

SEMANTICA: UN SISTEMA PER L'INDICIZZAZIONE E IL RETRIEVAL SEMANTICO DI LEARNING OBJECT

Francesca Cantone, Mario Caropreso, Angelo Chianese,
Francesco Ficetola, Vincenzo Moscato

Università degli Studi Napoli "Federico II"
DIS – via Claudio 21, 80125 Napoli
[francesca.cantone, angchian, vmoscato]@unina.it
[ma.caropreso, f.ficetola]@studenti.unina.it

Abstract: *Tra i problemi più sentiti nella comunità dell'e-learning vi è quello della "personalizzazione" dei percorsi di apprendimento mediante costruzione di materiale didattico presente in ambienti distribuiti ed eterogenei. La definizione e la realizzazione di strumenti per la ricerca del materiale didattico utile all'assemblaggio resta pertanto una delle sfide ancora aperte nel settore del knowledge management. Nel presente lavoro viene illustrato un nuovo sistema, ispirato alle tecnologie del Web-Semantico, capace di supportare metodologie per l'indicizzazione e la ricerca di Learning Object basate sulla semantica ad essi associata. E tale semantica viene "contestualizzata" nei domini didattici di afferenza dei LO, al fine di potere costruire tool semi-automatici di generazione di contenuti didattici a partire da materiale digitale archiviato.*

Introduzione

Oggi, nel settore dell'e-learning, si avverte l'esigenza di disporre di strumenti in grado di recuperare e condividere informazioni presenti in ambienti distribuiti ed eterogenei. E una sfida ancora aperta è quella relativa all'introduzione di metodologie e strumenti per la ricerca di materiale didattico con caratteristiche utili alla costruzione dinamica di percorsi di apprendimento personalizzati.

È ormai infatti condivisa l'opinione che per rendere i materiali riutilizzabili in contesti e situazioni differenti, bisogna strutturarli in modo che ne venga garantita l'*interoperabilità tecnologica*. Con il termine di *learning object* si identificano materiali digitali riusabili in piattaforme differenti e che possono adottare diversi formati. La riusabilità implica che esista anche un *retrieval che sia reso efficace* legando contenuto e proprietà educative e pedagogiche di ogni learning object. In altri termini, affinché i learning object possano essere riutilizzati, devono essere opportunamente catalogati ed indicizzati, soprattutto quanto più cresce il volume e la tipologia del materiale disponibile.

Gli standard *AICC*, *ADL*, *IEEE LTSC*, *IMS* e *DUBLIN CORE* risolvono il problema dell'interoperabilità tecnologica tra ambienti eterogenei, ma soffrono dei problemi tipici degli approcci basati su *metadati* espressi in *XML*, ovvero la mancanza di una *semantica formale*. Per tale motivo combinare materiale di differenti autori può risultare complesso: il retrieval non viene efficacemente supportato, ed inoltre, il tipo di interoperabilità che i metadati supportano è quella all'interno di uno specifico dominio, mentre introducono il problema dell'incompatibilità fra descrizioni appartenenti a schemi di metadati diversi.

Un approccio che secondo diversi autori può soddisfare i requisiti sopra esposti per l'e-learning è quello del *Web Semantico* [Lee 2001, Aroyo 2004], il quale introduce il concetto di *metadati semantici*. "Annotare semanticamente le risorse didattiche" significa associare ad esse apposite *descrizioni semantiche*, che consentono di collegarle logicamente ad un certo numero di classi e proprietà definite in apposite *ontologie*.

Il funzionamento del Web Semantico è basato sulla possibilità di definire in modo formale la semantica di un insieme di informazioni e, conseguentemente, di automatizzarne l'elaborazione. I metadati semantici sono di tipo "machine understandable" e, possono essere

utilizzati dalle applicazioni per ottenere una piena interoperabilità semantica ed un retrieval efficace.

Il consorzio W3C ha prodotto una serie di standard per la rappresentazione dei metadati semantici che supportano elaborazioni più o meno sofisticate (e computazionalmente complesse): tali standard (escludendo il ruolo puramente sintattico esercitato da XML) sono l'*RDF* e l'*OWL*.

Il linguaggio RDF consente di formulare delle *asserzioni* su un insieme di risorse e le proprietà che le descrivono. L'utilizzo di RDF ha diversi vantaggi. In primo luogo, ogni modello RDF è basato sull'assunzione di *mondo aperto* dove i metadati sono selezionati da ontologie eterogenee. I metadati standard, invece, lavorano sull'assunzione di *mondo chiuso* confinando i metadati utilizzabili soltanto a quelli della piattaforma utilizzata per l'apprendimento (Learning Management Systems). In secondo luogo, RDF permette la creazione di statement complessi: si pensi ad esempio al meccanismo di reificazione che permette di enunciare fatti relativi ad un altro fatto. I metadati standard, invece, non permettono l'espressione di metadati complessi se non tramite meccanismi di classificazione tassonomica.

OWL è stato invece proposto per offrire un linguaggio ontologico per il Web e viene utilizzato per rappresentare chiaramente il significato dei termini nei lessici e le relazioni tra questi termini. Una tale rappresentazione dei termini e delle loro interrelazioni si chiama ontologia. OWL è in grado di esprimere il significato e la semantica molto meglio di quanto non lo possano fare XML e RDF.

Il lavoro presenta un sistema che, partendo dalle tecnologie del Web-Semantico, offre una serie di metodologie per l'indicizzazione e la ricerca di learning object basate sulla semantica ad essi associata e "contestualizzata" nel dominio didattico di afferenza, al fine di potere definire tool semi-automatici di generazione di contenuti didattici a partire da materiale digitale archiviato.

Il lavoro è organizzato come segue: nel paragrafo 2 vengono illustrati quelli che sono i principali approcci in letteratura per la gestione "semantica" di learning object; nel paragrafo 3 viene descritto il sistema realizzato; nel paragrafo 4 sono riportate le conclusioni in riferimento alla sperimentazione condotta.

Stato dell'arte

Molte sono state ad oggi le proposte di sistemi per l'annotazione e il recupero semantico di risorse didattiche in ambienti di e-learning.

In [Stojanovic 2001] gli autori presentano quelle che devono essere le caratteristiche principali di un ambiente di e-learning basato sul Web Semantico. In particolare, essi suggeriscono di migliorare l'utilizzo dei metadati tramite metadati basati sulle ontologie. Le ontologie hanno quindi il compito di far corrispondere ad ogni metadato un preciso significato, in maniera tale da garantire l'interoperabilità. Inoltre, il problema di fornire un significato comune si presenta a diversi livelli fra loro ortogonali di un learning object. Infatti, dal punto di vista di uno studente un learning object si può considerare come un oggetto strutturato su tre dimensioni: il *contenuto*, la dimensione di ciò di cui il learning object parla; il *contesto*, la dimensione del modo in cui il contenuto viene fornito; la *struttura*, la dimensione che compete all'organizzazione dei learning object in corsi completi, in quanto raramente essi vengono utilizzati da soli.

Il sistema *Courseware Watchdog* [Tane 2004, Schmitz 2005] propone un approccio basato su ontologie per il retrieval di learning object in grado di: i) capire l'ontologia e navigare all'interno del contenuto; ii) essere in grado di recuperare il materiale rilevante; iii) dare la possibilità di interrogare repository di risorse semantiche annotate; iv) effettuare il *clustering* dei documenti secondo l'ontologia; v) essere in grado di aggiornare l'ontologia e la knowledge base quando giungono nuovi dati. L'architettura è centrata intorno ad un sistema di gestione delle ontologie realizzato utilizzando il framework *KAON*. Le ontologie, definite *OIMODELS* (*Ontology-instance-models*) consistono di concetti, relazioni fra concetti, istanze, proprietà delle istanze, tutte riferite come entità. L'altro elemento centrale dell'architettura è il *text corpus*, che contiene un insieme di documenti e per ogni documento contiene il testo, i metadati e altre informazioni. L'astrazione per il text corpus è offerta da *Text-to-Onto*, un'estensione del framework *KAON* che offre dei meccanismi di *text mining* per l'analisi dei documenti. I due elementi più interessanti di questo approccio sono il *focused crawling* utilizzato per recuperare le risorse didattiche "rilevanti" dal Web e il *subjective clustering* per il raggruppamento delle risorse didattiche restituite dal crawler. Infine, il sistema si configura come un consumer nei confronti del network *Edutella*

[Nejdl 2002], una rete peer-to-peer per lo scambio di risorse didattiche.

Il sistema *PeOnto* [Fok 2005] propone un insieme di 5 ontologie in grado di supportare i diversi servizi di una piattaforma di e-learning. Esse sono : la *curriculum ontology* che rappresenta l'insieme degli obiettivi formativi che un'organizzazione vuole che le persone raggiungano; la *language ontology* che descrive la struttura di un particolare dominio di studio; la *pedagogy ontology* che descrive le procedure di instructional design e le relazioni fra le risorse e le attività di apprendimento; la *people ontology* che descrive la struttura delle persone coinvolte nel processo educativo e le relazioni fra esse; la *PEA ontology* che descrive le responsabilità degli agenti software intelligenti utilizzati dalla piattaforma di e-learning.

Il sistema *e-aula* [Sancho 2005] genera dinamicamente dei corsi assemblando le risorse didattiche tenendo conto del background dell'allievo, dei suoi obiettivi e del suo stile di apprendimento. Per realizzare tali obiettivi, gli autori propongono di annotare i learning object (LO) con due tipi di informazioni: le *informazioni pedagogiche*, utili a catturare la funzione educativa svolta da un LO; le *informazioni sul contesto*, che descrivono con metadati il contenuto di una risorsa didattica. Un learning object viene quindi annotato in riferimento a due ontologie: l'*ontologia pedagogica*, che si riferisce al ruolo educativo, e l'*ontologia di dominio*, che si riferisce al dominio di appartenenza della risorsa. Il profilo dell'allievo viene modellato utilizzando *IMS Learning Design specification*. In base alle informazioni presenti, il sistema assembla i LO e genera un corso completo applicando una trasformazione *XSLT* ai LO selezionati.

Il sistema *ELENA* [Dolog 2004] propone un sistema in grado di riempire il gap esistente fra i correnti sistemi di adaptive e-learning e i repository dinamici di learning object. Gli autori propongono un'architettura basata su servizi in grado di offrire funzioni di personalizzazione tramite *web-service* che sfruttano le annotazioni semantiche dei learning object. Il sistema utilizza tre tipi di ontologie: la *ontologia dei learning object* per la descrizione delle risorse didattiche viene utilizzata un'ontologia modellata secondo lo standard *Dublin Core*; la *ontologia di dominio* che permette di tenere traccia dei concetti che fanno parte di un dominio e delle loro relazioni reciproche; la *ontologia per gli allievi* che permette di descrivere le informazioni relative ai discenti. Il componente centrale dell'architettura è il *Personal Learning Assistant Service*, il cui compito è quello di utilizzare i servizi offerti dal sistema per cercare risorse, corsi e percorsi di apprendimento completi e adatti all'utente. Il componente fornisce un'interfaccia di ricerca che interagisce con un'ontologia sia per costruire le query che per specificare query che presentano concetti non presenti nell'ontologia. Il *Query Rewriting Service* estende poi la query dell'utente aggiungendo ulteriori vincoli e variabili non presenti nella query originaria. Tale estensione è basata su euristiche. Ad esempio, il servizio può aggiungere dei vincoli basati sulle preferenze dell'utente e le abilità linguistiche. Oppure può estendere la query in base alle esperienze di apprendimento precedenti memorizzate all'interno del profilo se lo scopo è quello di migliorare le abilità dell'utente. Il *Recommendation Service* fornisce delle annotazioni delle risorse didattiche basate sulle informazioni del profilo dell'utente. Tali annotazioni si riferiscono al ruolo educativo della risorsa, alla sua compatibilità con il profilo dell'utente, etc. La derivazione delle annotazioni viene effettuata tramite l'uso di euristiche. Il *Link Generation Service* utilizza delle euristiche per creare collegamenti semantici fra le risorse didattiche tenendo in conto le informazioni contenute nel profilo dell'utente. L'*Ontology Service* mantiene una o più ontologie e può essere interrogato per restituire l'intera ontologia, una parte di essa, o può rispondere a query del tipo "trova tutti i sottoconcetti di un concetto C", ma anche query sulle relazioni, del tipo "chi è autore del concetto C". Il servizio è in grado di supportare anche l'evoluzione delle ontologie nel tempo. Il *Mapping Service* permette di definire delle corrispondenze fra ontologie in maniera tale che i vari servizi non siano obbligati ad utilizzare le stesse ontologie. Tale servizio può essere utilizzato per esempio per ottenere la corrispondenza di un concetto C da una certa ontologia verso un concetto C' di un'altra ontologia, oppure per far corrispondere un'istanza I formulata in termini di un'ontologia ad un'istanza I' formulata in termini di un'altra ontologia. Il *Repository Service* fornisce un accesso a qualunque repository di learning object connesso ad una rete. I repository possono essere semplici files, singoli database, federazioni di database o reti P2P.

L'approccio proposto in [Verbert 2006] mira a creare un'architettura per il riutilizzo dei learning object in maniera flessibile. Gli autori partono dalla constatazione che i correnti modelli di aggregazione per i learning object permettono il riuso di un intero LO, ma non permettono il

riutilizzo dei componenti a grana fine. Gli autori suggeriscono un generico content model chiamato *ALOCoM* che definisce i LO e i loro componenti.

Il modello fa una differenza fra *Content Fragment (CF)*, *ContentObject (CO)*, *Learning Object (LO)*. I CF sono unità di contenuto in una forma di base, come ad esempio testo, audio e video. Praticamente, i CF sono le risorse digitali grezze. Essi possono essere specializzati nei due raggruppamenti di risorse discrete (immagini e testo) o continue (audio, video, simulazioni e animazioni). I CO aggregano i CF e aggiungono funzionalità di navigazione. I CO possono aggregare anche altri CO. Il successivo livello di aggregazione prevede che i CO si associno in un LO che aggiunge anche gli obiettivi didattici.

Infine, il sistema *QBLS* [Dehors 2005] è un sistema che utilizza le tecnologie e gli standard del Web Semantico per permettere il riuso delle risorse didattiche. Il sistema si basa sull'utilizzo di due ontologie. La prima ontologia è un'ontologia di dominio. Anziché creare ontologie di dominio in OWL o RDF, gli autori suggeriscono di utilizzare lo standard *SKOS (Simple Knowledge Organization System)* proposto dal W3C, che è facile da usare e offre una discreta potenza espressiva. La seconda ontologia è un'ontologia pedagogica con la quale si indica la strategia pedagogica da seguire durante la generazione di un corso. Una strategia pedagogica definisce ad esempio l'ordine di successione dei learning object in base al loro ruolo, asserendo che un esempio deve sempre seguire una definizione. Il retrieval dei learning object avviene sulla base dell'ontologia di dominio: i learning object vengono annotati rispetto ai concetti contenuti nell'ontologia; in fase di retrieval, il sistema recupera tutte le risorse che contengono il concetto scelto dall'utente e altri concetti collegati che vengono scoperti tramite il meccanismo di inferenza.

Gli approcci precedentemente descritti delineano quelle che devono essere le linee guida sia funzionali che architetturali nell'implementazione di un sistema di annotazione e retrieval semantico di learning object.

Il sistema SEMANTICA: architettura e funzionalità

La nostra proposta si ispira o per caratteristiche architetturali o per funzionalità o per modello dei dati ai vari sistemi delineati nello stato dell'arte, giungendo ad una soluzione innovativa con le seguenti peculiarità:

- differenti modelli per la rappresentazione dei dati (*content-based* e *document-based*),
- supporto ontologico (utilizzo di ontologie multiple per la descrizione dei learning object)
- architettura multi-livello con interfacce di tipo web-services,
- presenza di tool per l'annotazione automatica dei learning object, per il retrieval semantico di risorse didattiche e per la generazione automatica di corsi.

Il sistema per l'indicizzazione e retrieval semantico di learning object si basa sull'architettura multi-livello descritta in figura 1.

Le risorse che il sistema tratta sono **Learning Object (LO)**, memorizzati all'interno del componente denominato **Learning Object Repository (LOR)** la cui finalità è quella di garantire l'interoperabilità tra diversi formati. Il sistema infatti utilizza sempre i LO contenuti all'interno del repository, ignorando il loro formato originario e come essi sono stati caricati all'interno del data base. Le applicazioni possono quindi prelevare i learning object da qualunque sorgente e ricondurli al formato previsto da Semantica tramite appositi moduli di importazione.

Il sistema supporta contemporaneamente due modalità di rappresentazione (vedi figura 2) di un lo: come insieme di **document** (*modalità non strutturata*), e come insieme di **content unit** (*modalità strutturata*).

Un document è un'unità testuale contenuta all'interno di una risorsa didattica e rappresenta la base su cui è possibile effettuare elaborazioni testuali. L'insieme dei documenti di un lo ne rappresentano il suo contenuto testuale. Per ottenere migliori prestazioni dall'algoritmo di retrieval è necessario che ogni documento corrisponda ad una parte di un lo ben definita, ovvero relativa agli stessi argomenti (ad esempio, in riferimento al modello di contenuto previsto da IDEA [Chianese 2006], ogni content unit può essere trasformata in un documento). Sempre in relazione al testo contenuto in una risorsa didattica, il modello dà la possibilità di associare al lo una **Semantic Table** (tavola semantica) e una serie di **Context KeySense** (sensi chiave del contesto).

Una tavola semantica è un insieme di coppie (T, S) , dove T è un termine appartenente al LO e S è un senso appartenente ad un dato dizionario. Una tavola semantica permette di conoscere preventivamente il significato con cui un determinato termine viene utilizzato all'interno del LO.

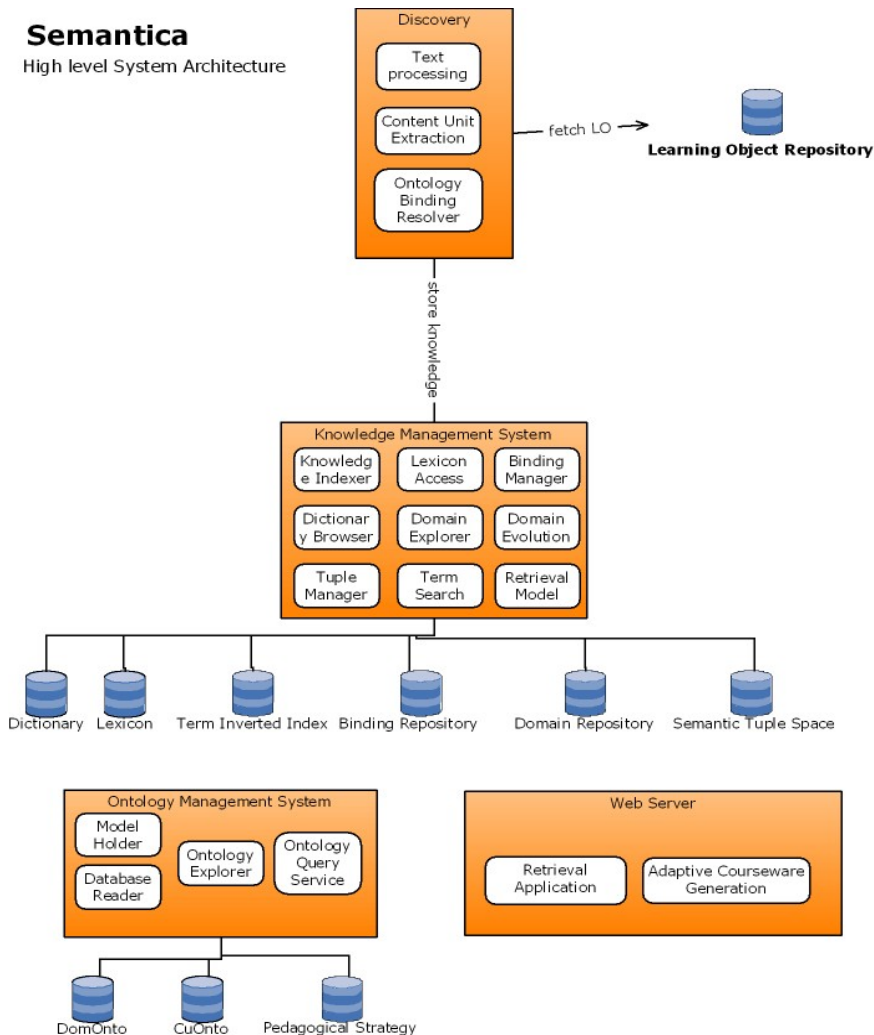


Figura 1 – Architettura di Semantica

Ciò permette di risolvere situazioni di ambiguità che possono nascere per il fatto che alcuni termini possono avere più di un significato. Nel caso in cui un termine compaia con la stessa forma ma con sensi diversi la tavola semantica permette di indicare esplicitamente quante volte il termine si presenta con un senso e quante volte con un altro.

L'utente può associare ad un lo una serie di sensi chiave, allo stesso modo di come al giorno d'oggi si assegnano ad una pagina web delle parole-chiave. I sensi chiave vengono utilizzati nel processo di *disambiguazione* di un testo. L'idea è quella di fare in modo che i sensi chiave diano un'idea del contesto dei termini utilizzati nel lo, in maniera tale da consentire la giusta interpretazione dei termini del documento.

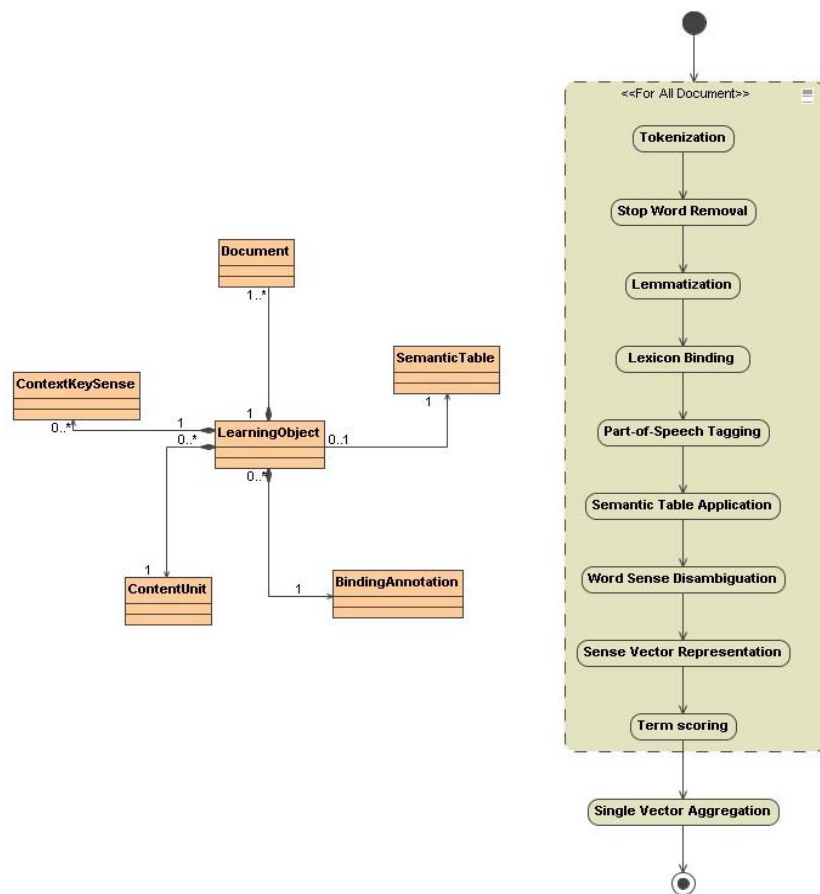


Figura 2 – Modello dei Learning Object e TextProcessing

La seconda modalità rappresenta invece un lo come insieme di unità di contenuto più piccole, dette content unit. Il nome di modalità strutturata evidenzia proprio il fatto che un lo non viene più visto come testo ma come insieme di costituenti più piccoli, che conferiscono la struttura. Un'unità di contenuto è una porzione del testo di un lo in grado di esprimere un determinato concetto di apprendimento. Ad esempio, in un testo di matematica le unità di contenuto possono essere i teoremi oppure le definizioni. Le unità di contenuto permettono di definire in maniera strutturata il messaggio educativo veicolato dal lo. Esprimendo infatti il contenuto come somma di unità di contenuto è possibile ad esempio stabilire quanto due lo sono simili, oppure è possibile stabilire se un dato lo contiene un certo argomento con una granularità molto spinta. E' interessante notare come le due modalità non siano esclusive, ma possano essere utilizzate contemporaneamente. Esse infatti sono due viste sul lo: la prima è non strutturata, in quanto rappresenta il lo come collezione di documenti; la seconda è strutturata, in quanto rappresenta il lo come collezione di unità di contenuto.

Indipendentemente dalla vista utilizzata per rappresentare un lo, ad ogni risorsa è possibile associare delle **Binding Annotation**. Una binding annotation è una sorta di metadato semantico. E' un metadato perché costituisce un'informazione su una risorsa, è semantico perché tale informazione viene scritta rispetto ad un'ontologia formale. Una binding annotation permette di legare un lo ad uno o più concetti presenti all'interno delle ontologie di dominio gestite dal sistema.

Dal LOR i lo vengono prelevati dal modulo denominato **Discovery**, il cui compito è quello di analizzare le risorse didattiche al fine di estrarre da essi il complesso di informazioni che costituiscono la *conoscenza* racchiusa all'interno dei lo.

In particolare, il modulo **TextProcessing** (vedi figura 2) opera sulla vista non strutturata e ha il compito di analizzare, mediante i classici algoritmi di *NLP (Natural Language Processing)*, i documenti contenuti all'interno di un LO e sottoporli ad una serie di passi di elaborazione al fine di ottenere una rappresentazione dell'oggetto sotto forma di vettore di significati. Per ogni lo il modulo fornisce in output un vettore di triple detto vettore dei significati. Ogni tripla contiene un termine del documento, il significato di quel termine prelevato dal dizionario e la frequenza della coppia termine-significato all'interno del documento.

Il modulo **ContentUnitExtraction** opera sulla vista strutturata di un lo e ha il compito di esaminare le content unit che compongono la risorsa didattica. Per ogni content unit trovata il modulo crea un'istanza memorizzabile all'interno della knowledge base.

Il modulo **OntologyBindingResolver** analizza le binding associations di un lo per determinare i concetti delle ontologie di dominio a cui esso si riferisce. Per ogni binding association il modulo genera un'istruzione di collegamento il cui scopo è quello di permettere alla knowledge base di associare il lo alla corretta ontologia.

La conoscenza del sistema è successivamente memorizzata all'interno di una **Semantic Knowledge Base (SKB)** gestita da un **Knowledge Management System (KMS)**. Il KMS esegue un'operazione di indexing sulle informazioni ricevute in maniera tale da fornire delle astrazioni dei diversi *repository di conoscenza* del sistema, offrendo alle applicazioni dei metodi per l'aggiunta di nuova conoscenza, la navigazione all'interno delle fonti di conoscenza e il recupero di informazioni tramite query. Al suo interno la SKB prevede diverse componenti, ognuna delle quali memorizza un tipo di informazione.

E' prevista la presenza di un **dictionary** (dizionario) che contiene tutti i termini di una determinata lingua. Per ogni termine ne vengono forniti i possibili significati. Il dizionario può essere definito come un'ontologia superiore linguistica, in quanto esprime anche le relazioni semantiche fra i termini. Le più importanti relazioni semantiche utili per il sistema sono quelle di sinonimia, iponimia/iperonimia (relazioni *IS-A*), meronimia/omonimia (relazioni *IS-PART-OF*). Nell'implementazione attuale del sistema il dizionario utilizzato è *WordNet* per la lingua inglese. Il **lexicon** tiene invece traccia di tutti i termini conosciuti dal sistema. Ogni qualvolta il sistema incontra un nuovo termine cerca di ricondurlo ad uno memorizzato nel lexicon e gli assegna un identificativo univoco. Il lexicon tiene traccia sia dei termini che sono presenti nel dizionario sia di quelli non presenti (come ad esempio nomi di località, di persone). Il **term inverted index** è la struttura dati scelta per la memorizzazione dei termini contenuti nei documenti. Tale struttura dati prevede che per ogni termine conosciuto dal sistema (e quindi presente nel lexicon) sia presente una lista, detta *posting list*, che contiene gli identificativi dei lo in cui quel termine viene utilizzato e la relativa frequenza. Si tratta di una struttura dati molto usata nel campo dell'information retrieval e su cui esiste consistente letteratura. Il **semantic tuple space (STS)** è il componente che permette la memorizzazione delle unità di contenuto di cui si compone un lo. Il STS rappresenta l'astrazione di uno spazio di memoria virtuale condiviso in cui le applicazioni possono scambiarsi dati tramite primitive di scrittura, lettura e rimozione. I dati che vengono scambiati sono detti tuple. Una tupla corrisponde ad un'unità di contenuto. Il modello è detto semantico perché le tuple che possono essere scambiate sono descritte da un'apposita ontologia del sistema [Ulrich 2005], detta **CuOnto**. Questo comporta due implicazioni. La prima è che ogni tupla ha un "significato", un "ruolo", all'interno dello spazio; la seconda è che lo spazio di memoria virtualizzato dal STS è uno spazio strutturato, in quanto le tuple sono in relazione fra di loro. Il **domain repository** contiene le descrizioni dei domini del sistema. Un dominio rappresenta una conoscenza di un certo ambito rappresentata sotto forma di insieme di concetti legati fra di loro. Il **binding repository** è un componente il cui compito è quello di tenere traccia delle associazioni fra i lo e i concetti contenuti all'interno del domain repository.

Di seguito è invece riportata la descrizione delle componenti del KMS.

Il **knowledge indexer** è il componente che ha la responsabilità di ricevere le informazioni dal modulo Discovery e prepararle per la memorizzazione all'interno della knowledge base. Il **tuple manager** mette a disposizione le primitive per scrivere, leggere, rimuovere tuple dallo spazio delle tuple ed eseguire ricerche. Il **term search** permette di eseguire ricerche sull'inverted index che coinvolgono uno o più termini. Il **dictionary browser**, il **lexicon access** e il **domain explorer** sono dei *wrapper* per il dizionario, il lexicon e il domain repository. Il **domain evolution** permette l'aggiornamento dei domini del sistema. Il **retrieval model** è un componente che permette al sistema di implementare diverse modalità di retrieval dei dati contenuti all'interno della knowledge base.

La semantica dei dati contenuti all'interno della knowledge base viene gestita dall'**Ontology Management System (OMS)**, che contiene i modelli delle ontologie gestite dal sistema. Il sistema si basa sull'utilizzo, di due ontologie base che forniscono la semantica dei dati contenuti all'interno della Knowledge Base.

La **Domain Ontology** è usata per rappresentare insieme di concetti che vengono utilizzati in una certa disciplina didattica (i.e. mappa concettuale), la **Task Ontology** (CuOnto) descrive la semantica/ruolo (e.g. introduzione, definizione, etc...) delle content unit che compongono un lo. Per ogni unità di contenuto sono individuate quindi tre dimensioni: il testo, il contesto e la struttura.

Il sistema contiene poi una serie di ontologie specifiche dell'applicazione denominate **Pedagogical Strategies**. Una strategia pedagogica è un insieme di regole che stabiliscono come scegliere le content unit per comporre un corso. Ad esempio, una strategia può prevedere che una content unit che spiega un determinato argomento debba precederla una che illustra un esempio sullo stesso argomento. Le regole vanno scritte utilizzando il formalismo proprio di *Jena*.

Il **DB Reader** permette di leggere un'ontologia da un insieme di tabelle in un database relazionale. La memorizzazione di un'ontologia all'interno di un database comporta notevoli vantaggi. Infatti il sistema non ha la necessità di copiare il contenuto dell'ontologia (che può contenere una mole notevole di dati) nella memoria principale, ma può lasciarli nel database e creare una vista ontologica dei dati, creando una sorta di ontologia virtuale. L'ontologia è virtuale nel senso che non è materializzata, ma ogniqualvolta il sistema necessita di un certo concetto accede direttamente al database. Questo modulo è realizzato utilizzando le funzionalità del framework *KAON 2*.

L'**Ontology Explorer** consente la navigazione all'interno dei concetti e delle relazioni presenti all'interno di un'ontologia, permettendone la visualizzazione. L'**Ontology Query Service** è un componente in grado di eseguire delle interrogazioni sull'ontologia e fornire al richiedente i concetti che rispondono alla query. Le query sono formulate in *SPARQL*.

Le funzionalità che il sistema offre, ovvero quelle di ricerca semantica e di generazione dei corsi sono contenute all'interno di un Web Server e accessibili via Web. Le applicazioni sono realizzate utilizzando il paradigma *MVC* di *Apache Struts* e sono costituite da una parte business il cui compito è interagire con l'OMS e il KMS per eseguire le ricerche dei contenuti didattici. Per quanto riguarda la prima sono previste le seguenti modalità di ricerca:

- **Sense Boolean Retrieval Model (SBRM)**: simile al meccanismo tradizionale utilizzato dai motori di ricerca con la differenza che lavora sui sensi e non sulla grafia del termine;
- **Content Unit Retrieval Model (CURM)**: permette di cercare un lo sulla base delle unità di contenuto che possiede;
- **Ontology Driven Retrieval Model (ODRM)**: cerca i lo sulla base dei concetti di un'ontologia di dominio.

Il modello SBRM utilizza le informazioni contenute all'interno del Term Inverted Index della Semantic Knowledge Base. Tale indice viene utilizzato per costruire una sorta di spazio in cui ogni lo viene rappresentato come un vettore di significati. Si tratta di una variazione del modello detto *Vector Space Model* che rappresenta ogni documento come un vettore di termini. In fase di ricerca l'utente specifica una query fornendo parole chiave connesse da *connettivi booleani*. Il sistema permette all'utente per ogni termine della query di indicare il senso del termine. In tale modo la query viene trasformata in un *vettore di sensi*. Il vettore dei sensi viene inviato al sistema, che interroga la tabella dei sensi per cercare una corrispondenza con il vettore di input. I sensi per i quali esiste una corrispondenza sono definiti *sensi vincenti*. Dall'elenco dei sensi il sistema estrae un elenco dei lo che contengono i sensi vincenti.

Ad ogni lo viene attribuito infine un punteggio che è pari alla somma delle densità di significato dei sensi vincenti.

La metodologia di ricerca CURM è basata invece sulle tecniche del Web Semantico ed ha l'obiettivo di consentire una ricerca di lo sulla base delle unità di contenuto. In fase di ricerca l'utente sottopone al sistema una query che viene costruita utilizzando l'approccio *Query by example*. L'utente costruisce la query istanziando degli oggetti di esempio dall'ontologia CuOnto e, per ognuno di essi, indica delle proprietà che gli oggetti da trovare dovranno rispettare. Un'altra modalità di ricerca è quella basata sul contesto e consiste nello scegliere alcune content unit contenute all'interno della knowledge base che presentino particolari

proprietà contestuali o strutturali. Ad esempio, l'utente può selezionare la Content Unit denominata "Teorema di Pitagora" e chiedere di ottenere tutte le Content Unit la cui proprietà *requires* ha come valore proprio il Teorema di Pitagora. La query, in qualunque modo sia stata composta, viene trasformata in una query SPARQL ed eseguita sulla knowledge base (KB). La KB restituisce un elenco di unità di contenuto che soddisfano la query inviata prelevandole dal Semantic Tuple Space. Un ulteriore passo elaborativo è necessario nel caso in cui l'utente abbia fornito dei valori per le proprietà testuali. Ad esempio, per fornire tutti i *Fundamental* che contengano i sensi delle parole "Napoleone", "Francia", il sistema prima accede alla tabella costruita nella fase di indexing per le proprietà testuali e cerca le ContentUnit che soddisfino la query sul testo, e dopo carica i lo che contengono le unità di contenuto vincenti ed associa ad ognuno di essi un *rank*. I lo risultati vengono ordinati secondo il rank e restituiti all'utente secondo tale ordine.

La modalità ORDM si basa sull'utilizzo massiccio di ontologie. In particolare l'approccio utilizza delle ontologie di dominio per descrivere i diversi domini di articolazione della didattica. In fase di ricerca l'utente utilizza un'interfaccia grafica per navigare all'interno delle ontologie di dominio. L'utente può selezionare un'ontologia di dominio e poi un dato concetto (istanza della classe *Concept*). Il processo può essere ripetuto più volte in maniera tale da permettere all'utente di definire una query complessa che coinvolga più concetti. E si possono selezionare due opzioni: la prima limita la ricerca soltanto a lo che siano stati esplicitamente collegati ai concetti dell'ontologia di dominio tramite binding annotations, la seconda invece abilita la ricerca anche a lo non esplicitamente annotati. Nell'ultimo caso il sistema preleva per ogni concetto della query la lista dei sensi collegati ed accede ai vettori dei sensi dei documenti che sono stati creati con l'approccio SBRM.

Infine, le informazioni contenute all'interno della KB possono essere utilizzate per assemblare dinamicamente i lo e creare dei corsi personalizzati. L'utente può utilizzare due meccanismi per la generazione automatica dei corsi che differiscono per la modalità in cui l'utente sceglie i lo componenti:

- il **ContentUnit Adaptive Courseware Generation (CUACG)** che utilizza il modello CURM per il retrieval dei lo e prevede la possibilità di generare automaticamente dei corsi didattici permettendo di definire in maniera precisa gli obiettivi di apprendimento;
- l'**Ontology Driven Adaptive Course Generation (ODACG)** che utilizza il modello ODRM per il retrieval di lo da utilizzare per la generazione automatica di corsi e prevede che l'utente navighi attraverso la knowledge base e selezioni un dominio (*ConceptScheme*) in cui generare il corso; il sistema mostra la lista dei concetti presenti in quel determinato dominio e l'utente seleziona se desidera generare un corso che copra tutto i concetti di quel dominio oppure sceglie quali concetti coprire.

Risultati della sperimentazione

Il sistema in oggetto è stato testato attraverso la generazione automatica di una decina di LO relativi al dominio dell'**Archeologia Informatica** sulla base di 1000 content unit indicizzate ed ha mostrato risultati molto interessanti. In particolare, più del 50% degli studenti che hanno seguito i corsi generati in automatico col modello CUACG hanno ritenuto che: i contenuti non erano troppo ripetitivi (*ripetitività dei contenuti*); i concetti fondamentali erano stati trattati tutti (*priorità dei contenuti rispetto ai concetti base*); i contenuti erano adeguati alle esigenze di apprendimento (*priorità rispetto agli obiettivi formativi*); i contenuti erano poco frammentari (*continuità dei contenuti*). I 10 corsi sono stati poi valutati da altrettanti esperti del dominio per misurarne la qualità dal punto di vista del docente sempre rispetto ai parametri di continuità, ripetitività e priorità rispetto ai concetti base espressi mediante keywords.

Dalla figura 3, che mostra i risultati della sperimentazione, si può notare come il sistema di generazione automatica dei corsi presenti prestazioni abbastanza soddisfacenti (la scala adottata va da 0 a 10).

Corsi generati	Esperti coinvolti	Non ripetitività	Continuità	Priorità	KEYWORDS
3D Graphics	Esperto 1	10	8	10	PIXEL
Database	Esperto 2	7	6	8	TABLE
Sharing Systems	Esperto 3	9	7	8	SERVER
Artificial Intelligence	Esperto 4	8	8	7	NEURON
Information Systems	Esperto 5	8	6	7	DATA
Virtual Reality	Esperto 6	7	7	7	REALITY
Digital Archives	Esperto 7	8	8	8	ARCHIVE
Database Management	Esperto 8	7	6	8	DBMS
Geographic Systems	Esperto 9	10	7	9	SATELLITE
Virtual Museum	Esperto 10	8	8	9	MUSEUM
		8,2	7,1	8,1	

Figura 3 – Risultato sperimentazione

Bibliografia

- (Lee, 2001) T. Berners-Lee et al., "The Semantic Web" - *Scientific American*, 2001.
- (Aroyo, 2004) L. Aroyo, D. Dicheva, "The New Challenges for E-learning: The Educational Semantic Web", *Educational Technology & Society*, 2004.
- (Stojanovic, 2001) L. Stojanovic, S. Staab, R. Studer, "eLearning based on the Semantic Web", *WebNet2001-World Conference on the WWW and Internet*, 2001.
- (Tane, 2004) J. Tane, C. Schmitz, G. Stumme, "Semantic Resource Management for the Web: An E-Learning Application", *World Wide Web conference*, 2004.
- (Schmitz, 2002) C. Schmitz et al., "Accessing Distributed Learning Repositories through Courseware watchdog", *World Conference on E-Learning*, 2002.
- (Nejdl, 2002) W. Nejdl et al., "EDUTELLA: a P2P networking infrastructure based on RDF", *International conference on World Wide Web*, 2002.
- (Fok, 2005) A. W. P. Fok, "Ontology-driven Content Search for Personalized Education", *13th annual ACM international conference on Multimedia*, 2005.
- (Sancho, 2005) P. Sancho et al., "Semantic Web Technologies Applied to e-learning Personalization in <e-aula>", *Journal of Universal Computer Science*, 2005.
- (Dolog, 2004) P. Dolog, M. Sintek, "Personalization in Distributed e-Learning Environments", *International World Wide Web*, 2004.
- (Verbert, 2006) K. Verbert, D. Gašević, J. Jovanović, E. Duval, "Ontology-based Learning Content Repurposing", *Educational Technology & Society*, 2006.
- (Dehors, 2005) S. Dehors, C. Faron-Zucker, "QBLS: A Semantic Web Based Learning System", *Workshop on Appl. of Semantic Web Techn. for E-Learning*, 2005.
- (Chianese, 2006) A. Chianese et al., "IDEA: A Developing Tool of Learning Resources for LMS Environments", *Conference on Advanced Learning Techn.*, 2006.
- (Ulrich, 2005) C. Ullrich, "Description of an Instructional Ontology and its Application in Web Services for Education", *International Workshop on Applications of Semantic Web Technologies for E-Learning*, 2005.