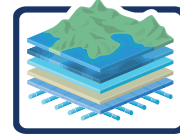




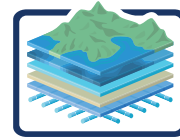
POLITECNICO
DI MILANO



SpatialDBgroup

GeoUML Methodology e Tools Organizzazione Complessiva

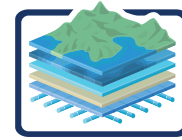
9 settembre 2011



INDICE

1. INTRODUZIONE
 - 1.1 Obiettivi
 - 1.2 Applicazioni
 - 1.3 Evoluzioni e rilasci Software
 - 1.4 Documentazione
 - 1.5 Notazione utilizzata nelle figure
2. SPECIFICHE DI CONTENUTO E GEOUML CATALOGUE
 - 2.1. Specifiche di contenuto
 - 2..2 Funzionalità del GeoUML catalogue
 - 2.3 Versioni Catalogue Viewer e Catalogue Editor
 - 2.4 File di trasferimento specifiche
3. CONFORMITA' ALLA SPECIFICA E GEOUML VALIDATOR
 - 3.1. Conformità Reale e Conformità Intrinseca
 - 3.2. GeoUML Validator
 - 3.3 Implementazione delle proprietà geometriche del GeoUML
4. I MODELLI IMPLEMENTATIVI E DATA PRODUCT SPECIFICATION
 - 4.1. Definizioni
 - 4.2. Data Product Specification (DPS)
 - 4.3. Struttura dei Tools e Modelli Implementativi
 - 4.4. Categorie di Modelli Implementativi

Appendice - RAPPORTO TRA GEOUML METHODOLOGY E GLI STANDARD ISO TC211



1. INTRODUZIONE

1.1 Obiettivi

La metodologia *GeoUML Methodology* e gli strumenti *GeoUML Tools* descritti in questo documento sono stati sviluppati con la finalità di supportare la gestione di uno *Schema Concettuale* geografico e di realizzare la validazione automatica della conformità di un *Data Product* a un dato Schema Concettuale. Il termine Data Product indica un qualsiasi tipo di organizzazione di dati spaziali: può essere costituito da un insieme di files oppure da un database.

I principi fondamentali alla base di questo sviluppo sono i seguenti:

- adesione agli standard ISO 19100 ove questi sono rilevanti
- essere implementabile con le tecnologie attualmente disponibili
- essere indipendente da qualsiasi specifico prodotto GIS (commerciale o open source)
- mantenere una netta distinzione tra il livello concettuale e quello fisico

Lo strumento che gestisce lo schema concettuale è detto *GeoUML Catalogue*; il modello utilizzato per definire lo schema è detto (modello) *GeoUML*.

Lo strumento utilizzato per controllare se un Data Product è conforme a un dato Schema Concettuale è detto *GeoUML Validator*.

Per trasferire uno Schema Concettuale tra diversi Catalogue oppure dal Catalogue al Validator viene utilizzato un file detto *File di Specifica*.

Il Data Product da validare deve essere implementato utilizzando uno tra diversi *Modelli Implementativi (MI)* predefiniti; un Modello Implementativo definisce le regole per trasformare uno Schema Concettuale in una Struttura Fisica.

I GeoUML Tools sono interamente scritti in linguaggio Java e possono essere utilizzati su qualsiasi piattaforma.

1.2 Applicazioni

La GeoUML Methodology e gli strumenti GeoUML Tools sono usabili per la progettazione e l'implementazione di Database Territoriali in generale, perché permettono:

- di specificare i contenuti di un Database a livello concettuale precisando anche le proprietà topologiche che i dati dovranno soddisfare
- di derivare dalla specifica concettuale non solo la struttura fisica del Database, ma anche le corrispondenti strutture fisiche di file in formato Shapefile o GML utilizzabili per i diversi scopi collegati al Database, ad esempio: file di fornitura, file di distribuzione dei dati agli utenti, file per scambiare aggiornamenti, ecc...
- di verificare la conformità di un Data Product alle specifiche

Un ambito nel quale la GeoUML Methodology può risultare particolarmente utile è costituito da una SDI a collaborazione stretta. Una SDI a collaborazione stretta è una Infrastruttura di Dati Territoriali (IDT) nella quale diversi Database Territoriali Locali sono gestiti indipendentemente da diverse Autorità Locali, ma nell'insieme questi database devono rappresentare *un modello integrato del territorio complessivo*. Una difficoltà particolare in questo contesto è costituita dalle diverse soluzioni tecniche adottate dalle diverse Autorità Locali per memorizzare fisicamente i loro database. Per questo motivo il contenuto informativo condiviso e i vincoli di integrità spaziale devono essere definiti a livello concettuale, ma devono poter essere validati su database realizzati secondo diversi modelli implementativi.

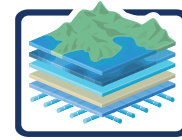
1.3 Evoluzioni e rilasci Software

La GeoUML Methodology è soggetta a evoluzioni per adattarla sempre meglio alle esigenze che emergono dalla effettiva utilizzazione in situazioni reali; questo documento si riferisce agli aspetti implementati dal *rilascio consolidato* del software, disponibile (tramite download) per l'utilizzazione da parte degli utenti autorizzati; altri aspetti, che sono stati ancora solamente oggetto di studio o di prototipazione sperimentale ed entreranno eventualmente in versioni future del software, non sono trattati.

Il meccanismo di numerazione delle versioni del software è legato alle evoluzioni dei singoli strumenti e a quella della struttura del file di specifica. La struttura generale del numero di versione è la seguente x.y, dove:

- x indica l'evoluzione del file di specifica, cioè x cambia quando viene cambiata la struttura di questo file
- y indica l'evoluzione del singolo strumento software (Catalogue o Validator)

Importante: dato che il file di specifica è prodotto dal Catalogue, la versione del file di specifica (cioè della sua struttura) è determinata dal valore x della versione del Catalogue che lo ha prodotto. Il Validator per leggere un file di specifica deve avere una versione con lo stesso x.



Per poter utilizzare specifiche esistenti nonostante l'evoluzione del formato del file di specifica sarà necessario garantire la compatibilità all'indietro del Catalogue rispetto a x, cioè che il Catalogue con versione x sia in grado di leggere una specifica con versione x inferiore.

Il rilascio consolidato del software alla quale si fa riferimento in questo documento (riportato anche sulla copertina) è il seguente

- GeoUML Catalogue r. 1.?
- GeoUML Validator r. 1.?

1.4 Documentazione (!!!questa sezione è più un piano di lavoro, soggetto ancora a precisazioni sugli accordi col CISIS, che una documentazione dello stato!!!)

La documentazione della GeoUML Methodology è costituita da due tipi di documenti:

- Specifiche: sono documenti nei quali è considerata prioritaria la completezza e la precisione delle definizioni, non sono orientati a spiegare le modalità operative:
 - *Il modello GeoUML (versione 2008)*
 - *Il Modello Implementativo Shape_Flat*
 - *Il Modello Implementativo Shape_Topo*
 - *Il Modello Implementativo ESF_SQL_Oracle (multigeometria)*
 - *Il Modello Implementativo ESF_SQL_PostGIS (multigeometria)*
 - *Il Modello Implementativo ESF_SQL_Oracle (monogeometria)*
 - *Il Modello Implementativo ESF_SQL_PostGIS (monogeometria)(implementazione da concordare)*
 - *Il Modello Implementativo ESF_GML*
- Guide (per gli utenti): sono orientati a spiegare come funzionano la metodologia in generale e gli strumenti; le guide sono:
 - *GeoUML Methodology e Tools - Organizzazione Complessiva (questo documento)*
 - *Guida all'uso del GeoUML Catalogue*
 - *Guida all'uso del GeoUML Validator*
 - *Guida ai Modelli Implementativi di tipo Flat*
 - *Guida alla implementazione delle proprietà geometriche del GeoUML*

1.5 Notazione utilizzata nelle figure

La maggior parte delle figure rappresentano flussi di informazione tra componenti software. In queste figure viene utilizzata la notazione illustrata in Figura 1.1

La distinzione tra normali documenti e informazione strutturata è fondamentale in questo contesto, perché un obiettivo del GeoUML Catalogue è quello di gestire una Specifica di Contenuto come informazione strutturata e automaticamente processabile, a differenza di quanto avviene tradizionalmente nei capitoli relativi a produzioni di dati territoriali.

Gli strumenti che costituiscono i GeoUML Tools sono usabili in collaborazione con altri strumenti, quindi in alcune figure è necessario distinguere, come indicato in figura 1.1, tra i componenti effettivi dei GeoUML Tools e altri software. In alcuni casi sono state sviluppate delle funzionalità aggiungibili ad altri software, specializzando di fatto i software esterni per la collaborazione con i GeoUML Tools, come mostrato in figura 1.1 (componenti misti).

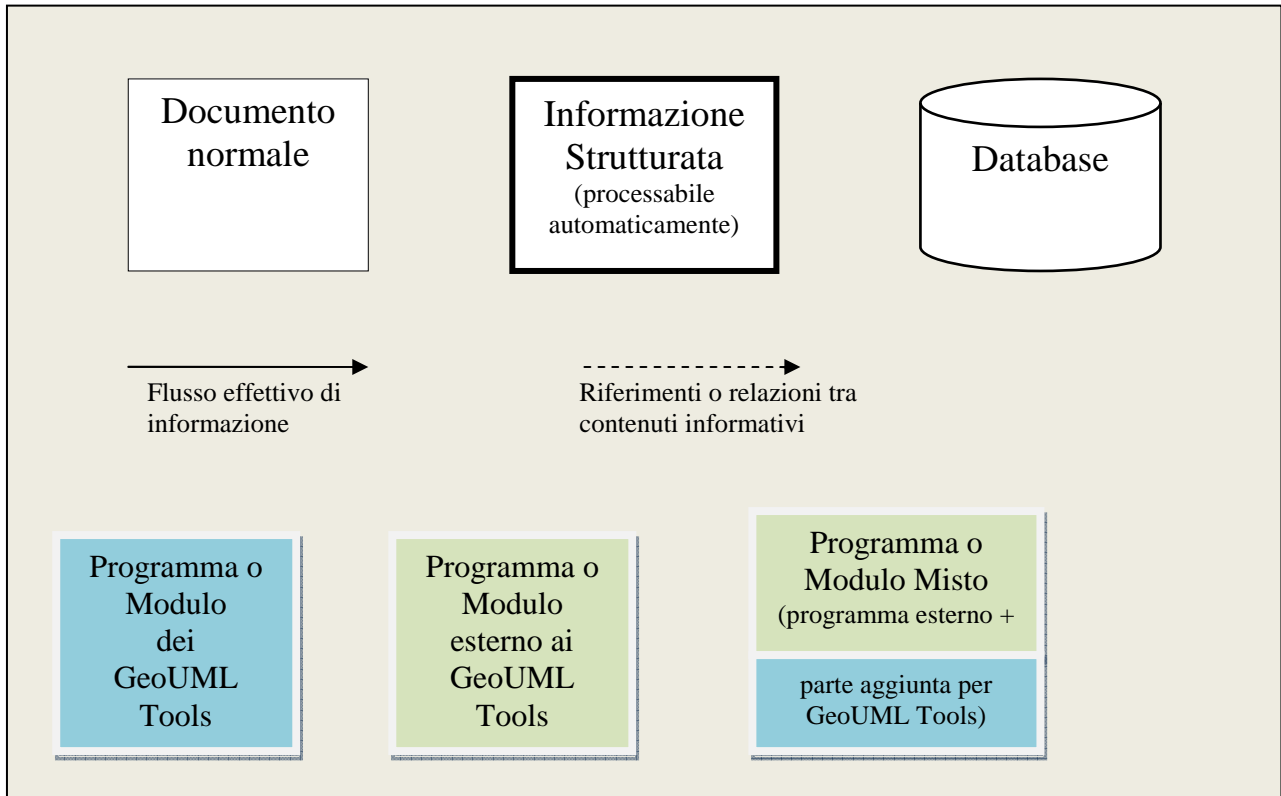
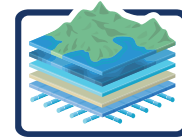


Figura 1.1



2. SPECIFICHE DI CONTENUTO E GEOUML CATALOGUE

2.1 Specifiche di Contenuto

Una Specifica di Contenuto (SC) descrive il contenuto informativo di un Database per mezzo di diversi tipi di definizioni che hanno scopi diversi:

1. definizione degli **elementi informativi**, cioè degli elementi che devono essere rappresentati nel Database, (classi, attributi, attributi geometrici, associazioni, domini) in maniera indipendente dal modo in cui tali elementi sono rappresentati in una particolare tecnologia di memorizzazione dei dati
2. definizione dei **vincoli di integrità**, cioè delle proprietà intrinseche che gli elementi informativi devono soddisfare (con proprietà intrinseche si intendono le proprietà verificabili sugli elementi informativi stessi, senza osservazioni dirette del mondo reale)
3. **elementi descrittivi**: in questa categoria rientrano tutte le informazioni utilizzabili dagli esseri umani per capire come interpretare un contenuto del database in termini di realtà rappresentata e viceversa come interpretare una situazione reale nei termini degli elementi informativi (ad esempio, la modalità di rilievo)

Gli elementi informativi e i vincoli di integrità costituiscono lo **Schema Concettuale**, cioè la parte formalmente strutturata delle specifiche (elaborabile automaticamente).

Le regole utilizzate per definire la parte strutturata delle specifiche costituiscono il **modello GeoUML**; la definizione precisa di tali regole è fornita nel documento "*Il modello GeoUML (versione 2008)*".

2.2 Funzionalità del GeoUML catalogue

Per creare, modificare e visualizzare una Specifica di Contenuto si utilizza un programma applicativo chiamato **GeoUML catalogue**.

GeoUML catalogue possiede essenzialmente le funzionalità per la gestione di una specifica di contenuto, in particolare:

- Funzioni di importazione ed esportazione di una specifica (per Amministratore della Specifica)
- Funzioni di Editing dei contenuti (per utenti abilitati dall'Amministratore)
- Funzioni di Visualizzazione e Ricerca dei contenuti (per utenti generici)
- Funzioni di preparazione e produzione della documentazione standard (per utenti abilitati dall'amministratore)

Il GeoUML Catalogue possiede anche una serie di funzioni orientate alla gestione degli aspetti relativi alla produzione di Schemi Fisici in base ai corrispondenti Modelli Implementativi; queste funzioni sono spiegate nel capitolo 4.

2.3 Versioni Catalogue Viewer e Catalogue Editor

Il GeoUML Catalogue viene distribuito in due versioni:

- il **Catalogue Editor** è completo di tutte le funzioni
- il **Catalogue Viewer** costituisce una versione ridotta che possiede solamente le funzioni di Visualizzazione

2.4 File di trasferimento specifiche

Il GeoUML catalogue memorizza la Specifica Concettuale sulla quale lavora in un suo Database Interno.

Il contenuto del Database Interno può essere esportato/importato in un formato XML il cui schema è pubblicato (XML schema). Il file che definisce tale XML schema si chiama "sc.xsd".

Il file che contiene una SC in tale formato è detto **File di Specifica** ed ha un'estensione di tipo ".scs".

Il File di Specifica permette di trasferire una SC non solo tra diversi GeoUML Catalogue, ma anche dal Catalogue al Validator.

Il File di specifica contiene, oltre alla specifica concettuale SC, anche le informazioni relative agli schemi fisici che sono stati definiti con riferimento a tale SC (vedi capitolo 4).

Riferimenti.

Per utilizzare il Catalogue è necessario comprendere il modello GeoUML, definito nel documento "*Il modello GeoUML (versione 2008)*"

L'uso del Catalogue per gestire una specifica concettuale è descritto nel documento "*Guida all'uso del GeoUML Catalogue*"

N.B. Gran parte di questo documento è dedicata agli aspetti di definizione degli schemi fisici (vedi capitolo 4)

3 CONFORMITA' ALLA SPECIFICA E GEOUML VALIDATOR

3.1 Conformità Reale e Conformità Intrinseca

La conformità di un Data Product DP a una specifica di contenuto SC è composta da due aspetti:

1. **Conformità Reale:** la conformità reale riguarda la corrispondenza tra il contenuto informativo del Dataset e la porzione di Mondo Reale (detta anche **Territorio di Riferimento**) alla quale il Dataset si riferisce; gli elementi informativi e i relativi elementi descrittivi della SC determinano come tale corrispondenza debba essere valutata
2. **Conformità Intrinseca:** la conformità intrinseca riguarda la consistenza dell'informazione contenuta nel Data Product; sia gli elementi informativi che i vincoli della SC sono utilizzati per valutare la conformità intrinseca

La valutazione della qualità di un Data Product deve tener conto sia della Conformità Reale che della Conformità Intrinseca alla Specifica di Contenuto.

Dato che i GeoUML Tools, in particolare il Validator, supportano esclusivamente la verifica della conformità intrinseca (perché solamente quest'ultima può essere valutata automaticamente), in questo documento con il termine conformità senza ulteriori specificazioni si intenderà la sola conformità intrinseca.

Vale la pena di sottolineare che anche le tradizionali operazioni di collaudo, orientate alla verifica della Conformità Reale, possono beneficiare sensibilmente del supporto del Validator, perché alcuni errori dei dati rispetto alla Conformità Intrinseca costituiscono spesso dei segnali di possibili errori di Conformità Reale. Ad esempio, la segnalazione dell'assenza di istanze di una classe può essere un utile avvertimento (Warning) per il collaudatore; la verifica che tale assenza corrisponda alla situazione del mondo reale oppure sia un errore riguarda la Conformità Reale.

Un Data Product DP è intrinsecamente conforme a una Specifica di Contenuto SC se e solo se:

- a. tutti i dati contenuti in DP corrispondono a elementi informativi di SC
- b. i dati contenuti in DP soddisfano tutti i vincoli di integrità definiti in SC

Si osservi che la condizione (a) non implica che per ogni elemento informativo di SC esistano dei dati corrispondenti; ad esempio, se nel Territorio di Riferimento di un Data Product DP non esistono istanze reali di una certa classe C definita in SC, nel Data Product DP non esisteranno istanze di tale classe e ciò non viola la Conformità Intrinseca di DP rispetto a SC.

3.2 GeoUML Validator

GeoUML validator è uno strumento in grado di operare il controllo di conformità intrinseca di un generico Data Product relativamente ad una qualsiasi SC gestita da GeoUML catalogue. Il validator utilizza la specifica esportata dal Catalogue per analizzare il Data Product e fornire delle informazioni diagnostiche (vedi figura 3.1).

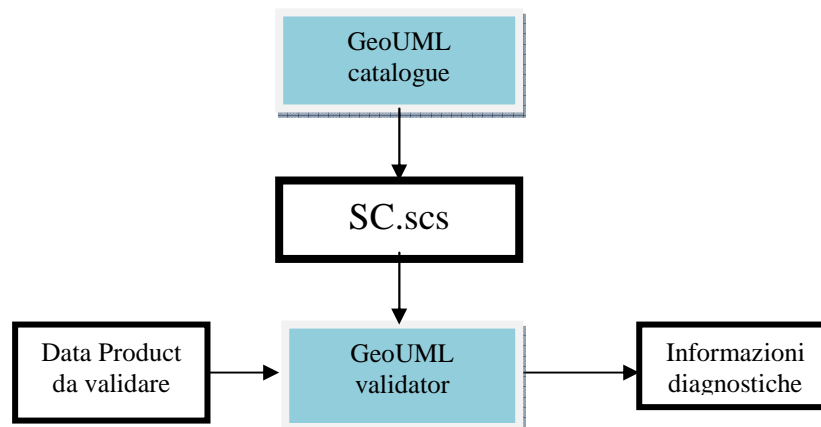


Figura 3.1

La scelta e organizzazione delle informazioni diagnostiche costituisce un problema complesso, per vari motivi:

- per essere utilizzabile, un'informazione diagnostica deve riferirsi agli elementi della specifica (Classi, Associazioni, Attributi nelle varie forme e vincoli) anche se gli errori vengono rilevati sulla struttura fisica del Data Product e anche quest'ultimo riferimento è importante

- l'organizzazione delle informazioni diagnostiche deve essere flessibile perché diversi usi richiedono una diversa organizzazione

Per superare il primo problema il Validator riporta per ogni errore sia l'indicazione dell'oggetto concettuale sul quale l'errore si è verificato, sia il riferimento all'oggetto fisico interessato tra tutti quelli che rappresentano l'oggetto concettuale.

Per superare il secondo problema, dato che la produzione di una reportistica è difficilmente adattabile alle varie esigenze, il Validator produce un **Database di Reportistica** sul quale è possibile operare con diverse modalità in base alle esigenze e alle capacità dell'utente, come mostrato in figura 3.2. Il Database di Reportistica è realizzato nella tecnologia Apache DERBY, open source e quindi usabile liberamente. L'utente deve installare DERBY sulla propria macchina e poi indicare al Validator la Directory in cui creare il Database di Reportistica.

La maggior parte delle informazioni diagnostiche prodotte dal Validator sono corredate di geometria, cioè contengono la geometria dell'oggetto errato. Dato che DERBY non è un Database spaziale, le geometrie sono memorizzate in formato standard WKB in un "binary object (blob)".

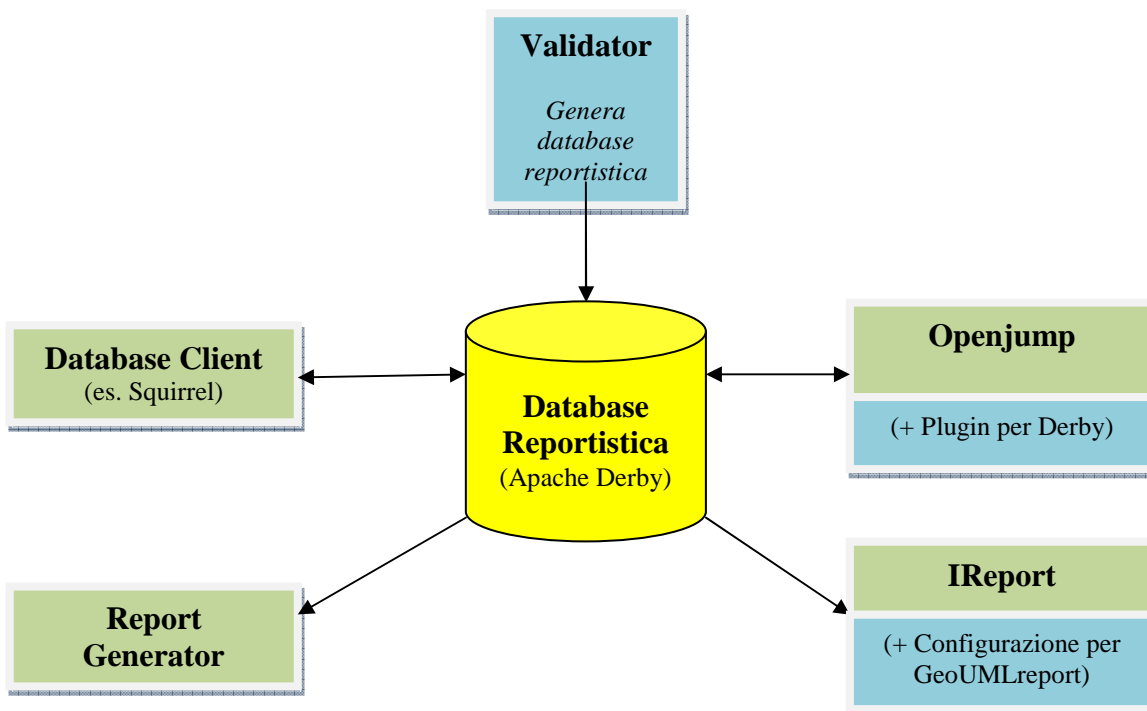
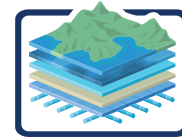


Figura 3.2

I 4 modi indicati in figura per utilizzare il Database di Reportistica sono:

- Uso di un generico Database Client, cioè di uno strumento in grado di interrogare un DB in tecnologia DERBY (esistono numerosi strumenti di questo tipo); questo metodo è il più potente ed è consigliabile per utenti esperti nell'accesso ai database relazionali, però non permette di analizzare direttamente le geometrie (che possono essere analizzate sui dati del Data Product, ad esempio)
- Uso del Client per dati territoriali Openjump, per il quale viene fornito un plugin per accedere DERBY: in questo modo è possibile esplorare i dati geometrici
- Uso di un Report Generator per prodursi dei Report adatti alle proprie esigenze: questo metodo è indicato per chi dovrà ripetere molte volte particolari analisi sulla diagnostica, ma richiede la prima volta un utente capace di configurare il Report Generator
- Uso del Report Generator IReport già preconfigurato per i rapporti standard, detti **GeoUMLreport**; questo è l'uso più semplice, adatto all'utente inesperto, ma non fornisce alcuna flessibilità



3.3 Implementazione delle proprietà geometriche del GeoUML

Le proprietà geometriche del GeoUML sono definite a un livello piuttosto astratto, basato sulla logica e la matematica, e in particolare su una rappresentazione nello spazio euclideo continuo. In tale spazio le coordinate dei punti sono numeri reali.

Nelle implementazioni le coordinate dei punti devono essere rappresentate da numeri finiti e quindi lo spazio delle geometrie è uno spazio discreto. Questo aspetto dell'implementazione introduce una serie di problemi di cui è necessario tenere conto nella creazione di un Data Product che deve essere conforme a una specifica GeoUML.

Il Validator per analizzare il Data Product deve applicare una interpretazione delle geometrie esistenti nello spazio discreto per verificare che soddisfino le proprietà definite a livello concettuale.

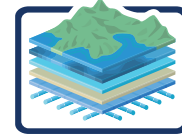
Riferimenti.

Le modalità operative di uso del Validator e la struttura del Database di Reportistica sono descritte nel documento:

“Guida all’uso del GeoUML Validator”

Le regole di interpretazione delle geometrie implementate rispetto ai requisiti concettuali sono trattate nel documento.

“Guida alla implementazione delle proprietà geometriche del GeoUML”



4. MODELLI IMPLEMENTATIVI E DATA PRODUCT SPECIFICATION (DPS)

4.1 Definizioni

Una Specifica di Contenuto a livello concettuale è volutamente astratta, nel senso che non definisce gli aspetti che dipendono dalla tecnologia. Per poter creare effettivamente un Data Product conforme a una SC è quindi necessario indicare le modalità di implementazione, che dipendono a loro volta dalla tecnologia adottata.

Terminologia di base:

1. **Schema Fisico:** è la definizione della struttura fisica del Data Product ; ad esempio, per citare i più comuni formati fisici, un XSD per un file GML, la struttura degli shapefile per un dataset in formato shape, le istruzioni DDL (create table, ecc...) per un Database SQL.
2. **Mapping Concettuale-Fisico (CFmapping):** è la definizione delle corrispondenze esistenti tra gli elementi dello schema fisico e gli elementi della Specifica Concettuale (ad esempio, l'indicazione che una certa classe è rappresentata fisicamente in una certa tabella relazionale o in un certo shapefile)
3. **Modello Implementativo:** è un insieme di regole che permettono di generare automaticamente uno Schema e un Mapping Fisico da una Specifica Concettuale.

Un Modello Implementativo è costituito da un insieme di regole che non dipendono dalla SC alla quale viene applicato; pertanto valgono ambedue le seguenti affermazioni:

- lo stesso Modello Implementativo può essere applicato a diverse Specifiche di Contenuto, producendo diversi Schemi Fisici che condividono gli aspetti legati alla tecnologia, ma si riferiscono a contenuti diversi;
- la stessa Specifica di Contenuto può essere materializzata secondo diversi Modelli Implementativi, ottenendo schemi fisici che rappresentano in maniera diversa gli stessi contenuti.

Le regole di base di un Modello Implementativo sono fisse.

In alcune situazioni risulta opportuno introdurre una certa flessibilità nella generazione del CFmapping; per questo motivo ad alcuni MI sono associati dei meccanismi che permettono di guidare il processo di generazione del CFmapping – tali meccanismi sono di 2 tipi:

- a. **Parametricità:** il processo di generazione del CFmapping può presentare delle possibilità di scelta
- b. **Adattabilità:** permettono di applicare alcune modifiche manuali sul CFmapping generato

In pratica tutti i MI definiti allo stato attuale, anche i più rigidi, possiedono un livello minimo di parametricità.

4.2 Data Product Specification (DPS)

Una DPS è una struttura dati che contiene le indicazioni necessarie al sistema per generare un CFmapping partendo da una SC. Tali indicazioni sono:

- il Modello Implementativo da adottare
- un valore per ognuno dei parametri eventualmente richiesti dal MI specificato

La definizione di una DPS avviene nel GeoUML Catalogue. Nel Catalogue è presente un solo Schema Concettuale (SC), ma possono essere definite diverse DPS (ogni DPS ha un proprio nome).

Sempre nel GeoUML Catalogue è possibile, dopo aver definito una DPS, produrre un CFmapping, che viene anch'esso memorizzato in una struttura dati interna. Sulla base del CFmapping può essere prodotto uno Schema Fisico.

Le operazioni eseguibili nel Catalogue relativamente a questi problemi sono quindi 3 ed hanno le seguenti dipendenze:

1. *definizione di una DPS:* è indipendente, può essere definita in qualsiasi momento e non risente di cambiamenti della SC
2. *generazione del CFmapping* (con eventuali operazioni di adattamento, se il MI prevede l'adattabilità): dipende sia dalla SC che dalla DPS adottata per la generazione, pertanto un cambiamento nella SC o nella DPS adottata invalida il CFmapping
3. *generazione dello Schema Fisico:* dipende dal CFmapping, e quindi viene invalidato se viene cambiato il CFmapping

Le 3 operazioni sono invocabili separatamente per permettere flessibilità nell'uso; infatti:

- l'esecuzione di modifiche sulla SC non richiede di modificare le DPS già definite, ma semplicemente di rigenerare il CFmapping e lo Schema Fisico
- la generazione dello Schema Fisico può essere svolta anche nella versione Viewer del Catalogue, perché è totalmente rigida e non modifica il contenuto del File di Specifica

Nel Catalogue sono memorizzate, oltre alla SC, anche le DPS definite e l'ultimo CFmapping generato, e tali informazioni sono anche esportate nel File di Specifica e quindi rese disponibili al Validator. Dato che per ogni DPS può esistere un solo CFmapping, il nome della DPS identifica anche il CFmapping e quindi tutta l'informazione necessaria, oltre allo SC, per interpretare la struttura di un Data Product.

Se una DPS viene modificata, il relativo CFmapping eventualmente esistente viene cancellato e deve essere rigenerato esplicitamente.

Si noti che lo Schema Fisico non viene memorizzato all'interno del Catalogue e quindi non è contenuto nel File di Specifica. Il formato dello Schema Fisico può variare moltissimo in base al tipo di tecnologia; ad esempio, per un MI orientato ai Database SQL il Catalogue produce le definizioni SQL (create table, ecc...) utilizzabili per creare il Database. L'utente può memorizzare lo schema fisico in maniera opportuna per gli scopi che intende perseguire.

Con il GeoUML Catalogue è possibile creare un *Database del CFMapping* in tecnologia Apache Derby contenente le informazioni che definiscono il CFmapping generato; da tale database è possibile ricavare la documentazione desiderata.

Il procedimento è analogo a quanto visto per il Database di Reportistica generato dal Validator e valgono considerazioni simili per l'accesso e la derivazione della reportistica. Anche in questo caso il sistema Ireport è stato personalizzato per produrre un *GeoUMLreport di Mapping*, che è in generale sufficiente per la maggior parte degli usi.

4.3 Struttura dei Tools e Modelli Implementativi

Dal punto di vista dell'architettura Software dei GeoUML Tools è resa possibile l'estensione sia del Catalogue che del Validator per gestire nuovi Modelli Implementativi. Tale estensione si basa sulla realizzazione di componenti software aggiuntivi, detti "Plugin", che possono essere aggiunti in maniera facile agli strumenti già esistenti.

Per ogni nuovo Modello Implementativo devono essere realizzati due nuovi Plugin:

- un Plugin per il Catalogue, detto *Mapping Generator*
- un Plugin per il Validator, detto *Reader*

In figura 4.1 si mostra un il Validator che legge un Data Product conforme a una specifica SCx e a un Modello Implementativo MIy; il Validator utilizza un Reader per interpretare il modello implementativo MIy.

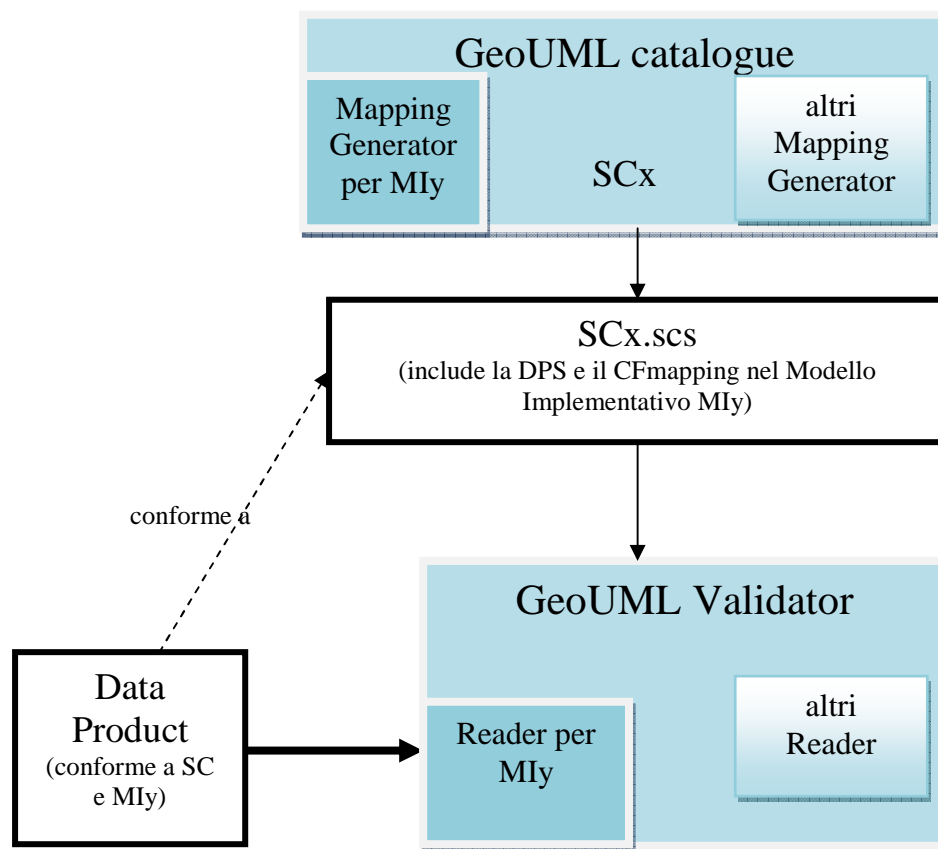
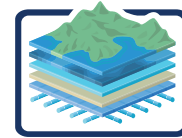


Figura 4.1



4.4 Categorie di Modelli Implementativi

I modelli implementativi dipendono dalla tecnologia alla quale fanno riferimento; al momento le tecnologie di riferimento sono principalmente gli Shapefile, il GML e i Database Georelazionali (spatial SQL).

Conviene classificare i MI anche in base alle funzione tipicamente svolta dai Data Product che li utilizzano:

- **MI di trasferimento** per i Data Product utilizzati per lo scambio di dati. Per i modelli di trasferimento è opportuno utilizzare un MI rigido non adattabile, in quanto il destinatario del trasferimento deve essere in grado di conoscere come il Data Product è strutturato. Tipicamente i modelli di trasferimento sono inoltre basati sulle tecnologie degli Shapefile e del GML.
- **MI di lavoro** per i Data Product utilizzati da un Ente per svolgere le sue attività. Per i modelli di lavoro è invece opportuno utilizzare un MI adattabile, in modo da adattare alcuni aspetti dello schema fisico generato ai requisiti specifici dell'ente che deve utilizzarlo. Tipicamente i modelli di lavoro sono associati alle tecnologie dei database georelazionali.

I principali problemi di mapping che le regole di un MI devono definire sono:

- **Mapping di struttura:** come trasformare la struttura a classi, associazioni, gerarchie, attributi enumerati e gerarchici del GeoUML in corrispondenti strutture della tecnologia di destinazione;
- **Mapping del modello geometrico:** come trasformare i tipi geometrici del GeoUML nei tipi disponibili nella tecnologia di destinazione;
- **Mapping dei costrutti speciali del GeoUML:** come implementare gli attributi a tratti e a sottoaree e le superfici B3D utilizzando i costrutti disponibili nella tecnologia di destinazione.

Il MI non deve invece tradurre i vincoli – non si suppone infatti che i vincoli del GeoUML vengano direttamente implementati nella tecnologia di destinazione; il controllo dei vincoli è delegato, come abbiamo visto, all'esistenza di uno strumento specializzato – il GeoUML Validator.

Per quanto riguarda il Mapping di Struttura una fondamentale categoria di strutture fisiche utilizzabili è costituita dalle strutture *piatte (flat)*, nelle quali il GeoUML si deve mappare su tabelle non nidificate. Pare sensato definire dei MI di tipo flat per le tecnologie GeoRelazionali e per gli Shapefile, mentre il GML supporta naturalmente la struttura articolata propria del (Geo)UML (strutture *nidificate (nested)*).

Per quanto riguarda il Mapping del Modello Geometrico si può osservare che al momento il modello geometrico implementato più diffuso è sicuramente quello basato sullo standard Simple Feature model (SF). Tutti i MI implementativi realizzati applicano tale standard con un'estensione, largamente supportata dai sistemi, per trattare la rappresentazione 3D dei punti e delle linee (Extended Simple Feature – ESF)

Nella seguente figura 4.2 è mostrata una struttura di classificazione dei Modelli Implementativi implementati nell'attuale rilascio dei GeoUML Tools. I rettangoli evidenziati rappresentano i 7 MI implementati, gli altri rettangoli le categorie nelle quali tali MI possono essere raggruppati. La classificazione è multipla, cioè un MI può appartenere a più categorie; ad esempio ESF_GML è strutturalmente di tipo nested e funzionalmente di tipo di trasferimento.

La struttura di classificazione si basa sulla impostazione seguente:

1. La maggior parte dei MI (6 su 7) sono di tipo flat; tale struttura è infatti molto semplice e largamente supportata; a loro volta questi modelli si suddividono secondo diversi principi:
 - 1.1. Esistono 4 MI orientati ai Database; tali MI si distinguono in base ai seguenti aspetti:
 - 1.1.1. I MI SQL multigeometria, nei quali una tabella relazionale può contenere più di un attributo geometrico; di questi esistono 2 versioni in base alla tecnologia (ORACLE o POSTGIS)
 - 1.1.2. I MI SQL monogeometria, nei quali una tabella relazionale può contenere solamente un attributo geometrico – questa limitazione è stata introdotta per supportare gli strumenti GIS che non sono in grado di operare sul modello multigeometria - anche di questi esistono 2 versioni in base alla tecnologia (ORACLE o POSTGIS)
 - 1.2. Esistono 2 MI di trasferimento basati su tecnologia Shape, che si distinguono in base alla rappresentazione della geometria:
 - 1.2.1. SHAPE_FLAT rappresenta la geometria oggetto per oggetto, come tutti gli altri MI classificati come MI a oggetti
 - 1.2.2. SHAPE_TOPO utilizza una strutturazione topologica delle geometrie
2. Un MI, denominato ESF_GML, si basa sulla struttura nested; questo modello segue gli standard ISO 19100 per i file di trasferimento in GML. Dato che il GeoUML è una estensione del UML e che il processo standard previsto dall'ISO per generare uno schema GML consiste nell'applicare un insieme di regole a partire da un Application Schema (AS) definito in UML, il passaggio fondamentale nel MI ESF_GML consiste nel trasformare uno schema GeoUML in un corrispondente AS. Sostanzialmente questo passaggio consiste nella conversione dei costrutti

propri del GeoUML (attributi a tratti e sottoaree, superfici B3D, domini gerarchici) nei costrutti standard di un AS. I passaggi successivi seguono regole dettate dagli standard, anche se non sono totalmente automatici, perché le regole permettono delle scelte.

Riferimenti.

Gli aspetti operativi relativi alla definizione di una DPS e alla generazione del CFmapping e dello Schema Fisico sono trattati nel documento:

“Guida all’uso del GeoUML Catalogue”

Gli aspetti generali relativi alla struttura flat, comune a 6 dei 7 MI, inclusa la struttura del corrispondente database di CFmapping e del GeoUML report di mapping sono trattati del documento:

“Guida ai Modelli Implementativi di tipo flat”

Per ogni singolo MI xxx esiste inoltre un documento di specifica

“Il Modello Implementativo xxx”

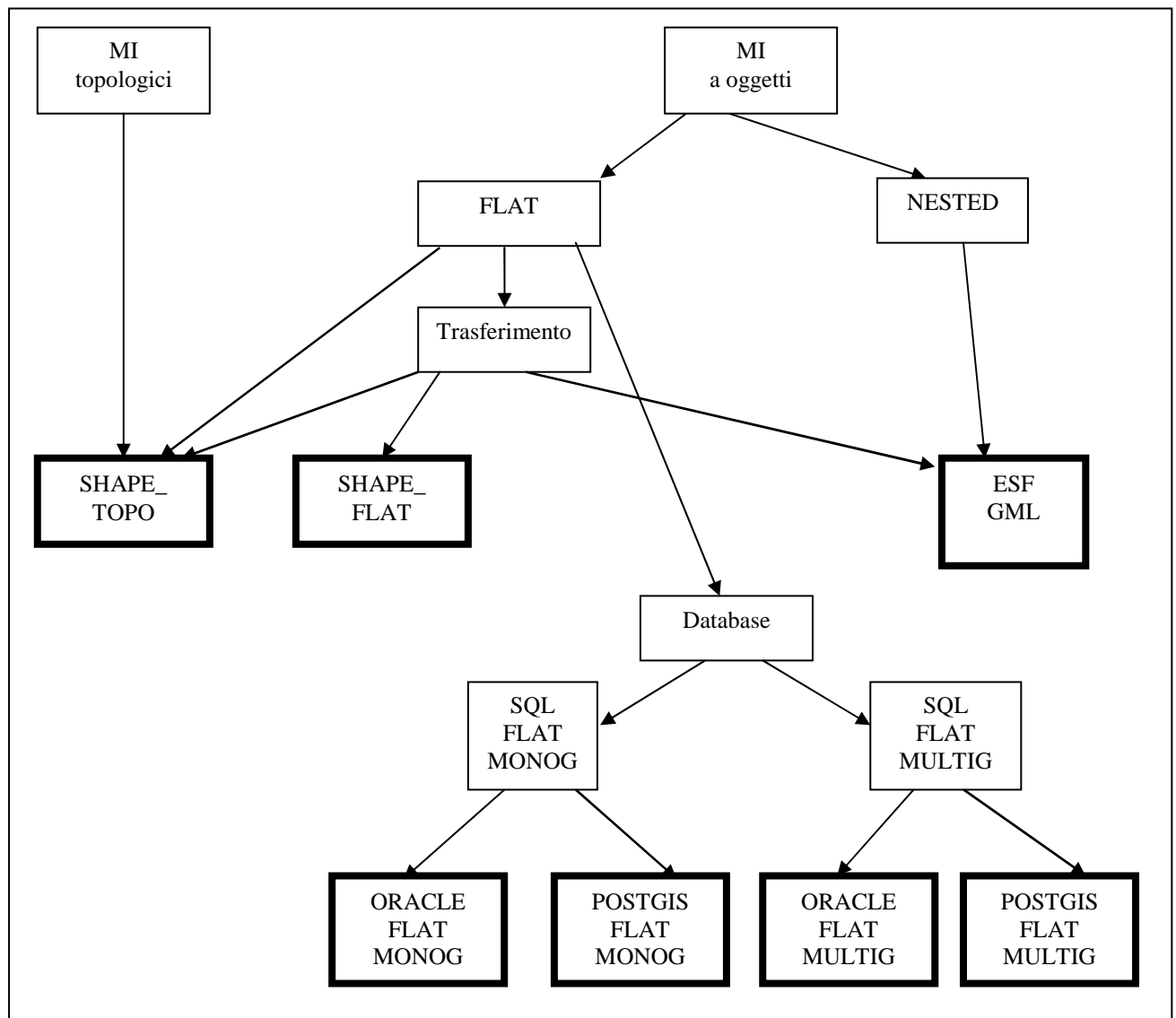
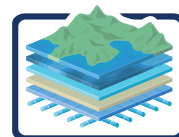


Figura 4.2



Appendice – RAPPORTO TRA GEOUML METHODOLOGY E GLI STANDARD ISO TC211

Impostazione generale

La GeoUML methodology copre un'area di problemi già regolamentati dagli standard ISO TC211.

Alla base della GeoUML methodology è posto il principio di essere compatibile con l'impostazione ISO TC211 in modo da poter usufruire di tutta la strumentazione conforme a tali standard che verrà presumibilmente prodotta; tuttavia la GeoUML methodology estende l'impostazione del ISO TC211 per venire incontro a tre tipi di esigenze:

1. aumentare, rispetto a tali standard, il **disaccoppiamento tra le Specifiche di Contenuto e le Strutture Implementative**, in modo da rendere il livello di astrazione delle Specifiche di Contenuto più conforme alla nozione classica di un Modello Concettuale, che è quella di una “descrizione dei contenuti informativi precisa ma indipendente da tutti gli aspetti di implementazione (che dipendono dalla tecnologia)”;
2. **gestire con un unico strumento** non solo la documentazione strutturata, ma anche la documentazione descrittiva, destinata ad essere letta dagli esseri umani, garantendo in tal modo l'allineamento tra questi due tipi di documentazione.

Si analizzano ora le motivazioni delle esigenze di arricchimento dell'impostazione ISO enunciate sopra e della soluzione proposta nella GeoUML methodology.

Disaccoppiamento tra le Specifiche di Contenuto e le Strutture Implementative

Nell'impostazione ISO esistono due componenti per la definizione dei contenuti di una specifica: l'Application Schema (AS) e il Feature Catalogue (FC).

Nell'impostazione ISO TC211 il livello di astrazione del AS (e, in parte, anche del FC) è più dipendente dalla implementazione di quanto non sia opportuno per una Specifica di Contenuto concettuale; in particolare gli aspetti che nella GeoUML methodology si è ritenuto di dover correggere sono i seguenti:

- a. i tipi geometrici ISO definiscono tutti gli aspetti di implementazione (interpolazione ecc...) che sono irrilevanti nella analisi e comprensione dei contenuti (ad esempio, a livello concettuale non è necessario distinguere se una “curva” è implementata come linestring, curva composta, ecc...), sono sostituiti nella GeoUML methodology da tipi più generici che vengono poi trasformati in tipi ISO tramite il Modello di Implementazione
- b. a causa delle regole molto strette per la produzione del GML definite in ISO 19136 l'AS deve contenere scelte di dettaglio che determinano il tipo di codifica GML che viene prodotto, ma che sono irrilevanti a livello concettuale e incomprensibili ai non esperti di GML
- c. la specifica dei vincoli in ISO TC211 è carente – il problema viene liquidato facendo riferimento al linguaggio OCL, che però non è adatto allo scopo per diversi motivi:
 - c1. OCL non è integrato con gli aspetti relativi ai dati geometrici, le relazioni topologiche, ecc..
 - c2. OCL non è sintatticamente adatto a una specifica concettuale, essendo eccessivamente difficile per un utente non specialista
 - c3. OCL non è implementabile
- d. mancano in ISO alcuni costrutti ad alto livello, come gli Attributi a Trattati e gli Attributi a Sottoaree, che devono essere rappresentati tramite la loro struttura implementativa in termini di Geometrie Composte (in questo caso risentendo oltretutto della carenza nell'espressione dei vincoli citata al punto precedente) oppure in forma di ascissa curvilinea, per la quale non è ancora definito l'apposito standard; nella GeoUML methodology la distinzione tra i diversi tipi di implementazione degli Attributi a Trattati è confinata nel Modello di Implementazione
- e. mancano in ISO TC211 costrutti ad alto livello per gestire la descrizione delle geometrie 2D e 3D, che caratterizzano i contenuti che vengono realizzati allo stato attuale dell'arte, caratterizzato da una impossibilità di realizzare un modello tridimensionale completo.

Avere isolato nella nozione di Modello di Implementazione gli aspetti che non appartengono al livello concettuale permette all'impostazione della GeoUML methodology di ottenere in ultima analisi due benefici:

1. la definizione delle Specifiche di Contenuto, che costituisce un processo complicato di analisi applicativa e confronto tra punti di vista disciplinari diversi, può concentrarsi solamente sugli aspetti veramente di contenuto, senza dover tener conto di aspetti implementativi, e permettere il confronto tra agli esperti applicativi senza la mediazione degli specialisti di tecnologia
2. le Specifiche di Contenuto non sono impattate da ogni evoluzione della tecnologia geomatica e possono resistere più a lungo nel tempo; è infatti possibile produrre nuovi Modelli Implementativi per supportare tale evoluzione, salvaguardando l'investimento in analisi applicative.



Gestione delle Specifiche con uno strumento unico

La strumentazione disponibile per gestire l'AS e il FC definiti da ISO non è unica; in particolare nel modello di gestione delle specifiche del ISO TC211 il componente fondamentale è il generico strumento CASE UML che gestisce l'AS. In effetti, questa scelta è comprensibile alla luce della storia dello sviluppo delle specifiche; ad esempio, tutto lo Spatial Model non avrebbe potuto essere facilmente prodotto senza questo approccio.

Tuttavia, nelle specifiche di contenuto l'esistenza dei diversi Package Geometrici è irrilevante; inoltre, nei database territoriali la scarsa numerosità delle associazioni presenti rispetto al numero di classi e di attributi rende l'uso di diagrammi UML assai poco utile. In sostanza, rispetto alla funzionalità di uno strumento CASE UML, le funzionalità veramente utili alla rappresentazione di una specifica di contenuto sono relativamente poche, mentre mancano le funzionalità necessarie alla gestione dei numerosi e importanti aspetti descrittivi.

Per rispondere a questo difetto, nell'impostazione ISO TC211 l'AS gestito dal CASE UML è affiancato da un Feature Catalogue che deve essere gestito con un diverso strumento. Il rapporto tra queste due rappresentazioni dei contenuti non è chiaro, e in effetti diversi enti in nazioni diverse hanno proceduto in maniera diversa (produzione prima del AS e poi del FC o viceversa).

Il GeoUML catalogue si propone di superare queste limitazioni tramite l'inserimento a monte rispetto ad AS e FC di una documentazione integrata rigorosamente di livello concettuale; da questa documentazione è possibile produrre l'Application Schema e il Feature Catalogue previsti da ISO TC211. Tale produzione avviene tramite la definizione di un Modello di Implementazione che aggiunge alla Specifica di Contenuto la definizione delle scelte implementative.

Nell'impostazione della GeoUML methodology il GeoUML catalogue è più simile a un Feature Catalogue, ma contiene molti elementi di un AS, con le precisazioni relative al livello di astrazione svolte nella sezione precedente, superando in tal modo i limiti degli strumenti CASE.