

Sperimentazioni di soluzioni geo-database Open Source per la gestione e la distribuzione di informazioni georeferenziate

Battista BENCIOLINI, Marco CIOLLI, Alfonso VITTI, Paolo ZATELLI, Fabio ZOTTELE

Dipartimento di Ingegneria Civile e Ambientale
Università di Trento
Via Mesiano 77, 38100 Trento, Italy
tel. +39 0461 882618, fax +39 0461 882672
e-mail paolo.zatelli@ing.unitn.it

Riassunto

La creazione e la gestione di archivi centralizzati di dati geografici offre molti vantaggi, ma la sua realizzazione può presentare alcune difficoltà. In particolare, l'interoperabilità tra il server dati e i client di diversa natura può essere una significativa fonte di problematiche. I software Free ed Open Source offrono una serie di soluzioni che possono essere proficuamente utilizzate sia per realizzare archivi centralizzati di dati geografici sia per accedervi. Deve, comunque, essere testata l'effettiva interoperabilità dei vari software. Questo articolo riporta e descrive una serie di test di connessione tra vari client GIS commerciali e un database spaziale relazionale Open Source.

Abstract

The creation and management of centralised repository for GIS data offer many advantages but pose some problems. In particular, the interoperability of the server with a mixed set of clients is troublesome. Free and Open source Software (FOSS) provide an appealing solution to this problem, but real usability must be checked. This paper describes the setup of a spatial data base management system and reports a series of tests of connection of various GIS client

Introduzione

I dati geografici sono oggi diffusi sia nell'ambito della ricerca sia in quello della moderna progettazione ambientale. Da ciò si deduce che devono essere messe in atto una serie di pratiche per la buona gestione e per la diffusione dei dati. Infatti, mentre fino a pochi anni fa le informazioni geografiche venivano usate da piccoli e specifici gruppi di lavoro all'interno di una più vasta organizzazione, oggi queste informazioni vengono largamente utilizzate in molti, se non tutti, i livelli. Quando l'uso dei dati geografici è limitato, la loro distribuzione avviene attraverso formati di file standardizzati che vengono accettati da tutti i sistemi coinvolti nell'elaborazione delle informazioni all'interno di un progetto. Quando, invece, l'uso diventa più diffuso questo metodo di gestione presenta molti svantaggi in quanto:

1. il numero di software per la gestione e l'elaborazione dei dati cresce significativamente;
2. cresce il numero e la tipologia degli utilizzatori, la maggioranza dei quali non sono specialisti dei GIS;
3. deve essere messa a punto una forma di controllo sul flusso dei dati.

In particolare, quando la mole dei dati geografici diviene considerevole devono essere messe a punto delle procedure per la loro validazione, il loro controllo e la loro diffusione, controllandone l'accesso da parte degli utenti e da parte dei fornitori i quali, inevitabilmente, avranno privilegi differenti. Inoltre deve essere garantita la consistenza e l'integrità dei dati.

Il modo migliore di soddisfare queste necessità è usare un database centralizzato (DBMS), dove

sono immagazzinate e gestite tutte le informazioni, sia geometriche che semantiche.

L'approccio del database con estensione spaziale

Lo stato dell'arte per la gestione integrata dell'informazione geografica è l'uso di un database con estensione spaziale: esso offre molti vantaggi dal punto di vista di flessibilità, affidabilità e di controllo totale del flusso delle informazioni. Alcuni dei database relazionali più maturi nel loro sviluppo, offrono la possibilità di immagazzinare non solo tipologie tradizionali di dati quali le informazioni semantiche (alfanumeriche), ma anche informazioni geometriche e topologiche. Esistono all'uopo tipi di dato e funzioni che operano su di essi: questo set di tecnologie viene solitamente chiamato *estensione spaziale* e consente la gestione integrata sia del dato geografico sia di quello semantico combinando l'efficienza gestionale di un DBMS con l'abilità di un GIS nell'operare con le entità geometriche e le topologiche. Ci sono indubbi vantaggi in questo approccio dovuti principalmente al dovere gestire un'unica banca dati centralizzata:

- ❑ client differenti possono accedere contemporaneamente ai dati con differenti privilegi, aumentando la sicurezza generale del sistema;
- ❑ possono essere sviluppate interfacce semplici, come WebGIS o DB-browser, per gli utenti non esperti;
- ❑ gli aggiornamenti dei dati nel database possono essere disponibili agli utenti in tempo reale: in questo modo si evitano duplicazioni di dati;
- ❑ si può controllare l'aggiornamento delle informazioni, garantendo il privilegio di scrittura solamente ai tecnici autorizzati;
- ❑ il sistema può validare automaticamente la geometria e la topologia.

Questo tipo di approccio viene definito *approccio client-server*: è possibile accedere ad un database sia localmente, sia in remoto utilizzando vari tipi di client, lavorando su differenti set di dati e con privilegi differenti. L'accesso al server può avvenire con software che possono differire sia per il tipo di dati ai quali accedono che per le capacità di analisi ed elaborazione. Si va dal visualizzatore di dati geografici, spesso distribuito gratuitamente come versione *light* dei pacchetti GIS commerciali, alle *suite GIS* complete, ai visualizzatori del contenuto dei DB, fino ai WebGIS. Ognuna di queste opportunità deve essere configurata per accedere ai dati e per visualizzarli.

Inoltre, per gli utenti occasionali, senza particolari competenze tecniche riguardanti i GIS, possono essere allestite interfacce addizionali. Si possono allora realizzare dei WebGIS interfacciati al database spaziale centralizzato e specializzati a seconda della tipologia di utente. Questi sistemi permettono l'accesso simultaneo sia alla parte geometrica che alla parte semantica delle tabelle: talvolta, comunque, all'utente interessa solamente la parte semantica. Esistono anche DB-browser che permettono la pubblicazione delle tabelle del database via Web.

Con il crescere del volume delle informazioni cresce la complessità nella gestione del sistema: uno dei problemi principali è garantire la congruenza dei dati ed il loro aggiornamento tenendo in conto l'accesso concorrentiale tra differenti utenti: questo può essere garantito proprio dall'uso di un'unica banca dati centralizzata nella quale avviene l'accesso diretto evitando trasferimenti asincroni di informazione dai fornitori delle informazioni geografiche all'utente finale. Gli aggiornamenti e l'upload dei dati saranno, in questo modo, istantaneamente disponibili a tutti con il presupposto che solamente gli utenti autorizzati possono di fatto compiere tali operazioni.

La soluzione Open Source

Tutte le maggiori *software house* che sviluppano GIS forniscono DBMS con estensione spaziale: in quest'ottica, però DBMS proprietari *preferiscono* interfacciarsi con GIS della stessa casa madre specialmente nel caso in cui non vengano usati degli standard sui tipi di dati e sulle transazioni con

il database. Molti GIS dichiarano la piena, o la loro parziale, compatibilità con DBMS spaziali delle principali ditte. I test effettuati, comunque, dimostrano che questa compatibilità non sempre trova riscontro nella realtà. Inoltre, i software proprietari supportano solamente pochi DBMS esterni e, poiché il codice sorgente dei programmi non è disponibile, è praticamente impossibile sviluppare interfacce per i DBMS non supportati.

Mentre i dati semantici immagazzinati in database remoti sono solitamente accessibili attraverso ODBC (con una considerevole perdita di efficienza), un simile protocollo di accesso ai dati geometrici e topologici non è attualmente disponibile.

I software Open Source (FOSS) sono un'alternativa ai sistemi proprietari per una serie di motivi: i sistemi FOSS per la gestione di dati geografici offrono una buona interoperabilità e aderenza agli standard OGC (OpenGeospatial Consortium). Inoltre, sviluppare un ambiente che sfrutti la sinergia tra molti software Open Source specializzati è, solitamente, una procedura abbastanza semplice per coloro che ne abbiano le competenze. Per ultimo, disporre del codice sorgente rende possibile sviluppare interfacce verso altri sistemi, benché questa possibilità risulti più attuabile nel caso in cui i software in questione siano tutti Open Source.

Esistono software FOSS per ogni tipologia di elaborazione, gestione e distribuzione delle informazioni georeferenziate, benché con diversi gradi di maturità dello sviluppo.

L'architettura software: il server

Un database spaziale deve essere affidabile, flessibile, scalabile e performante. L'affidabilità è una condizione necessaria in quanto i client devono affidarsi al server DBMS per una gran quantità di processi e un *crash* non sarebbe ammissibile. Poiché, inoltre, molti client dovranno interfacciarsi contemporaneamente al server, deve essere garantita la velocità di elaborazione dei dati. Nel caso di dati geometrici, inoltre, il volume delle informazioni diventa ben presto elevato. La flessibilità è richiesta in quanto le tipologie di client sono molto differenti ed in rapida evoluzione. L'affidabilità può essere garantita sia da una corretta configurazione hardware sia dalla scelta di un'architettura software modulare: la modularità può inoltre migliorare la flessibilità poiché gli elementi possono essere aggiunti, rimossi o aggiornati senza interruzione del servizio.

Il server è stato allestito esclusivamente con software FOSS, come riportato in tabella 1.

<i>software</i>	<i>Versione</i>	<i>Descrizione</i>
Debian GNU	3.1	sistema operativo
PostgreSQL	7.4.7	DBMS
PostGIS	1.0	estensione spaziale al DBMS
GEOS	2.1.1	libreria per la gestione della topologia
Proj4	4.4.9	sistema di gestione dei Datum
GDAL/OGR	1.2.6	libreria per l'interpretazione dei formati GIS
Apache	2.0.54	Web server
php	4.3.10	linguaggio di sviluppo di pagine web dinamiche
phpPgAdmin	3.5.2	interfaccia web al DBMS
Mapserver	4.6.2	webGIS
Chameleon	2.2	ambiente di sviluppo per i servizi di web mapping

Tabella 1: software installati sul server

Il sistema operativo Debian/Linux è stato scelto per la sua comprovata stabilità ed affidabilità.

Inoltre il sistema di aggiornamento avanzato del software rende semplice e configurabile l'upgrade del sistema senza interrompere il servizio.

Il cuore del servizio è PostgreSQL, un RDBMS relazionale, che aderisce agli standard SQL, al protocollo ACID, scalabile, molto efficiente nel gestire grandi quantità di dati e nella definizione e gestione delle gerarchie di accesso ai dati. L'estensione spaziale a PostgreSQL, PostGIS, fornisce un set di dati geometrici e le funzioni per lavorare con essi. PostGIS è conforme agli standard dell'Open Geospatial Consortium. Le funzionalità topologiche sono gestite dalla libreria GEOS.

Con questa configurazione l'accesso ai dati avviene direttamente connettendosi al database o, solamente per la parte semantica, attraverso ODBC. Un modo efficiente per pubblicare i dati è appoggiarsi ad un WebServer e visualizzare le informazioni in formato HTML dinamico. A questo scopo si è installato il web server Apache che, grazie alla maturità del programma, garantisce un'elevatissima efficienza e sicurezza. Il link tra il DBMS ed Apache è dato dal linguaggio di scripting PHP ed in particolare dal client phpPgAdmin, un front-end al database consultabile via web.

I dati geometrici sono pubblicati usando il webGIS Mapserver: esso costruisce mappe e tematismi partendo dai dati grezzi di PostGIS per via dell'adesione di Mapserver agli standard OGC. Il programma fornisce, inoltre, un elevato numero di funzionalità per visualizzare e interrogare interattivamente le mappe create.

L'architettura software: il Client

Lo scopo principale dello studio è testare l'effettiva interoperabilità tra un database spaziale FOSS e i più diffusi GIS. L'interoperabilità può avvenire a differenti livelli: accesso ai dati semantici e geometrici (e topologici) o solamente alla parte semantica, l'accesso può avvenire in lettura e scrittura o in sola lettura.

La lista dei GIS testati è riportata in tabella 2: nessuno di questi è un software FOSS. Alcuni di questi client hanno una struttura di tipo modulare e la loro connettività dipende dalla presenza o meno dei moduli a ciò dedicati. Inoltre, è possibile l'utilizzo di strumenti di connettività esterna, ad esempio FME della Safe Software.

<i>software</i>	<i>version</i>	<i>vendor</i>	<i>web site</i>
ArcView	3.2	ESRI inc.	www.esri.com
ArcGIS	8 & 9	ESRI inc.	www.esri.com
Geomedia	5.0.2	InterGraph Inc.	www.intergraph.com
AutoDesk land	2005	AutoDesk Inc.	www.autodesk.com
MapInfo	6.0	MapInfo Corporation	www.mapinfo.com

Tabella 2: client testati

Attualmente, ogni software GIS ha un differente protocollo per connettersi ad un database spaziale. Tale protocollo, per i prodotti che hanno un mercato specifico e limitato è orientato verso prodotti *di fatto* standard come Oracle, mentre le compagnie *leader* cercano di imporre i loro standard creando estensioni spaziali legate ai loro prodotti GIS.

I client GIS sono stati installati con la configurazione più completa possibile, controllando la presenza dei *tool* di connettività forniti dai venditori. Il sistema operativo installato sulle macchine remote è Windows 2000.

Test

Test di connettività sono stati condotti caricando nel database spaziale alcuni tematismi e tentando

di accedere ad essi con un GIS installato su di una macchina remota. L'accesso ai dati è stato effettuato a livelli differenti:

1. dati semantici (attributi)
2. dati semantici e geocoding di punti utilizzando due colonne del database come coordinate (le coordinate sono state estrapolate dalla colonna contenente le geometrie utilizzando le funzionalità dell'estensione spaziale)
3. accesso totale sia ai dati geometrici che ai loro attributi

Su qualsiasi macchina remota era possibile visualizzare e modificare (se era concesso il privilegio) le tabelle attraverso l'interfaccia web phpPgAdmin (installata sul server e visualizzabile solamente con un browser web). Tutti i client GIS sono stati installati sulla medesima macchina remota: non esistono differenze riguardo la disponibilità dei driver o sulle impostazioni della connessione.

I dati immagazzinati nel database centralizzato consistevano in tabelle suddivise per ogni tipologia di dati vettoriali e per complessità topologica. Condizione necessaria per un corretto funzionamento dei test è stata la validazione della topologia dei dati in ingresso: per questa operazione ci si è avvalsi di un software Open Source: JUMP.

Tutti i client hanno dimostrato la capacità di connettersi a tutte le tabelle nel database mediante protocollo ODBC. La parte geometrica (in formato standard WKB) non è stata, però, interpretata. Inoltre ArcGIS, ArcView e Geomedia non sono riusciti a modificare le tabelle (pur avendo il privilegio di scrittura).

<i>software</i>	<i>dati semantici</i>	<i>Geometrie</i>	<i>geocoding</i>	<i>add-ons specifici</i>
ArcView	direttamente	sola lettura con pgArc	sì	pgArc
ArcGIS	direttamente	sola lettura con pgArc	sì	pgArc
Geomedia	via ODBC	No	sì	-
AutoDesk Land	via ODBC	No	sì	-
MapInfo	via ODBC	No	sì	-

Tabella 3: risultati delle prove

Il driver ODBC (con licenza GPL) deve essere installato sulla macchina dove risiedono i client che mostrano, comunque, un diverso comportamento nel loro processo di modifica dei dati: alcuni richiedono un *commit* esplicito, altri modificano senza chiedere conferma, alcuni permettono di impostare la modalità di modifica.

I risultati dei test sono mostrati nella tabella 3: la maggior parte dei client riescono ad effettuare il *geocoding*.

Nessuno dei client è riuscito ad accedere direttamente al DBMS, sia per la parte attributiva che per quella geometrica: la connettività può essere migliorata utilizzando software aggiuntivo. Questi *add-ons* possono essere sia degli interpreti generali tra il database ed il client (FME), o specifici per ogni software GIS. In particolare pgArc, un progetto Open Source, consente la connessione diretta di ArcGIS a PostGIS, consente la lettura diretta dei dati geometrici e semantici (eccetto per il tipo MULTIPOINT che non è supportato correttamente), ma non è possibile modificare i campi delle tabelle. Questa estensione, scritta per ArcGIS 8.1 deve essere modificata per funzionare correttamente con ArcGIS 9.

Conclusioni

A breve l'utilizzo di database spaziali sarà inevitabile ogniqualvolta i dati geografici dovranno

essere utilizzati contemporaneamente da molti utenti che si interfacceranno al sistema con client differenti. Oggigiorno, però, non esistono soluzioni che garantiscano la piena interoperabilità.

I test hanno dimostrato che esiste ancora un considerevole margine tra la piena interoperabilità e le opportunità dichiarate dai software.

Lo sviluppo futuro potrà indirizzarsi verso due possibili scenari: la messa a punto di un driver specifico tra PostgreSQL/PostGIS ed ogni client oppure lo sviluppo di un traduttore in tempo reale tra i dati in formato OracleSpatial e PostGIS in quanto, di fatto già esistono i driver Oracle per la maggior parte dei GIS commerciali. Un traduttore tra PostgreSQL e Oracle è in fase di sviluppo, ma solamente per la parte semantica. In questo ambito il mondo Open Source sta proponendo nuove soluzioni, ad esempio ZigGIS che dovrebbero permettere la connessione tra PostgreSQL-PostGIS ed ArcGIS.

Mentre i principali software GIS si focalizzano nel creare suite di programmi che coprono tutti gli aspetti della gestione dei dati geografici, l'adozione di formati standard risulta ancora lenta. I sistemi FOSS, da questo punto di vista, presentano indiscutibili vantaggi proprio grazie alla loro naturale preferenza nell'adottare standard condivisi ed aperti.

In quest'ottica il software Open Source risulta essere l'ambiente naturale per lo sviluppo di ambienti integrati ed interoperanti sia nell'ambito della ricerca che in quello delle applicazioni ambientali.

Ringraziamenti

Il lavoro è stato parzialmente sponsorizzato dal Ministero dell'Università e della Ricerca (MIUR) all'interno del progetto MIUR-COFIN 2004-Strutture evolute della cartografia numerica per i GIS e l'ambiente Web.

Riferimenti bibliografici

Valée R. (2004), "SIG et SGBD : Etude sur l'interconnexion et l'interopérabilité de Oracle 9i et de PostgreSQL",

http://www.carto.net/papers/romain_valee/romain_valee_interoperabilite_oracle9i_PostgreSQL?070704

ESRI (1998), "ESRI Shapefile Technical Description - An ESRI White Paper – July 1998",

<http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>

OpenGeospatial Consortium (1999), "OpenGIS Simple Features Specification for SQL",

<http://www.opengis.org/docs/99-049.pdf>

Lake R. (2006), "The Future of Interoperability",

<http://www.geoplance.com/gw/2004/0405/0405int.asp>

Smith S., Dangermond J. (2004), "On standards and Interoperability",

http://www10.giscafe.com/nbc/articles/view_weekly.php?newsletter=1&run_date=30-Aug-2004

Buehler K. (2004), "GIS Technology Designed for User benefit, ESRI White Paper",

http://www10.giscafe.com/nbc/articles/view_weekly.php?newsletter=1&run_date=30-Aug-2004

Betancourt T.L., Wilhelmi O. (2003), "Final Report on the GIS Demonstration Project, National Center for Atmospheric Research, USA", <http://www.gis.ucar.edu/FinalReport.pdf>