e con un nome privo di caratteri speciali,

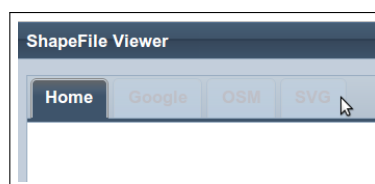


Figura 3.2: Dimostrazione delle Tab Bloccate

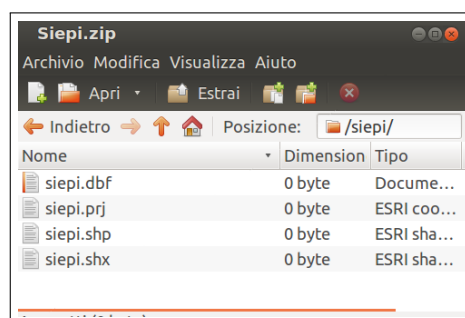


Figura 3.3: Struttura di un Shapefile

come ci dimostra la figura 3.3, in cui si può notare che il pacchetto principale ha un'estensione “.zip”, che contiene i tre file obbligatori e il file “.prj”.

Oltre alla documentazione relativa al servizio offerto, nella home, è presente la sezione relativa all'upload del file

“.zip”, come si può notare, in basso a destra, nell'immagine che descrive la

pagina iniziale del progetto(Figura 3.1). Gli elementi presenti in questa sezione, che necessitano di una qualche precisazione scritta, oltre quella visiva, sono:

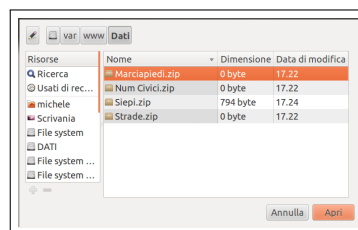


Figura 3.4: Finestra di Scelta Shapefile.

- Il Campo di Testo:

Una nuova politica di sicurezza adottata dai browser (Chrome, Firefox, Safari, Opera..) impedisce la visualizzazione del percorso del file selezionato, che si vorrebbe caricare o allegare per esempio ad una mail. In sostituzione del solito percorso (C:/Desktop/Cartella/file.zip), si presenta la dicitura “C:fakepath/nome del file.zip” per questo, in fase di sviluppo, si è deciso di rinominare la dicitura compresa di “fakepath”, con il solo nome del file, seguito dalla sua estensione.

Questa scelta si è effettuata per evitare possibili problemi per un utente inesperto.

- Il tasto “Select ZipFile.”:

Questo tasto apre una nuova finestra, visibile nella figura 3.4, che permette di selezionare qualsiasi, tipo e formato, di file presente nel proprio pc, in modo da catturarne il percorso in cui si trova, per poi poter eseguire l'upload del file corretto.

- Il tasto “Upload”:

Questo tasto esegue l'operazione di upload dello Shapefile solo nel caso in cui sia stato già selezionato il file “.zip” con le giuste proprietà.

In caso contrario, il programma avviserà l'utente con un messaggio d'errore.

Dopo aver premuto il tasto "Upload" viene visualizzata una finestra di caricamento, la cui durata varia in base alla grandezza del file. Eseguito l'upload di un file corretto e, trascorso il tempo di loading, il programma ci avvisa dell'avvenuto caricamento del file con un messaggio di "Successo" (Figura 3.5).

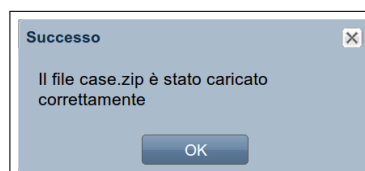


Figura 3.5: Finestra di Successo

A questo punto, in caso di avvenuto successo, le Tab contenenti le mappe subiscono un passaggio di stato, da inattive ad attive (Figura 3.6), fornendo all'utente la possibilità di cliccarci con il mouse e di visualizzare così le mappe desiderate. Le mappe saranno già centrate e avranno lo zoom impostato alla giusta distanza, per visualizzare tutti i componenti del vettore.

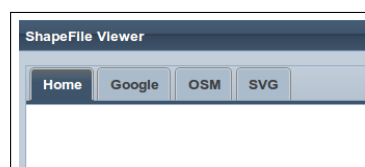
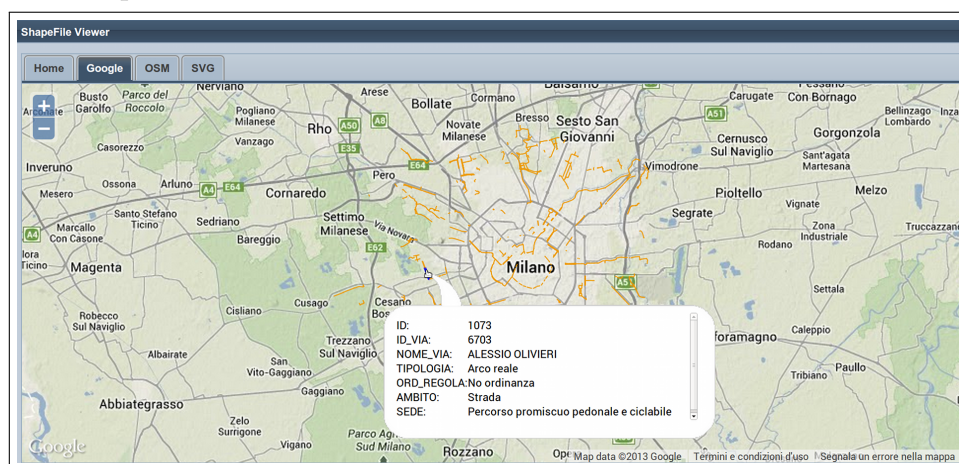
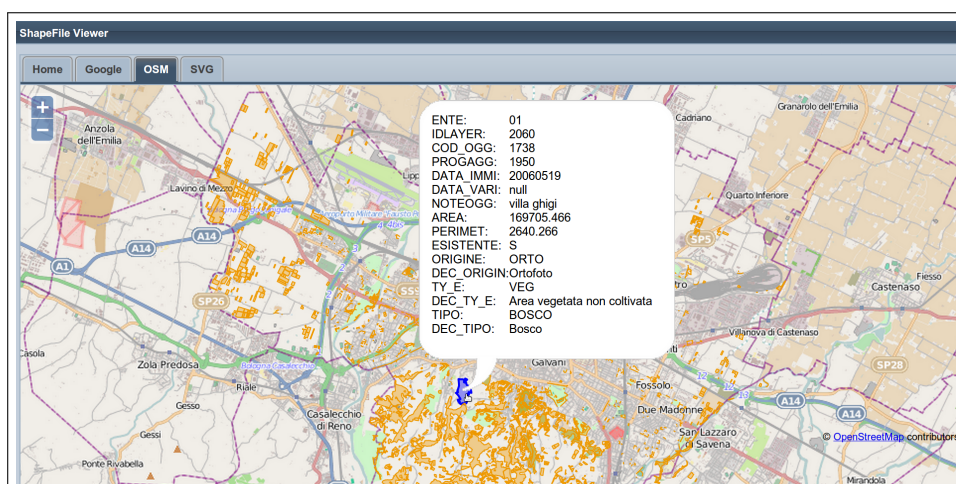


Figura 3.6: Tab Attive

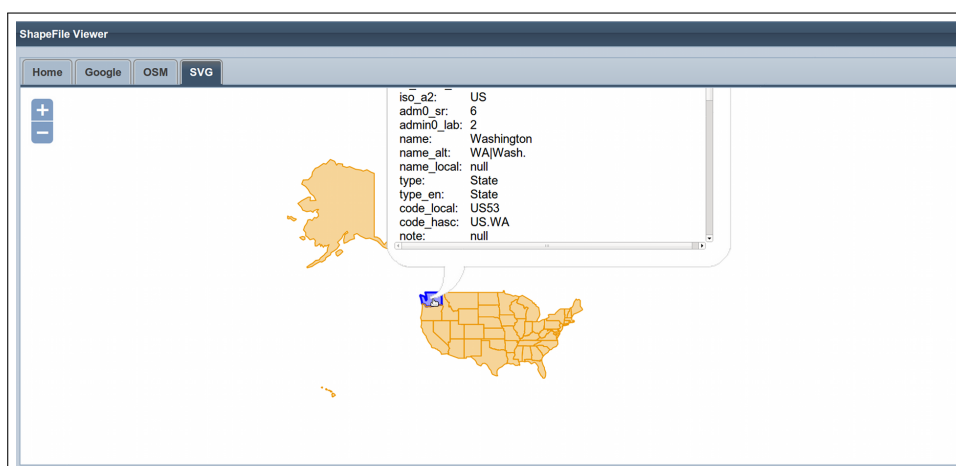
La figura in alto rappresenta la schermata con la mappa di google, in cui è stato caricato e visualizzato, il vettore rappresentante le piste ciclo pedonali di Milano, preso dal sito OpenData Milano [20] in formato Shapefile.



La figura in alto rappresenta la schermata con la mappa di google, in cui è stato caricato e visualizzato, il vettore rappresentante le piste ciclo pedonali di Milano, preso dal sito OpenData Milano [20] in formato Shapefile.



La figura in alto rappresenta la schermata con la mappa di OpenStreetMap, in cui è stato caricato e visualizzato, il vettore relativo al suolo appartenente al comune di Bologna, preso dal sito OpenData Bologna [19] in formato Shapefile.



La figura in alto rappresenta la schermata che visualizza il semplice vettore in SVG. In questa immagine è stato caricato uno Shapefile preso dal sito naturalearthdata.com [21], contenente gli USA. Come si può notare, in ognuna delle tre immagini, è presente una finestra che contiene all'interno gli attributi riferiti ad un solo elemento cliccato sulla mappa. È da sottolineare inoltre, che il Sistema è stato pensato con la finalità di far caricare il proprio Shapefile senza dover effettuare il refresh della pagina.

3.2 Casi d'uso.

Numero:	1
Nome:	Avvio ShapeFile Viewer.
Obiettivo:	Visualizzare nel browser il progetto web Shapefile viewer.
Precondizioni:	Avere una connessione Internet Veloce.
Successo:	In caso di successo il browser mostrerà la pagina iniziale del progetto web Shapefile viewer.
Errore:	In caso d'errore controllare la connessione.
Evento Scatenante:	Inserimento indirizzo http in cui è posto il progetto.
Numero:	2
Nome:	Scelta Shapefile.
Obiettivo:	Sceglie il file da caricare sul Web Shapefile Viewer.
Precondizioni:	Aver effettuato l'accesso al sito.
Successo:	In caso di successo nel campo di testo verrà visualizzato il nome del file da caricare.
Evento Scatenante:	Click con il mouse sul pulsante Scegli fileZip...
Numero:	3
Nome:	Upload Shapefile.
Precondizioni:	Aver selezionato un file obbligatoriamente con estensione “.zip”.

Obiettivo:	Effettuare l'upload dello Shapefile sul servizio web.
Successo:	In caso di successo apparirà una finestra di caricamento e il successivo messaggio del caricamento del file avvenuto correttamente.
Evento Scatenante:	Click con il mouse sul pulsante Upload.
Numero:	4
Nome:	Render Google.
Obiettivo:	Visualizzare la mappa di google con il vettore caricato.
Precondizioni:	Aver sbloccato le Tab Google, Osm, SVG.
Successo:	In caso di successo verrà caricata e visualizzata la google map con all'interno il vettore.
Errore:	In caso di errore non viene eseguita la visualizzazione del vettore, probabilmente il file non era del tutto corretto.
Evento Scatenante:	Click con il mouse sulla tab Google.
Numero:	5
Nome:	Render OpenStreetMap.
Obiettivo:	Visualizzare la mappa di OSM con il vettore caricato.
Precondizioni:	Aver sbloccato le Tab Google, Osm, SVG.
Successo:	In caso di successo verrà caricata e visualizzata la google map con all'interno il vettore.

Errore:	In caso di errore non viene eseguita la visualizzazione del vettore, probabilmente il file non era del tutto corretto.
Evento Scatenante:	Click con il mouse sulla tab OSM.
Numero:	6
Nome:	Render SVG.
Obiettivo:	Visualizzare la pagina HTML con il vettore caricato.
Precondizioni:	Aver sbloccato le Tab Google, Osm, SVG.
Successo:	In caso di successo verrà caricata e visualizzata la una pagina HTML con all'interno il vettore.
Errore:	In caso di errore non viene eseguita la visualizzazione del vettore, probabilmente il file non era del tutto corretto.
Evento Scatenante:	Click con il mouse sulla tab SVG.
Numero:	7
Nome:	Visualizza attributi.
Obiettivo:	Visualizzare la finestrella con la lista degli attributi legati ad un singolo elemento del vettore
Precondizioni:	Aver caricato un vettore.
Successo:	In caso di successo si aprirà un infoWindow con le informazioni richieste.
Evento Scatenante:	Click su di un singolo elemento del vettore.

3.3 La funzione del Web ShapeFile Viewer.

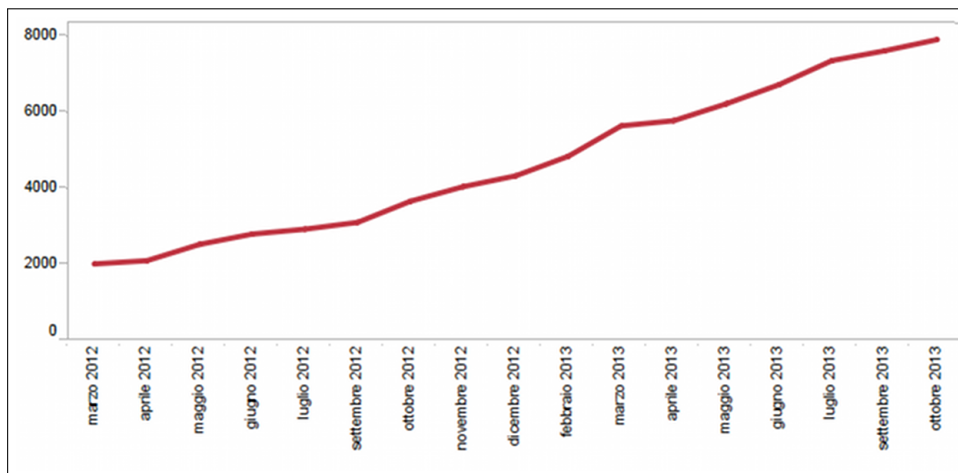


Figura 3.7: Sviluppo OpenData Italia

Il servizio Web ShapeFile Viewer è nato con lo scopo di soddisfare il bisogno di utenti con difficoltà nel visualizzare i propri open data. La diffusione degli open data è sempre in aumento, come ha definito Lorenzo Benussi al Big Data del 2013 [22] e anche come si può vedere dai grafici presenti sul sito data.gov.it [23] da cui è stata estrapolata la figura numero 3.7, che ci dimostra come si siano sviluppati gli open data dal mese di marzo 2012 ad oggi, ottobre 2013. È molto rilevante il picco di dati diffusi in solo un'anno e mezzo, infatti, ad oggi, si registrano circa 8000 data set rilasciati in Italia, approssimativamente 6000 in più rispetto ai 17 mesi precedenti.

Recentemente, gli utenti che utilizzano gli open data per motivi di lavoro, di studio, o per motivi personali, sono in continua diffusione, anche perché è un discorso nuovo e ancora molto aperto in Italia; difatti, leggendo sul sito "datiopen.it" [1]: "solo il 18 Dicembre 2012 il Decreto Legge 18.10.2012, n. 179 è stato convertito in legge (17.12.2012 n. 221) e pubblicato nella Gazzetta Ufficiale: l'art. 9 prevede che i dati e le informazioni forniti dalla pubblica amministrazione dovranno essere obbligatoriamente pubblicati in formato aperto".

I nuovi utenti che hanno bisogno di aprire file open data ed in particolare gli Shapefile, spesso incontrano delle difficoltà nel trovare un programma adatto alle proprie esigenze, come si può leggere da alcuni blog trovati tramite il motore di ricerca google:

- L'utente Mohang il 09/02/2011 chiede: 'How to open the Shape files and extract the data from these files?'. [24]
- L'utente Nathan W 30/08/2013 scrive: "Opening .shp file under Add Vector Layer?". [25]
- L'utente Forester7 il 30/07/2013 scrive: "Is there free software to open Shapefiles?". [26]

Di conseguenza, la funzione di questo progetto è di impedire, ad utenti inesperti, con la sola necessità di visualizzare il contenuto del loro Shapefile, di avere questi problemi:

- Necessità d'avere una connessione ad internet per cercare e scaricare un GIS desktop.
- Ricercare un programma adatto al sistema operativo in possesso.
- Necessità di installare un GIS avanzato, con le proprie difficoltà, per una semplice visualizzazione.

Tutti questi problemi vengono risolti con un sistema per la visualizzazione di vettori Shapefile posto sul web, come ad esempio il Web ShapeFile Viewer.

Capitolo 4

Descrizione della architettura del Web ShapeFile Viewer.

4.1 Composizione dell'architettura software.

Una premessa importante da fare, prima di descrivere cosa fa realmente il codice, riguarda la struttura dei dati utilizzati da questo software. Gli Shapefile sono file binari che custodiscono, al loro interno, dati geografici e, in specifico, coordinate geospaziali. Le coordinate, in base alla proiezione di riferimento, possono essere composte da un numero elevato di cifre. Per questo motivo, per evitare l'arrotondamento delle ultime cifre e limitare i tempi di lettura del file binario, si è deciso di utilizzare una applicazione desktop chiamata ogr2ogr, che verrà descritta nel paragrafo 4.3. A seguito di tale premessa, si può passare alla descrizione della composizione strutturale del progetto.

Il Web ShapeFile Viewer è stato sviluppato grazie all'ausilio di due linguaggi di programmazione, il php, conosciuto per le potenzialità di programmazione serverside ed il javascript un linguaggio di scripting clientside, che offre la possibilità di includere framework e librerie esterne. Iniziamo la descrizione del progetto dai file scritti in javascript, in quanto risulta essere il linguaggio con cui l'utente, ospite del sito, ha il primo contatto.

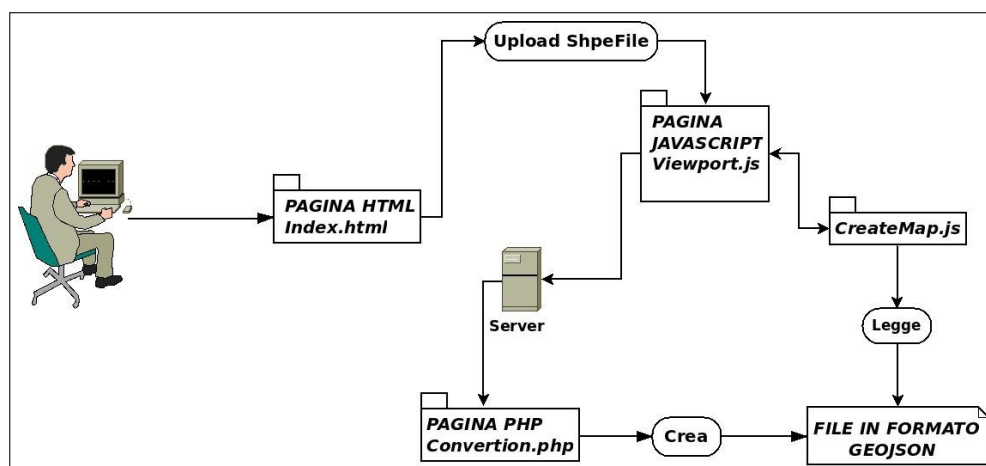


Figura 4.1: Processo Software

In questo linguaggio sono stati scritti 2 file, il cui compito è creare l'interfaccia grafica con cui l'utente interagisce. Il file con nome "CreateMap.js" ha la funzione di fornire le mappe e il vettore da visualizzare nel sito. Il file principale invece, è il "Viewport.js", che contiene molte funzioni del framework ExtJS4, permettendo la creazione di un ambiente grafico di facile utilizzo, dinamico e a mio avviso: "con un aspetto gradevole".

L'unico file in php di cui è importante parlare, invece, è "Conversion.php" che attiva le sue funzioni di conversione, come si può dedurre dal nome, quando è contattato dal file "Viewport.js".

La figura 4.1 rappresenta un po' quello che avviene quando un nuovo utente si connette al Web ShapeFile Viewer:

1. L'utente che si connette al sito, visualizza la home page, perché il sistema contatta la pagina "Index.html" e mostra gli elementi in essa contenuti.
2. L'utente sceglie un file .zip da caricare e clicca sul tasto "upload" presente nella pagina html, per dare il via al processo di caricamento.
3. La pagina Viewport.js viene contattata dalla pagina html per eseguire l'operazione di upload.

4. Il Viewport.js invia una richiesta al server in cui è presente la pagina Conversion.php.
5. La pagina Conversion.php effettua le conversioni necessarie per creare un file GeoJson.
6. Nel frattempo, il file CreateMap.js, contattato dal Viewport.js, aspetta che sia creato il file GeoJson per poterlo leggere e visualizzare.

La struttura complessiva dei file è abbastanza semplice, questo perché una gran parte delle operazioni più difficili le fornisce la libreria ogr2ogr. Quindi, le difficoltà di questo progetto sono improntate sull'integrazione di diversi moduli con funzioni completamente differenti, come ad esempio, l'integrazione di Openlayers con il framework ExtJs4, che verrà descritta nel paragrafo 4.5 inerente alle difficoltà incontrate durante la progettazione.

4.2 Ruolo e struttura delle funzioni rilevanti

In questo paragrafo verrà descritto nello specifico come funziona il sistema e come si comportano le diverse funzioni, presenti sia negli script serverside che su quelli clientside. Verranno specificate le operazioni di sistema sia nei casi di “funzionamento ottimale”, che in caso di comportamenti erranei:

- **Funzionamento Ottimale:**
Il sistema riesce ad effettuare tutte le operazioni realizzando il risultato sperato.
- **Comportamenti Erranei:**
Il sistema riscontra qualche problema dovuto ad un errore umano (come il caricamento di un file sbagliato) o un problema di sistema nel caso non sia stata gestita, volutamente dagli sviluppatori, qualche funzione particolare che non permetta la fruizione del risultato desiderato.

Ogni file presente in questo progetto verrà preso in esame e descritto in dettaglio. I file con meno importanza, non per il funzionamento ma per il contenuto privo di innovazione, verranno descritti in generale senza entrare nello specifico e magari nell'ovvio.

- **“Index.html”**:

È la pagina html che avvia il sistema. Quando viene inserito l'url del progetto Web ShapeFile Viewer, viene effettuata di default una chiamata a questa pagina che a sua volta, integra e richiama tutti gli script e le librerie necessarie al proprio funzionamento come per esempio: le api google, extjs4, openlayers e gli script javascript descritti nei punti successivi.

- **“Viewport.js”**: Questo file contiene le funzioni che eseguono le seguenti operazioni:

1. Effettua l'importazione delle funzioni, contenute nelle librerie ExtJs4 e GeoExt2, relative agli elementi grafici usati in questo progetto.
2. Crea la funzione Ext.onReady, presenti nelle api ExtJs4, che avvia la creazione degli elementi grafici e all'interno richiama le funzioni del file “CreateMap.js”, che avviano la realizzazione delle mappe da inserire nelle schede.
3. Crea il Viewport, da cui deriva il nome, ovvero un container di elementi grafici che si adatta dinamicamente a tutta la grandezza della pagina web.
4. Inserisce all'intero del Viewport un elemento grafico chiamato TabPanel¹ che crea una finestra centrale con 4 schede intercambiabili:² la Home, Google, OSM e SVG, di cui le ultime 3 sono

¹Visibile nella figura 3.1 del capitolo 3 con la particolarità delle Schede ancora disattivate.

²Visibile alla figura 3.6 del capitolo 3.

inizialmente disattivate per non visualizzare le mappe prima di caricare un vettore.

5. Inserisce, all'interno della scheda "Home", il form relativo al caricamento del file sul server: la FileField, che permette di selezionare un file presente sul proprio pc, e il tasto "Upload", che invia la richiesta POST al server passando come parametro il percorso del file scelto.

Come già anticipato nel capitolo 3, i browser, per motivi di sicurezza, non permettono di visualizzare all'interno della FileField il percorso esatto del file ma una stringa del tipo: "C:/fakepath/nome del file.zip" che abbiamo preferito ridurre al solo nome del file.zip. In questa sezione sono state inserite anche le funzioni che gestiscono il successo e gli errori risultanti dal caricamento del file. Nel caso in cui il caricamento avvenga con successo, verrà visualizzato un messaggio di conferma e verranno chiamate le funzioni di creazione del vettore, presenti nel file "CreateMap.js".

In caso di Errore, il sistema si limiterà a visualizzare il messaggio d'errore che il server da in restituzione alla chiamata effettuata.

6. Inserisce un listener legato alle schede Google, OSM e SVG che gestisce il primo click su una di esse richiamando una funzione di zoom che permette la visualizzazione dell'intero vettore; questo avviene solo nel caso in cui sia stata già effettuata la creazione delle mappe e il caricamento del vettore.

- **"CreateMap.js":**

In generale questo file contiene le funzioni che operano maggiormente sul vettore e sulle mappe:

1. Funzione che imposta lo stile che deve avere il vettore; viene predefinito che nel caso in cui si passa con il mouse al di sopra di una feature, si effettua un select, ovvero l'elemento cambia il colore da quello di default in blu.

2. Funzioni `mappa1()`, `mappa2()`, `mappa3()` che, con l'ausilio della libreria `openlayers`, effettuano la creazione dei 2 elementi `OpenLayers.Map`, contenenti le mappe Google Maps e OSM, e la creazione dell'elemento `OpenLayers.Map` che conterrà il vettore in formato SVG che grazie alla proprietà `"AllOverlays"` delle api `Openlayers`, permette la visualizzazione di un vettore senza l'obbligo di rendering di una mappa.
3. Funzione che, con l'ausilio della libreria `openlayers`, effettua una chiamata asincrona GET per il caricamento del file in formato GeoJson presente sul server, ed effettua, inoltre, il parse delle features e posiziona il vettore sulla mappa passata come parametro. In questa funzione sono aggiunti inoltre diversi listeners:
 - `FeaturesAdded`:
Quando tutte le features del vettore vengono inserite sulle mappe, effettua una chiamata al file `"Remove.php"` presente sul server e opera il passaggio di stato delle schede del `tabPanel`, da inattive a attive³
 - `FeatureClick`:
Quando si clicca su di un feature, ovvero un singolo elemento del vettore, si apre una finestra detta `PopUp`, in cui vengono visualizzati gli attributi legati a questo singolo elemento.
 - `NoFeatureClick`:
Nel caso in cui sia già aperto un `PopUp`, quando si clicca al di fuori del vettore, quindi non su di un feature, viene chiamata la funzione che rimuove i `PopUp` dalla mappa.

- **"Conversion.php"**:

Questo è il file che esegue le operazioni più importanti, ovvero la conversione della proiezione del vettore e quella dello Shapefile in un file di formato GeoJson con estensione `".json"`. Il file viene contattato trami-

³Come si può vedere nelle figure 3.2 e 3.6 del capitolo 3

te una chiamata POST dalla funzione che gestisce l'evento di click del pulsante Upload, presente nel file "Viewport.js", in cui gli viene passato come parametro il percorso del file su cui effettuare le operazioni. Le attività effettuate in successione alla ricezione della chiamata sono:

1. Controllo che il path(percorso), ricevuto come parametro post, sia riferito ad un file di formato zip, in caso contrario restituisce, come risposta alla chiamata, un messaggio d'errore che riassume il problema descritto.
2. Nel caso in cui il path, passato come parametro, si riferisca ad un file zip, effettua l'estrazione del contenuto del file, in una cartella.
3. Effettuata l'estrazione con successo, viene richiamata la funzione `cercaFile()`, in a cui vengono passati come parametri il nome della cartella e del file da ricercare.
4. Si effettua una ricerca all'interno della cartella, per verificare che ci sia un file con estensione ".shp"; nel caso in cui non ci sia, il server risponde con un messaggio che specifica l'inesistenza di questo file.
5. Effettuata la verifica con responso positivo, si passa al controllo della proiezione con l'applicazione `ogr2ogr`; nel caso in cui il responso sia che la proiezione ha come identificativo UTM32, l'applicazione `ogr2ogr` effettuerà un riproiezione da EPSG:23032, la proiezione italiana, a EPSG:900913, la proiezione necessaria per visualizzare i vettori sulle mappe.

Questo controllo viene effettuato perché, nel caso della proiezione UTM32, l'applicazione `ogr2ogr` necessita di specificare anche la proiezione in input per effettuare una riproiezione migliore e senza imperfezioni; così facendo, la riproiezione avviene in modo più preciso perché l'applicazione `desktop proj4`, che descriveremo nel paragrafo 4.3, si attiva in aiuto dell'applicazione `ogr2ogr`.

Nel caso in cui invece la proiezione di riferimento non sia una

UTM32, l'applicazione ogr2ogr effettua una conversione di proiezione un po' più generica, ovvero senza avere una proiezione di input; in questo modo per le proiezioni più comuni verrà comunque fatta una riproiezione ottimale.

6. In contemporanea alla riproiezione, l'applicazione ogr2ogr effettua la conversione del formato da Shapefile a GeoJson, creando così un file con estensione “.json” contenente le informazioni presenti nello Shapefile.
7. Le ultime operazioni sono l'eliminazione dello Shapefile dal server, la modifica dei permessi dei file creati e la risposta di successo da inviare all'applicazione client.

- **“Remove.php”:**

Questo file è un semplice script serverside che viene contattato con una richiesta POST http dalla funzione creaVettore(), presente nel file “CreateMap.js”, a cui viene passato come parametro il nome del file con estensione “.json”, che deve essere eliminato dal server dopo la lettura da parte del sistema client. Questa operazione è necessaria per non tenere traccia dei dati altrui e per non occupare lo spazio disponibile nel server, rendendolo inutilizzabile.

4.3 Tecnologie Utilizzate.

Sono le tecnologie utilizzate a rendere questo progetto funzionale e di bell'aspetto, tra cui le più importanti sono:

- **Linux Ubuntu Precise Pangolin:**

Ubuntu Precise Pangoline è il sistema operativo open source della macchina dell'università di Bologna, in cui è installato il server Web e dove è stato caricato il progetto Web ShapeFile Viewer. Ubuntu 12.04 è una versione “LTS” (Supporto a lungo Termine) come si può leggere nella

descrizione ufficiale del sito marchiato Ubuntu [27]; tale release, avendo questa caratteristica, definisce un supporto duraturo e quindi una stabilità che nel tempo può risultare molto utile.

- **Server Apache:**

Il server apache, in cui è depositato il progetto Web Shape File Viewer, non è aggiornato all'ultima versione stabile di questo software, ovvero la 2.4.6, ma alla versione 2.0. Questa scelta non è stata dettata da motivazioni di funzionalità o come risultato della progettazione del Web ShapeFile Viewer, ma è stata necessaria per il funzionamento del progetto sulle macchine della facoltà e per l'utilizzo delle librerie importate. Questa release, come si può leggere nella documentazione ufficiale [28], al momento del rilascio, differentemente alla versione precedente, ha presentato queste nuove funzionalità:

- Unix Threading:

Apache can now run in a hybrid multiprocess, multithreaded mode. This improves scalability for many, but not all configurations.

- New Build System:

The build system has been rewritten from scratch to be based on autoconf and libtool.

- Better support for non-Unix platforms:

Apache 2.0 is faster and more stable on non-Unix platforms such as BeOS, OS/2, and Windows.

- IPv6 Support:

On systems where IPv6 is supported by the underlying Apache Portable Runtime library, Apache gets IPv6 listening sockets by default.

- **OpenLayers:**

La versione della libreria utilizzata in questo progetto è la 2.13.1 l'ultima release stabile ad essere stata sviluppata; sul proprio sito [29] si

leggono testuali parole:

“OpenLayers makes it easy to put a dynamic map in any web page. It can display map tiles and markers loaded from any source. OpenLayers has been developed to further the use of geographic information of all kinds. OpenLayers is completely free, Open Source JavaScript, released under the 2-clause BSD License (also known as the FreeBSD)”. Quindi la possibilità di caricare ogni tipo formato e di usare ogni informazione geografica, rendono questa libreria indispensabile per questo progetto.

- **ExtJS4:**

È a mio avviso un framework dalle potenzialità elevate, che permette la realizzazione in modo semplice ed immediato, di pagine web dinamiche. L'utilizzo di questa libreria da un tocco di professionalità maggiore al Web ShapeFile Viewer.

La versione utilizzata in questo progetto è la 4.1.3; anche se è la più aggiornata, non fornisce in maniera ottimale il rendering di più mappe nello stesso sito, infatti, per unire l'utilizzo di openlayers che fornisce le mappe e Extjs che fornisce l'ambiente grafico, è stato necessario utilizzare una terza libreria chiamata GeoExt2 che descriveremo successivamente.

ExtJS4 ha un'ottima reputazione riguardo il suo funzionamento nella maggior parte dei browser in circolazione, infatti, gli sviluppatori lo sottolineano anche nella documentazione ufficiale [30], affermando: *“that works on all modern browsers from IE6 to the latest version of Chrome”*.

- **GeoExt2:**

Questa libreria crea il collegamento tra le potenzialità grafiche del framework extjs e le funzionalità geografiche della libreria openlayers. Questo framework adoperato nel progetto, permette di definire il “type” del contenuto presente nelle tab come un oggetto geoExt, cosicché openlayers riesce ad operare all'interno della tab mostrando le mappe

ed il vettore. Nella descrizione ufficiale [31] si legge:

“GeoExt is Open Source and enables building desktop-like GIS applications through the web. It is a JavaScript framework that combines the GIS functionality of OpenLayers with the user interface of the ExtJS library provided by Sencha”.

- **Ogr2ogr:**

Questa applicazione è un modulo della libreria denominata GDAL (Geospatial Data Abstraction Library) [32] la cui descrizione è: *“is a translator library for raster geospatial data formats that is released under an X/MIT style Open Source license by the Open Source Geospatial Foundation”*. Il modulo ogr2ogr permette la conversione delle coordinate dei vettori utilizzati in questo progetto e del formato del file da Shapefile a GeoJson, infatti gli sviluppatori affermano nel sito di questo modulo [33]:

“This program can be used to convert simple features data between file formats performing various operations during the process such as spatial or attribute selections, reducing the set of attributes, setting the output coordinate system or even reprojecting the features during translation”.

4.4 Informazioni per l’eventuale modifica da parte di terzi.

Questo paragrafo è stato pensato per aiutare coloro che vogliono usufruire dei codici del progetto per un utilizzo proprio, per apportare delle modifiche o meglio per un integrazione con le proprie tecnologie.

- **Supporto Per Proiezioni Particolari:**

Questo progetto potrebbe non garantire una “riproiezione” precisa per alcune delle proiezioni di minore utilizzo, quindi, nel caso in cui si abbia necessità di effettuare riproiezioni di questo tipo, le modifiche da

effettuare sono riferite al file `Conversion.php` nella funzione `cercaFile()` e in particolare, nelle righe in cui si specifica anche la conversione di input. Naturalmente, andrà sostituita la proiezione UTM32 con la proiezione desiderata.

- **Aggiunta di Nuove Mappe:**

OpenLayers fornisce molti altri servizi di mappe che, in fase di sviluppo del Web ShapeFile Viewer si è deciso di non implementare. Le modifiche per l'aggiunta di una nuova mappa si riferiscono a due file principali, ovvero `Viewport.js` e `CreateMap.js`; nel primo bisogna semplicemente creare una nuova tab e richiamare le funzioni di creazione della mappa e quella di creazione del vettore, il cui parametro sarà la mappa creata in precedenza; di conseguenza, la modifica da effettuare nel `CreateMap.js` sarà la definizione di un'ulteriore funzione, per la creazione delle mappe, con i parametri settati in modo da far fruire il servizio di mappe desiderato.

- **Supporto Per Ulteriori Formati:**

Per definire il supporto per ulteriori formati diversi dal file.zip bisogna agire nel file `Conversion.php`, in cui la prima parte si occupa dell'upload del file.zip e della sua estrazione.

- **Aggiunta di Nuovi Tool:**

Per l'aggiunta di nuovi tool è necessario uno studio approfondito delle librerie OpenLayers, ExtJS e GeoExt, in quanto, di default, contengono molti tool adattabili a questa applicazione, come ad esempio: Effetti grafici, tabelle delle features, zoomSliders di diverso tipo e molte altre utilità.

4.5 Difficoltà incontrate durante la progettazione.

Considerando la quantità di codice ma non la sua qualità, questo non può essere definito un “Gran Progetto”, in quanto le dimensioni non sono tali da rendere veramente necessaria l’unione di un team per lo sviluppo. È da considerare che lo sviluppo di questo progetto derivi da delle conoscenze veramente misere riguardo la materia che concerne la creazione di GIS e WebGIS, infatti lo sviluppatore, prima di procedere alla progettazione del sistema, ha necessariamente dovuto effettuare uno studio inizialmente basilare e successivamente più approfondito, della suddetta materia. Nonostante ciò, in fase di progettazione, lo sviluppatore, non conoscendo le tecnologie esistenti che operano in questo campo, ha dovuto testarne la maggior parte, tra le quali:

1. OGRE:

Un servizio web basato sulla libreria GDAL, che permette una trasformazione dello Shapefile in geoJson. Le limitazioni di questo servizio, che hanno definito il suo abbandono sono, l’obbligo d’avere degli Shapefile con proiezione WGS84 e le tempistiche elevate di creazione e del download sul server del file.

2. FwTool:

Una Applicazione Desktop che offre la possibilità di utilizzare tutte le funzionalità della libreria GDAL direttamente sul server. L’abbandono di questa applicazione è decretato da un’ incompatibilità con i sistemi delle macchine della facoltà e dalla necessità di un sistema leggero, senza funzionalità inutilizzate.

3. Shapefile-js:

Utilizza le funzionalità del Html5 per effettuare una lettura del file binario “.shp” e del file “.dbf”. Questa libreria ha un risultato in lettura molto positivo, ma non offre una riproiezione ottimale dello Shapefile.

4. D3.js Una libreria che offre potenzialità elevate nella creazione di grafici e vettori dinamici. Per poterla integrare con il sistema necessitava comunque dell'utilizzo del servizio OpenLayers. In successione alla scoperta delle funzionalità di OpenLayers si è deciso di eliminare questa libreria dal sistema.
5. Proj4.js e Proj4.php
Questi due moduli sono una traduzione dell'applicazione Proj4 presente nella libreria GDAL; la loro funzione è principalmente quella di effettuare la proiezione delle coordinate. Dopo diversi test il risultato è stato inconcludente, perché i dati riproiettati presentavano degli arrotondamenti considerevoli che modificavano la rappresentazione del vettore.
6. GeoExt1:
La versione precedente della libreria GeoExt2 compatibile solo ed esclusivamente con la libreria ExtJs3 e con una versione poco recente di OpenLayers. Anche essendo ancora in fase di sviluppo, si è deciso di utilizzare la seconda versione per la possibilità di integrare in questo progetto le librerie più recenti.

Capitolo 5

Valutazione funzionale del Web ShapeFile Viewer

5.1 Efficienza e non efficienza

Il Web shapeFile Viewer è stato progettato in modo tale da supportare tutte le primitive del vettore, ovvero: la linea, il punto, il poligono e il testo, ed inoltre in modo tale da poter visualizzare la maggior parte degli Shapefile presenti sul web con le proiezioni più svariate.

Oltre ai test, effettuati in fase di sviluppo, per la verifica della correttezza delle nuove funzioni aggiunte, sono stati effettuati dei test per testarne l'efficienza, quando il progetto ormai era terminato. I test effettuati si riferiscono ad un campione di 12 file con dati relativi al territorio regionale italiano, nazionale ed internazionale. Prima di passare alla descrizione dei dati presenti nei grafici successivi, bisogna mettere in risalto il successo del test più importante, che mette sotto verifica l'utilità di questo progetto: "Di 12 Shapefile con diverse caratteristiche, scaricati da alcuni siti openData, 12 sono stati caricati e visualizzati senza errori sul servizio Web ShapeFile Viewer". Questa affermazione dimostra l'utilità di questo sistema in quanto è possibile scaricare e visualizzare immediatamente gli Shapefile presenti sul web.

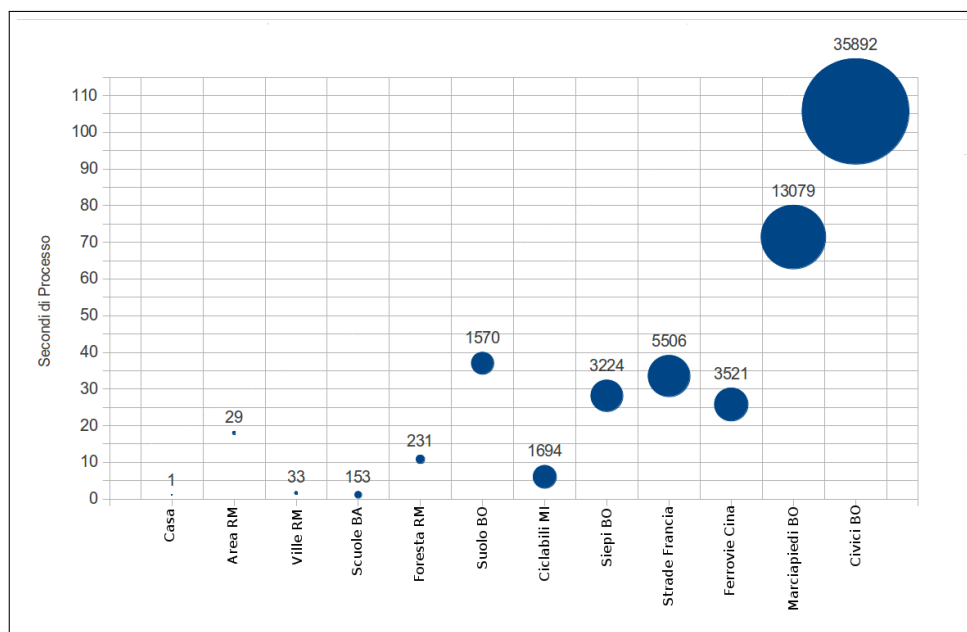


Figura 5.1: Tempi di Upload e Visualizzazione.

La grandezza delle bolle mostra il numero delle features.

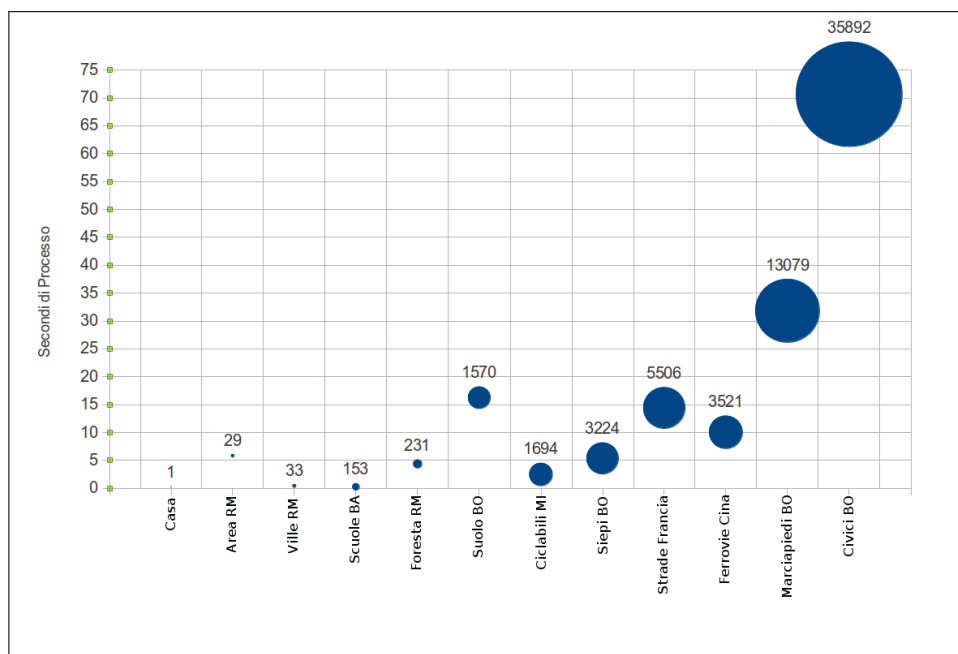


Figura 5.2: Tempi per la Visualizzazione

Nei grafici 5.1 e 5.2 sono raffigurati rispettivamente i tempi di “Upload e Visualizzazione” per il primo e i tempi di “Sola Visualizzazione” per il secondo, di Shapefile molto diversi, che ora andremo a descrivere:

1. Casa:

Shapefile con 1 solo feature di tipo “LineString”, con sistema di Riferimento UTM32, creato appositamente per la rappresentazione.

Tempo Upload e Visualizzazione: 1,181 secondi.

Tempo sola Visualizzazione: “0,278 secondi”.

2. Aree Protette Roma:

Shapefile con 29 features di tipo “Polygon”, con sistema di riferimento EPSG:3004, il download è stato effettuato dal sito:

“<http://dati.comune.roma.it/>”.

Tempo Upload e Visualizzazione: 18,010 secondi.

Tempo sola Visualizzazione: 5,847 secondi.

3. Ville Storiche Roma:

Shapefile con 33 features di tipo “Polygon”, con sistema di riferimento EPSG:3004, il download è stato effettuato dal sito:

“<http://dati.comune.roma.it/>”.

Tempo Upload e Visualizzazione: 1,637 secondi.

Tempo sola Visualizzazione: 0,442 secondi.

4. Scuole Bari:

Shapefile con 153 features di tipo “Point”, con sistema di riferimento WGS84, il download è stato effettuato dal sito:

“<http://www.datiopen.it/>”.

Tempo Upload e Visualizzazione: 1,147 secondi.

Tempo sola Visualizzazione: 0,275 secondi.

5. Deforestazione Roma:

Shapefile con 231 features di tipo “Polygon”, con sistema di riferimento EPSG:3004, il download è stato effettuato dal sito:

- “<http://dati.comune.roma.it/>” .
Tempo Upload e Visualizzazione: 10,837 secondi.
Tempo sola Visualizzazione: 4,378 secondi.
6. Suolo Bologna: Shapefile con 1570 features di tipo “Polygon”, con sistema di riferimento UTM32, il download è stato effettuato dal sito: “<http://dati.comune.bologna.it/>” .
Tempo Upload e Visualizzazione: 37,005 secondi.
Tempo sola Visualizzazione: 16,247 secondi.
7. Percorsi Ciclabili Milano:
Shapefile con 1694 features di tipo “LineString”, con sistema di riferimento EPSG:3003, il download è stato effettuato dal sito: “<http://dati.comune.milano.it/>” .
Tempo Upload e Visualizzazione: 6,044 secondi .
Tempo sola Visualizzazione:2,491 secondi.
8. Siepi Bologna:
Shapefile con 3224 features di tipo “Polygon”, con sistema di riferimento UTM32, il download è stato effettuato dal sito: “<http://dati.comune.bologna.it/>” .
Tempo Upload e Visualizzazione: 28,142 secondi.
Tempo sola Visualizzazione: 5,37 secondi.
9. : Strade Francia:
Shapefile con 5506 features di tipo “LineString”, con sistema di riferimento EPSG:4326, il download è stato effettuato dal sito: “<http://www.diva-gis.org/gdata>” .
Tempo Upload e Visualizzazione: 33,526 secondi.
Tempo sola Visualizzazione: 14,416 secondi.
10. Ferrovie China:
Shapefile con 3521 features di tipo “LineString”, con sistema di riferimento WGS84, il download è stato effettuato dal sito:

“<http://www.diva-gis.org/gdata>”.

Tempo Upload e Visualizzazione: 25,782 secondi.

Tempo sola Visualizzazione: 10,058 secondi.

11. Marciapiedi Bologna:

Shapefile con 13079 features di tipo “LineString”, con sistema di riferimento UTM32, il download è stato effettuato dal sito:

“<http://dati.comune.bologna.it/>”.

Tempo Upload e Visualizzazione: 71,433 secondi.

Tempo sola Visualizzazione: 31,831 secondi.

12. Civici Bologna:

Shapefile con 35892 features di tipo “Point”, con sistema di riferimento UTM32, il download è stato effettuato dal sito:

“<http://dati.comune.bologna.it/>”.

Tempo Upload e Visualizzazione: 105,663 secondi.

Tempo sola Visualizzazione: 70,653 secondi.

Come si può vedere dai grafici, al variare della tipologia e della proiezione dello Shapefile, lo scostamento in secondi del tempo di processo, non varia in modo significativo; invece, considerando il numero di features e quindi il peso del file, si nota una crescita importante dei tempi di processo, all’aumentare del numero delle features contenute negli Shapefile. Dai tempi di processo, si può dedurre quindi, che con file di grandi dimensioni, il sistema non mostra una grande efficienza; ma sicuramente svolge il proprio compito in modo impeccabile, evitando all’utente di dover scaricare una applicazione “GIS Desktop” perdendo così molto più tempo, in confronto ai tempi di processo analizzati.

5.2 Pregi e difetti

In questo paragrafo verranno descritti i pregi ed i difetti, riguardanti il funzionamento del sistema Web Shapefile Viewer, prendendo in considerazione gli elementi, che per un utente semplice, sono necessari o sono un problema per la fruizione di un buon servizio. I difetti maggiori di questo sistema sono:

- Nel caso della mappa di google, quando si opera uno spostamento, per esempio, per centrare il vettore, oppure la funzione di zoom, si notano degli spostamenti, spesso evidenti del vettore; il problema è scaturito dalla libreria openlayers con non effettua una “bind”, ovvero un’unione perfetta, fra il vettore e la mappa offerta da google.
- Per eseguire una riproiezione ed una trasformazione, ottimale, dello Shapefile, è necessario dover caricare il file sul server, portando così tempi di processo a delle grandezze elevate; la necessità di effettuare questa operazione deriva da ogr2ogr che obbliga la lettura di un file e non di altri tipi di formato.
- Temporaneamente, non è stato possibile aggiungere la funzione per l’esportazione delle mappe in altri formati o semplicemente come immagini; il problema si è verificato per la necessità di dover installare l’applicazione mapFish, non supportata dai server della facoltà.
- Il funzionamento ottimale di questo, essendo un Web Gis, è dipendente dal tipo di servizio internet in possesso del client; nel caso in cui il client avesse una connessione lenta, il programma, funzionerebbe lo stesso ma i tempi di processo raggiungerebbero cifre molto elevate.
- Temporaneamente è stata offerta la sola possibilità di caricare file di formato “.zip”, escludendo così la visualizzazione di altri formati contenenti dati georeferenziali, ma come si può leggere nelle conclusioni, sarà una delle novità presenti nella prossima release.

- Al primo avvio da parte del client, non avendo nessun dato salvato in “cache”, il caricamento del servizio risulterà più lento; questo problema è scaturito dalla necessità, di dover caricare sulla pagina, le mappe offerte da OpenStreetMap e Google Map.

I pregi e l’efficacia di questo progetto, invece si riferiscono ad i seguenti punti:

- Questo progetto consente una visualizzazione dinamica dei vettori; qualsiasi sia il tipo (Line, Point, Polygon) e qualsiasi sia la proiezione (WGS84, UTM32, ecc), il sistema visualizzerà i dati, in modo impeccabile, sulle mappe.
- Come abbiamo detto in precedenza, per gli Shapefile con un numero di features eccessivo, il sistema non offrirà un servizio ottimale ma, in contrario, per Shapefile di piccola-media grandezza, il Web Shape-File viewer, offrirà un servizio migliore, in confronto alle applicazioni presenti sul web e analizzate nel capitolo 2.
- A differenza di altri visualizzatori web di vettori, questo servizio, offre anche la funzione che permette di mostrare i dati relativi ad un singolo elemento, con il semplice click del mouse.
- Grazie all’utilizzo di openlayers, questo sistema, fornisce l’opportunità di visualizzare il vettore in mappe diverse e con livelli di zoom differenti.
- La struttura dei file e l’utilizzo di ExtJs, permette, una manutenzione ed un ampliamento delle funzionalità facilitato. Consentendo, agli sviluppatori interessati a questo codice, di effettuare le modifiche senza problemi eccessivi.
- In confronto ad altri sistemi presenti sul web, il Web ShapeFile Viewer è supportato dai diversi browser ad oggi maggiormente utilizzati, così facendo qualsiasi siano le preferenze dell’utente riguardo i browser, il servizio sarà disponibile.

- Grazie ad ExtJs ed agli elementi già sviluppati, l'interfaccia grafica presente in questo servizio, si mostra di facile utilizzo ed intuizione, evitando la necessità di consultare guide supplementari.
- Come abbiamo detto in precedenza, spesso gli utenti sono restii a mettere in circolazione i propri dati, quindi per evitare la fuga di informazioni, il sistema è stato sviluppato, in modo da eliminare le informazioni, dopo averle visualizzate .

Capitolo 6

Conclusioni

Io tesista dopo aver effettuato questo studio approfondito del argomento open data, sviluppando così le conoscenze adatte alla creazione di un webgis, ritengo che il risultato conseguito da questa tesi, potrebbe essere il primo passo per la creazione di una applicazione webGis dalle funzionalità elevate. Si può quindi affermare, in conclusione, che il servizio creato potrebbe essere di grande utilità, per quella fascia di utenti presi in esame, ovvero, gli utenti con delle conoscenze non elevate in materia gis.

Da una ricerca dei gis funzionanti presenti sul web, è risultata una carenza sostanziale di questi servizi, quindi la fruizione di un webGis funzionante e di facile utilizzo, potrebbe garantire un impiego maggiore, da parte degli utenti semplici, degli open data di tipo geografico. Le carenze risultanti dalla ricerca, si riferiscono soprattutto alla presenza di webGis che accettino, senza bloccarsi, tutte le proiezioni esistenti e prendano in input Shapefile con un numero elevato di features.

Gli elementi di questo progetto, per me tesista, motivo d'orgoglio, sono i risultati ottenuti dai test, che garantiscono la vera funzionalità dell'applicazione web, e soprattutto, la consapevolezza, d'aver creato un sistema semplice, che riesce a garantire il proprio funzionamento, con le sole utilità per cui è stato pensato. Nel futuro prossimo sono in programma diverse ampliamenti del sistema, con l'aggiunta di diversi nuovi servizi, ma riservando la semplicità

e lo scopo del progetto iniziale. Nelle prossime release le possibili modifiche da apportare sono, la creazione di un nuovo modulo per l'esportazione delle mappe o in un formato diverso dallo Shapefile oppure in immagine; il supporto per l'upload di altre estensioni di file compressi e di altri formati contenenti dati geospaziali; l'aggiunta di un modulo opzionale per la modifica sia delle features che degli attributi; infine l'aggiunta di una nuova scheda contenente la tabella con gli attributi di ogni singola feature.

Dopo aver conosciuto questo nuovo mondo, spero che un giorno, con l'ausilio delle informazioni oggi acquisite, riuscirò a creare un progetto simile, che visualizzi e fornisca, i migliori punti di interesse turistico, conosciuti in precedenza solo dagli abitanti del luogo prescelto. In conclusione la redazione di questa tesi sarà di utilità per tutti gli utenti, che avessero deciso di abbandonare servizi già presenti sul web con delle funzionalità instabili, per incominciare l'utilizzo di un servizio stabile, sempre disponibile e con le funzionalità necessarie per una corretta visualizzazione dei propri open data.

Bibliografia

- [1] Cosa sono gli Open Data, Ultima Visita: 23 ottobre 2013, Reperibile al sito: http://www.datiopen.it/it/cosa_sono_gli_open_data
- [2] OpenGovernment: in italia si comincia, Autore: Fabio Deotto Data: 13 settembre 2010, Reperibile al sito: <http://daily.wired.it/news/politica/open-government-in-italia-si-comincia.html>
- [3] Mappa delle Leggi Regionali in Materia di Open Data, Autore: Ernesto Belisario Data: 9 marzo 2012, Reperibile al sito: <http://blog.ernestobelisario.eu/2012/03/09/mappa-delle-leggi-regionali-in-materia-di-open-data/>
- [4] Direttiva 2003/98/CE, Ultima Visita: 28 ottobre 2013, Reperibile al sito: <http://dati.emilia-romagna.it/normativa.html>
- [5] TED 2009 Tim Barners-Lee e il Web prossimo venturo, Autore: Tim Barners Lee Data: febbraio 2009, Reperibile al sito: http://www.ted.com/talks/lang/it/tim-berners_lee_on_the_next_web.html
- [6] Direttiva OpenGovernment USA, Autore: Barack Obama Data: 8 dicembre 2009, Reperibile al sito: www.whitehouse.gov/omb/assets/.../m10-06.pdf
- [7] OpenStreetMap, Ultima Visita: 28 ottobre, Reperibile al sito: <http://www.gfoss.it/drupal/osm>

-
- [8] Voglio Capire Di Più, Ultima Visita: 28 ottobre, Reperibile al sito: <http://www.dati.gov.it/content/voglio-capire>
- [9] ShapeFile Technical Description, Data: luglio 1998, Reperibile al sito: <http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>
- [10] Inside ArcView GIS 8.3, Autore: Scott Hutchinson Data: 2004.
- [11] Principles of geographical information systems for land resource assessment, Autore: Burrough P.A, Data: 1998.
- [12] Fabio Famoso , Ultima Visita: 28 ottobre 2013, Reperibile al sito: www.fscpo.unict.it/ssp/Dispense/Geografia-Economico.../GIS.pdf
- [13] Render Local ShapeFile , Ultima Visita: 28 ottobre 2013, Reperibile al sito: <http://www.prodevelop.es/es/blog/11/12/11/rendering-local-shapefiles-html5>
- [14] Rashad's Blog ShapeFile , Data: 29 maggio 2012, Reperibile al sito: <http://bugbrains.blogspot.it/2012/05/lsiviewer-free-online-geospatial-data.html>
- [15] Open Data, Ultima Visita: 28 ottobre 2013, Reperibile al sito: http://it.wikipedia.org/wiki/Dati_aperti
- [16] Wiki OpenStreetMap, Ultima Modifica: 13 ottobre 2013, Reperibile al sito: http://wiki.openstreetmap.org/wiki/IT:Pagina_Principale.
- [17] Webzine Sanità Pubblica Veterinaria: Numero 77, Aprile 2013 [<http://spvet.it/>] ISSN 1592-1581
- [18] What is SVG?, Ultima Visita: 22 ottobre 2013, Reperibile al sito: <http://www.w3.org/Graphics/SVG/>.
- [19] ShapeFile Bologna, Ultima Visita: 22 ottobre 2013, Reperibile al sito: <http://dati.comune.bologna.it/node/195>.

-
- [20] ShapeFile Milano, Ultima Visita: 22 ottobre 2013, Reperibile al sito: <http://dati.comune.milano.it/dato/item/60-60-piste-ciclabili.html>.
- [21] ShapeFile Natural Earth, Ultima Visita: 22 ottobre 2013, Reperibile al sito: <http://www.naturalearthdata.com>
- [22] Big Data Internet Festival 2012, Data Svolgimento: 05 ottobre 2012, Reperibile al sito: <http://www.slideshare.net/lorenzobenussi/benussi-open-datainternetfestival3>
- [23] Open Data Diffusi In Italia, Ultima Visita: 23 ottobre 2013, Reperibile al sito: <http://www.dati.gov.it/content/infografica>
- [24] Utente: Mohang, Data: 09 febbraio 2011, Reperibile al sito: <http://gis.stackexchange.com/questions/5962/how-to-open-the-shape-files-and-extract-the-data-from-these-files>
- [25] Utente: Nathan, Data: 30 agosto 2013, Reperibile al sito: <http://gis.stackexchange.com/questions/70244/opening-shp-file-under-add-vector-layer>
- [26] Utente: Forester7, Data: 30 settembre 2013, Reperibile al sito: <http://gis.stackexchange.com/questions/5962/how-to-open-the-shape-files-and-extract-the-data-from-these-files>
- [27] Precise Pangoline Release, Data Ultima Modifica: 20 settembre 2013, Reperibile al sito: <https://wiki.ubuntu.com/PrecisePangolin/ReleaseSchedule>
- [28] Apache Server 2.0, Data Ultima Visita: 26 ottobre 2013, Reperibile al sito: http://httpd.apache.org/docs/2.0/new_features_2.0.html
- [29] OpenLayer, Data Ultima Visita: 26 ottobre 2013, Reperibile al sito: <http://openlayers.org/>
- [30] ExtJs 4.1 Documentation, Data Ultima Visita: 26 ottobre 2013, Reperibile al sito: <http://docs.sencha.com/extjs/4.1.3/>

- [31] GeoExt2, Data Ultima Visita: 26 ottobre 2013, Reperibile al sito:
<http://geoext.github.io/geoext2/>

- [32] GDAL, Data Ultima Visita: 26 ottobre 2013, Reperibile al sito:
<http://www.gdal.org/index.html>

- [33] Ogr2ogr, Data Ultima Visita: 26 ottobre 2013, Reperibile al sito:
<http://www.gdal.org/ogr2ogr.html>