

Indice

Sommario	1
Introduzione	3
1 Capitolo Primo: La didattica della geometria con le nuove tecnologie	6
1.1 Strumenti e obiettivi per la didattica sul Web	6
1.2 La geometria con i modelli matematici	8
1.3 Dalla didattica tradizionale alla didattica sul web	16
1.4 I Learning Object	28
2 Capitolo Secondo: L'E-Learning	32
2.1 Un framework di rappresentazione per oggetti 3D nel Web Semantico	32
2.2 L'E-learning e la didattica della geometria	56
2.3 Contenuti e metodologie didattiche per un corso di Geometria sul Web: le lezioni sulle quadriche di Luigi Campedelli	59
2.4 Modelli di E-Learning con filtri collaborativi	70
2.5 Principali tecniche di Web Recommending	77
2.5.1 Prodotto scalare di attributi	77
2.5.2 La Graph HIT distance	78

3	Capitolo Terzo: Algoritmi di approssimazione per il Grid-Service Discovery	80
3.1	3D-Resource brokering con algoritmi basati su Nurbs	80
3.1.1	Funzioni di base B-Spline	82
3.1.2	Curve B-Spline e curve NURBS	83
3.1.3	Superfici B-Spline e superfici NURBS	87
3.1.4	B-Spline Approximation nel senso dei minimi quadrati	89
3.1.5	NURBS Approximation nel senso dei minimi quadrati	103
3.2	Dettagli implementativi dell'algoritmo di approssimazione	105
3.3	Risultati dell'algoritmo	109
3.3.1	Approssimazione di superfici del secondo ordine	109
3.3.2	RDF di una superficie quadrica	111
3.4	Una shell semantica per il 3D-resource broker	118
	Riferimenti bibliografici	125

Sommario

La tesi è il risultato di un'attività di ricerca in un ambito che coinvolge la computer grafica e l'E-learning.

Nello specifico, la tesi traccia una "roadmap" per la trasposizione on-line dei corsi di geometria.

L'E-learning richiede la possibilità di riutilizzare le risorse didattiche e costruire *learning object* più complessi. Pertanto, nella tesi sono messi a punto dei modelli -sia teorici che implementativi- per il riutilizzo delle collezioni di modelli rappresentanti superfici matematiche costruiti nel secolo scorso [7].

Nel Capitolo 1 le lezioni di geometria sul Web sono contestualizzate con lo stato dell'arte delle tecnologie didattiche e si descrive una lezione on-line sulla geometria delle superfici del secondo ordine.

Alla fine del primo capitolo si definiscono i concetti dell'E-learning, dagli aspetti "in the small" come i learning object, a quelli "in the large" come i *content providers*, i *learning service providers* e le *comunità di apprendimento*.

Nel secondo capitolo, la descrizione degli strumenti tecnici è completata dalla costruzione di un learning object per la didattica della

geometria.

Nel Capitolo 3 si descrive un algoritmo di approssimazione di un insieme di punti che campionano una superficie, mediante superfici NURBS. Successivamente si inserisce questo algoritmo in una "shell" del Web Semantico, che consente ai Learning Object grafici presenti sulla rete di passare dalla "machine representability" alla "machine processability".

Introduzione

L'E-learning è la didattica e l'apprendimento in uno spazio virtuale, pertanto si svolge in mancanza di riferimenti assoluti, sia di carattere fisico -come lo spazio e il tempo- sia di carattere topologico -come le tre dimensioni del testo- e il percorso didattico tradizionale è diventato un grafo di interpretazioni, l'Ontologia, che si ricombinano in maniera ordinata nell'interazione tra ipertesto, utente e mediatore didattico.

Il docente -o *instructor*- prepara l'Ontologia sotto forma di *grafo* e il discente sceglie l'interpretazione adatta con la mediazione di un *tutor*, detto anche *mediatore didattico* o *facilitator*.

Nel caso della geometria, il compito del mediatore didattico è quello di selezionare, nella struttura del corso preparata dal docente, un percorso didattico adattivo e associare il materiale didattico

Si determina un nuovo modello di E-learning, in cui sono identificati tre tipologie di attori

- I fornitori di contenuto (Distributed Content Provider),
- I fornitori di servizi di E-learning (Learning Service Provider),
- Le comunità virtuali (Virtual Community).

Per restare fedeli alla trasposizione didattica originale di Luigi Campanelli si è messo a punto un ambiente di apprendimento 3D in cui lo studente può percepire come autoevidenti delle proprietà che possono essere dimostrate nel sistema di assiomi della geometria.

L'E-learning *in the small* interseca la Computer Science in un'area

in cui si sono definiti modelli e definizioni computabili. Partendo dalla definizione di Dewey, accettata in letteratura, si è descritta una teoria computabile basata su una nuova definizione [86], in cui un Learning Object è “*la più piccola unità di apprendimento indivisibile rispetto alla sua valenza didattica*”. In base a questa definizione si è elaborato un modello di Learning Object computabile, in cui un Learning Object è una coppia di due operatori che, al limite, converge ad uno stesso operatore di chiusura.

Questa definizione include il modello standard SCORM (CAM) basato su metadata e si avvale di una nuova definizione in cui un Learning Object è un punto fisso di un processo che è vincolato da due possibili interpretazioni: un assemblaggio illimitato (*bottom-up*), oppure una frammentazione illimitata (*top-down*).

Nel terzo capitolo si affronta il problema del *3D-Resource Discovery*, inserito nel contesto degli *shape descriptor*, definiti da Kazhdan alla Princeton University, e dal punto di vista della rappresentazione della conoscenza con il *Resource Description Framework*.

In particolare, si studia un algoritmo di approssimazione NURBS (*Non Uniform Rational B-Spline*) di un insieme di punti, chiamato *sampling set*, si descrive una *RDF* delle quadriche e delle superfici e si studia un modello di compilazione del pointset con *similarity based shape descriptor*. Con la definizione di uno shape descriptor basato su un insieme di punti, è possibile estendere il motore di ricerca Shape, sviluppato alla Princeton University, con un tool di generazione di superfici NURBS che si può usare sia per generare superfici richieste da un linguaggio di query che per confrontare la query stessa con degli

oggetti 3D già esistenti nelle repository.

La rappresentazione RDF consente di includere le applicazioni di *resource discovery and brokering* nel framework del *Web Semantico*; in questo modo, i broker possono svolgere la loro funzione in modo naturale, interfacciando le richieste del livello applicativo delle architetture distribuite (i *Learning Management System*) con le risorse disponibili nelle repository dei Learning Object 3D.

Il Learning Object che si è messo a punto, descritto nel primo capitolo, è dotato di un socket di interfaccia con il livello di brokering dell'algoritmo di approssimazione NURBS. Si vuole permettere al docente/tutor di cambiare la superficie che deve essere analizzata dallo studente, in relazione al problema in esame.

Nella terza parte si descrive, quindi, l'algoritmo basato sull'approssimazione nel senso dei minimi quadrati di una superficie, mediante una superficie NURBS, a partire da una griglia di punti nello spazio.

CAPITOLO 1: La didattica della geometria con le nuove tecnologie

1.1. Strumenti e obiettivi per la didattica sul Web

L'E-learning è probabilmente la didattica in uno spazio antropologico senza riferimenti assoluti di tempo e locazione. E la didattica -in una definizione di Maragliano- è tutto quanto attiene alla forma che assumono i saperi per diventare oggetto di insegnamento.

La scelta del metodo di insegnamento non è neutra rispetto alla considerazione del luogo dentro cui essa viene esercitata. Si distingue, allora, senza emettere un giudizio di valore, tra una didattica informale, per molti aspetti empirica ed esposta al senso comune, ed una didattica formalizzata.

Nel caso dell'E-learning, si capisce che esso non può darsi una didattica con una base di insegnamento troppo fragile, anche se la tendenza naturale - osserva Negroponte [71]- è di appropriarsi di un nuovo mezzo nello stesso modo in cui se ne usano i predecessori.

Il complemento naturale dell'E-learning sono le tecnologie multimediali, integrate in una galassia mediatica che -almeno in misura potenziale- può arricchire le basi dell'insegnamento, costringendo ogni

nuova didattica per l'E-learning a reggimentare risorse e procedure. In pratica, per dirla con Marshall Mac Luhan, [Marshall Mac Luhan, *La galassia Gutenberg*, Feltrinelli, Milano 1962] per usare con successo l'E-learning è necessario "connettere e connettersi".

L'E-learning è l'evoluzione del Computer Based Training, a sua volta originato dalle "macchine per insegnare" degli anni '60, che venivano modellate come scatole nere della teoria dei sistemi.

Per contro, la complessità della didattica sul Web non si deve misurare con indici quantitativi, quanto piuttosto va riferita a quella asimmetria che scaturisce dall'adeguamento di ogni sistema all'operare del singolo soggetto agente [86]. In pratica, nell'E-learning non può darsi obiettivo didattico senza un sistema adattivo -umano o automatico- che tiene conto delle caratteristiche del singolo studente.

Con queste linee guida, ci proponiamo di implementare le lezioni di Geometria del professor Luigi Campedelli con lo stato dell'arte delle tecnologie Web.

1.2. La geometria con i modelli matematici

La storicizzazione delle lezioni di Geometria di Luigi Campedelli è occasione per rilevare il processo di trasformazione delle tecnologie didattiche, dalle opere in cui stata codificata la geometria descrittiva, -*Gaspard Monge Geometrie Descriptive, 1794*- all'affermazione dei metodi analitici - *Gaston Darboux, Géométrie Analytique 1917*.

Dalle rappresentazioni descrittive con riga e compasso nella seconda metà dell'ottocento si è passati all'uso dei modelli rappresentanti superfici matematiche, realizzati -per lo più in gesso e legno- dapprima da artigiani e poi con metodi per la produzione in serie:

"L'accettazione di un'esperienza iniziale, di convalida e conforto all'intuizione, per avere il modo di fissare le basi, i punti di partenza (i postulati), al ragionamento deduttivo. Poi più nessun ricorso all'esperienza o all'intuizione, se non come metodo euristico, cioè per trarne suggerimento alla scoperta".

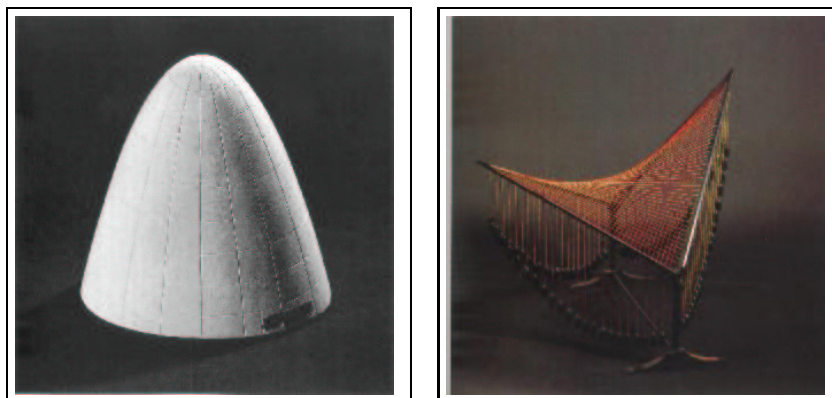


Figura 1.1: *Modelli del 1878 rappresentati un paraboloide ellittico e un paraboloide iperbolico.*

La ricerca storica sui metodi per l'insegnamento della Geometria è un'attività che non manca di punti da chiarire, di paradossi curiosi e di evidenze sorprendenti.

La storia della matematica dice che, dopo il 1920, i modelli matematici delle superfici vengono progressivamente abbandonati; a chiedersene il motivo, ci si imbatte nei contatti dei matematici del gruppo "Bourbaki" -in particolare Weyl, Brower e Beth- con lo psicologo Jean Piaget e il filosofo Michel Foucault, che impongono anche in matematica uno dei cardini della filosofia di Martin Heidegger *essere è linguaggio*, cui fanno da contrappunto i risultati di Jacques Herbrand, un logico francese che nel 1930 trova delle teorie del primo ordine in cui il modello è contenuto interamente nel loro linguaggio.

Questi risultati, che completano le ricerche avviate nel secolo precedente da Ludwig Wittgenstein, inducono i matematici della scuola russo-francese a considerare inutile ogni interpretazione che in qualche modo si estende oltre i confini delle strutture linguistiche e si arriva

alle posizioni radicali come quella del russo Sylow, che, nel suo Corso di Analisi, afferma che la matematica è lo studio delle strutture del pensiero matematico.

La consultazione dei testi conservati nei dipartimenti di matematica italiani e nelle varie sedi della Mathesis (Società italiana di scienze matematiche e fisiche, fondata nel 1895) [8] porta a concludere che i modelli di superfici matematiche si sono estinti nella cultura underground di un circolo dei nostalgici. Riprenderle per un corso universitario -in questo contesto-non ha molto senso.

I modelli matematici non valgono il gesso con cui sono costruiti? Nient'affatto!

L'evidenza ci fa vedere che le più importanti costruzioni dell'architettura moderna sono ispirate da modelli di quadriche, così come tutte le strutture più recenti contengono superfici quadriche realizzate con superfici NURBS. Una scuola di progettazione codificata da Peter Eisenman e seguita da architetti di fama mondiale, come Frank Gehry, Gary Lynn e Zahah Hadid, si è imposta come avanguardia nell'architettura contemporanea per l'uso intensivo nei loro progetti dei modelli matematici "nurbizzati". Se si pensa che la didattica dei modelli matematici era essenzialmente la loro costruzione, vuol dire che da qualche parte qualcuno ha continuato ad insegnarli. E senza sbagliare molto, a giudicare dai risultati dei propri allievi.

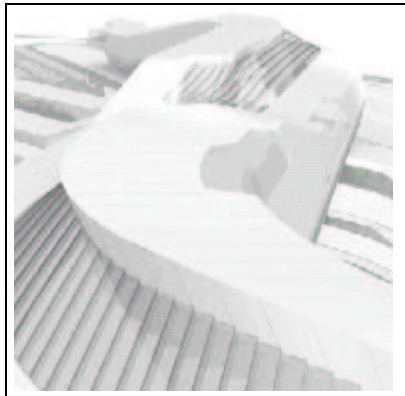


Figura 1.2: *NURBS per la Nuova Stazione Ferroviaria TAV di Napoli-Afragola.*

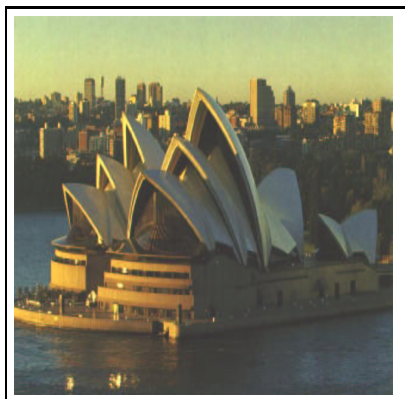


Figura 1.3: *Parte esterna della Sideny Opera House. Le falde dell'iperboloide.*



Figura 1.4: *London Swiss Re Building, a forma di ellissoide.*

La matematica non è più quella degli anni '30, sia nell'accumulazione dei suoi risultati, sia, soprattutto, nelle interpretazioni epistemiche. Ad esempio, è interessante confrontare *l'art de raisonner sur des figures mal faites* da parte di Henri Poincaré, che mostra l'incombenza della topologia

On a dit souvent que la géométrie est l'art de bien raisonner sur des figures mal faites. (...) Mais qu'est-ce qu'une figure mal faite? C'est celle que peut exécuter le dessinateur maladroit dont nous parlions tout à l'heure ; il altère les proportions plus ou moins grossièrement; ses lignes droites ont des zigzags inquiétants ; ses cercles présentent des bosses disgracieuses ; tout cela ne fait rien, cela ne troublera nullement le géomètre, cela ne l'empêchera pas de bien raisonner. Mais il ne faut pas que l'artiste inexpérimenté représente une courbe fermée par une courbe ouverte, trois lignes qui se coupent en un même point par trois lignes qui n'auraient aucun point commun, une surface trouée par une surface sans trou. Alors on ne pourrait plus se servir de sa figure et le raisonnement deviendrait impossible. (...) Cette observation très simple nous montre le véritable rôle de l'intuition géométrique ; c'est pour favoriser cette intuition que le géomètre a besoin de dessiner des figures, ou tout au moins de se les représenter mentalement. Or, s'il fait bon marché des propriétés métriques ou projectives de ces figures, s'il s'attache seulement à leurs propriétés purement qualitatives, c'est que c'est là seulement que l'intuition géométrique intervient véritablement

con la nozione di topologia recepita dagli esponenti della scuola di

Peter Eisemann, che probabilmente allo stesso Poincarè non sarebbe familiare. Tra le superfici chiuse rappresentabili, solo le NURBS possono essere deformate dinamicamente.

Architectural topology means the dynamic variation of form facilitated by computer-based technologies, computer-assisted design and animation software. The topologising of architectural form according to dynamic and complex configurations leads architectural design to a renewed and often spectacular plasticity, in the wake of the baroque and of organic expressionism.

Il recente sviluppo delle superfici NURBS in ingegneria e in architettura è una motivazione alla ripresa delle didattiche della Geometria con i modelli matematici in versione NURBS. E' altresì necessaria l'utilizzazione di strumenti di E-learning, sia per le lezioni remote che per quelle tradizionali. Riprendere un corso di geometria con i modelli matematici vuol dire capire cosa è successo tra gli anni '30- quando i modelli matematici sono scomparsi nella clandestinità- e gli anni '90, quando i modelli matematici sono ricomparsi -con le superfici NURBS-come espressione canonica nelle avanguardie dell'architettura.

Tra i lavori di Hilbert e di Peano sui fondamenti della geometria a fine Ottocento e il tentativo tarskiano di fondare la geometria con le tecniche meta-matematiche dei linguaggi elementari, sono trascorsi cinquant'anni. A inizio secolo la logica matematica si è incentrata sul problema dei fondamenti e sulle dimostrazioni di coerenza delle teorie: poiché lo studio della coerenza della geometria poteva es-

sere ricondotto a quello della coerenza dell'aritmetica, le riflessioni meta-matematiche sulla geometria e sulla sua assiomatizzazione anziché limitarsi all'analisi della coerenza si sono sviluppate in direzioni diverse.

L'approccio assiomatico ha avuto in Pasch lo scopo di permettere una caratterizzazione intrinseca della geometria in grado di renderla indipendente dalla coordinatizzazione e dalla rappresentazione empirica, ossia dall'analisi e dalla fisica. Tarski ha ripreso il criterio di definibilità di Padoa e, con un'opportuna limitazione del linguaggio che rendesse possibile la definizione dei concetti coinvolti (definibilità e trasformazioni sui modelli), ha stabilito gli assiomi della geometria elementare, fondati sui concetti di "essere tra" e di "equidistanza". Tarski ha inoltre dimostrato il seguente teorema [Tarski A., *A Decision Method for Elementary Algebra and Geometry*, 2nd ed. Berkeley, CA: University of California Press, 1951] :

Teorema (teorema di rappresentazione delle geometrie elementare):

ogni modello della geometria elementare è isomorfo alla geometria cartesiana su un campo reale chiuso.

Da questo teorema segue la completezza e la decidibilità della geometria: poiché il concetto di decidibilità e completezza di una teoria in un'altra è invariante per isomorfismi di modelli, se la teoria dei campi reali chiusi è completa e decidibile, allora anche la geometria elementare, che può essere rappresentata nella prima, è completa e decidibile.

La scelta della limitazione del linguaggio ha in Tarski una ragione non logica ma geometrica: la nozione meta-matematica di definibilità copre sia le figure definite in modo sintetico sia quelle della geometria analitica.

1.3. Dalla didattica tradizionale alla didattica sul web

Per la didattica delle matematiche, e forse per tutta la disciplina "tout court", l'E-learning rappresenta l'opportunità di un *paradigm shift* che - per illustrarne la portata- può essere descritto con un estratto del saggio di Roberto Maragliano sulle tecnologie didattiche [56]:

"Sapere la matematica significa non soltanto possedere concetti regole e teorie della disciplina, i suoi contenuti, ma anche saperli usare, saper ragionare con essi, addirittura saper vivere e atteggiarsi di fronte al mondo coerentemente con gli elementi di percezione, conoscenza, azione, sentimento che sono propri dell'universo matematico. Significa anche saper usare gli attrezzi materiali del matematico, avere familiarità con i media di cui questi si serve per esplorare, organizzare, memorizzare e trasmettere il suo sapere. La visibilità pubblica del matematico è sempre stata bassa. Adesso sta crescendo, per merito anche delle tecnologie che egli usa e del loro valore simbolico, su cui a loro volta poggiano le rappresentazioni sociali dei media. Tipico caso di questa svolta -che fa del matematico un personaggio pubblico- è il romanzo di Ian Malcom *Jurassic Park*:

...Questi studiosi avevano rotto con la tradizione di isolamento dei matematici. Per prima cosa si servivano continuamente del computer, poi lavoravano quasi esclusivamente con equazioni non lineari nel campo emergente del cosiddetto caos. Terza cosa, sembravano voler

fare tutto il possibile affinché i loro sistemi matematici descrivessero qualcosa che di fatto esisteva nel mondo reale....”

Con il termine "E-learning" ci si riferisce genericamente all'impiego di Internet nelle attività di formazione, considerandolo spesso come sinonimo di altri termini in reciproca relazione di prossimità: online learning, web based training, open learning.

L'E-learning, che può tradursi come "online learning" ma anche come "online education", "computer mediated education", "distance learning", "Web-based training", a cui possiamo far corrispondere in italiano le espressioni "didattica basata su Internet" o "formazione in rete", coniuga istanze provenienti dalle teorie dell'educazione "Anywhere-Anytime-Anyhow", e, più in generale, si collega agli scenari aperti della globalizzazione.

In realtà "E-learning" non è un neologismo per definire semplicemente la formazione a distanza, ma un termine che sfuma dalla progressiva convergenza di più tendenze, modelli, paradigmi teorici e soluzioni tecnologiche, verso un unico significato, profondamente legato sia a fattori culturali -come il decostruzionismo e l'era dell'accesso- sia a fattori economici -la globalizzazione e il decentramento produttivo. L'E-learning si delinea quindi sempre più come un "bundle" di nuove opportunità metodologiche e didattiche che le tecnologie telematiche e l'Internetworking permettono di sperimentare, con una particolare attenzione ai modelli aperti e flessibili, da un lato, e alla centralità del ruolo del discente, dall'altro.

L'E-learning, a differenza della didattica tradizionale, ha come condizione necessaria la "decostruzione" delle conoscenze e la scomparsa

di ogni riferimento assoluto, sia in termini del "tempo per imparare", che nella presenza fisica del docente. La decostruzione di un testo didattico dà a ciascuno studente la possibilità di costruire la propria interpretazione. Si inserisce così una necessaria componente creativa nel processo di apprendimento.

L'E-learning può vedersi come "il filtraggio di conoscenze distribuite all'interno di uno spazio antropologico virtuale", dove il "virtuale" -nella concezione del filosofo Pierre Levy- è la dispiegazione dell'incompletezza del sistema educativo rispetto alla complessità introdotta dalle tecnologie.

Per dirla in qualche modo come Luhmann, l'E-learning è la conseguenza necessaria del fatto che "Il sistema non è tutto". Se consideriamo l'E-learning come un paradigma essenzialmente tecnologico, viene fuori la conseguenza -paradossale- che l'E-learning risolve problemi che lui stesso ha creato. Non è facile uscire da questa condizione. Una possibilità è che, come tutti i paradossi, l'E-learning è la problematizzazione di nuove metodologie che sono intrinsecamente complesse e più vicine ai sistemi biologici, come prospettato dalle teorie dell'apprendimento enattivo.

"L'apprendimento nei sistemi complessi -sostengono gli studiosi della scuola di Humberto Maturana - è un processo con molteplici esiti, in cui il caso, che non ne permette la previsione esatta, non è un limite ma, anzi, la sua unica possibilità" [57].

Ad un'analisi più pessimista, -ma non meno approfondita- l'E-learning si rivela come la risposta autoriflessiva del sistema educativo all'incapacità di fronteggiare l'aumento di complessità del sovrasis-

ma sociale. Ciò nondimeno il termine "E-learning" sottende la necessità di una trasposizione didattica che trasforma l'incertezza ineludibile dell'epistemologia contemporanea in un'apertura verso un "creative unexpected" (inatteso creativo) necessario a evitare la sclerotizzazione dei processi di insegnamento e di apprendimento.

L'innovazione tecnologica determina il passaggio dalla "società dell'informazione", caratterizzata da un'informazione di massa per lo più unidirezionale, a una "società della conoscenza" in cui ogni singolo individuo è sollecitato alla costruzione attiva delle proprie competenze. Le tecnologie di comunicazione satellitare hanno reso possibili nuovi strumenti per l'educazione a distanza. L'uso della parabola per il broadcasting televisivo è ormai comune, ma la sua collocazione all'interno di un sistema integrato di apprendimento a distanza può rivelarsi un mezzo decisivo. L'educazione a distanza, l'E-learning, coinvolge il trasferimento asincrono di grandi volumi di dati -nell'ordine di 1Mbit/sec- che le reti terrestri DSL non riescono a gestire. Si rende pertanto necessaria l'estensione della rete su canali satellitari. Attualmente la parabola viene adoperata sia per la modalità sincrona di diffusione dei contenuti -videolezioni - che per la modalità asincrona -chat- con protocolli di comunicazione dati a pacchetto, analoghi a quelli usati per l'internetworking delle reti a terra.

Nella cultura anglosassone la formazione a distanza (*FAD*) vanta una solida tradizione. Fin dal secolo scorso varie università e istituzioni "vendevano" formazione veicolata dai servizi postali dell'epoca. Con il passare del tempo, questo tipo di didattica si è diffusa anche in altre regioni, soprattutto nei Paesi in cui -come in Australia, Nuova

Zelanda e America Latina- la densità di popolazione è bassa rispetto all'estensione del territorio. Con il progredire dei mezzi di comunicazione, si è passati dalla posta, al telefono, al fax e, più recentemente, alla distribuzione dei contenuti via TV (sia etere che satellitare); infine si è approdati a Internet. La didattica si è servita dei canali di comunicazione essenzialmente come vettori per la distribuzione di materiali sotto forma di libri, audio cassette, video cassette, cd-rom ecc. In sostanza, però, il metodo non è mutato: l'allievo viene raggiunto dal materiale didattico che può studiare con l'assistenza a distanza di un tutor.

Al termine del processo formativo una sessione di valutazione gli permette, eventualmente, di conseguire una certificazione che si concretizza in un titolo di studio.

La formazione on-line si misura con le unità del cyberspazio, in cui gli oggetti non sono più gli atomi, ma i bit, ovvero le unità di informazione. Nell'E-learning non è più il singolo file a essere una risorsa - una memoria di massa entry-level è dell'ordine di 100 Gigabyte - ma è il valore d'uso dell'informazione ad essere tanto più elevato quanto minore è l'entropia della sorgente. Alla molteplicità distribuita di risorse nel sistema si aggiunge la possibilità, da parte dello studente, di scegliere un ritmo ed uno stile di apprendimento, e la molteplicità dei media utilizzabili. Sul piano pedagogico e didattico questo implica un cambiamento radicale nell'atteggiamento del docente, che dovrebbe trasformarsi sempre più in un "mediatore didattico", [40]. L'emergenza della mediazione didattica è conseguenza dell'aumento esponenziale delle informazioni. Nella pedagogia di una società complessa si rende

pertanto necessaria una figura che filtri la grande quantità di informazione che, oggi, la galassia dei media rende immediatamente disponibile. L'obiettivo dell'educatore -che una volta era far conoscere agli allievi quello che prima ignoravano - è fornire gli strumenti interpretativi -ermeneutici- per distillare le conoscenze dotate di senso, dalle informazioni che permeano l'ambiente. Verso la fine degli anni '90 i nuovi modelli didattici hanno modificato il framework di riferimento dei progettisti.

Oggi si parla sempre meno di CBT (Computer Based Training) e sempre più di WBT (Web Based Training)[14]. Apparentemente, si potrebbe pensare ad una semplice trasposizione delle *teaching-machine* o dei prodotti multimediali in ambienti Internet, che in questo contesto sarebbe usato, prima di tutto come un mezzo di distribuzione -delivery system-. In realtà, l'evidenza che abbiamo raccolto sembra dimostrare che questo non è possibile: l'E-learning ha successo nella misura in cui i discenti considerano Internet come uno spazio antropologico -il cyberspazio- e non soltanto come strumento di comunicazione

Ovviamente, tra le ragioni della diffusione di formazione in rete, non vanno dimenticati i vantaggi organizzativi e logistici, e conseguentemente economici, che un ambiente di apprendimento online garantisce rispetto all'allestimento di un sistema basato su lezioni in aula e tecnologie didattiche tradizionali, vantaggi considerati molto importanti, soprattutto in ambito aziendale.

Al momento sono principalmente due gli ambiti di applicazione dell'E-learning: da una parte vi sono le scuole e le università, dall'altra vi sono le aziende, che sono il segmento del mercato di maggior

fatturato. L'efficienza di un sistema educativo migliora ulteriormente quando sono gli studenti a individuare le proprie esigenze di approfondimento, a formulare gli obiettivi didattici, a scegliere i contenuti, a sviluppare strategie di apprendimento e a controllare e valutare da soli l'apprendimento; aspetti, questi, che complessivamente fanno preferire il termine di "personalizzazione" dell'apprendimento, rispetto a quello più tradizionale di "individualizzazione", che, secondo la pedagogia classica, è il miglior modo, ma anche il più dispendioso, per imparare.

Il Corporate Learning

L'E-learning in azienda è associato alla capacità di identificare, condividere e valorizzare le conoscenze correlate con il *core business* delle imprese. Secondo gli esperti e gli operatori del settore, numerose sono le aziende capaci di offrire soluzioni di valore, ma tra queste nessuna è in grado di conquistare la leadership. Si prevede, tuttavia, che ben presto si verificherà un assestamento, dovuto alla creazione di alleanze tra le imprese e alcune università, grazie alle quali sarà possibile soddisfare esigenze diversificate. In America l'E-learning ha sviluppato l'offerta nel segmento "business" (in particolare, aziende e istituzioni accademiche: due soggetti che dispongono, appunto, di collegamenti adeguati). Le istituzioni accademiche si sono mosse siglando accordi di integrazione -come nel caso di Princeton, Yale e Stanford che, insieme all'Università inglese di Cambridge, hanno costituito una "Corporate University"- oppure fornendo formazione di tipo just-in-time ai privati in modalità business-to-business. E' questo il caso dell'accordo tra la Motorola e la Stanford University per la fornitura di corsi on-line di

tipo *on-demand*.

Sempre sul fronte universitario, il Massachusetts Institute of Technology di Boston -con l'iniziativa MIT OCW- ha messo on-line, a disposizione di chiunque, il materiale didattico dei corsi che l'hanno resa celebre nel mondo: nelle intenzioni dei realizzatori l'obiettivo è la creazione di una comunità distribuita di apprendimento che porta nuovi studenti e nuove joint-venture con le università di altri Paesi.

Al di fuori dall'ambito universitario, troviamo delle realtà ben consolidate quali SmartForce, Click2Learn, DigitalThink, Global Knowledge Network e Netg, mentre il tema dell'E-learning trova sempre maggior spazio all'interno di eventi di richiamo internazionale, come il World Education Market (la fiera mondiale dell'istruzione, della formazione e della conoscenza).

Gli scenari più probabili che si prefigurano nell'ambito dell'E-learning -in un mondo in cui la realtà è sempre più identificata con le sue possibili rappresentazioni- sono simili a quelli già in atto nell'entertainment. La complessità del mercato -e l'asimmetria dei rapporti di forza economici tra il Nord America e il resto del mondo- favorisce la concentrazione e lo sviluppo di "Major" a carattere globale che comprendono università e società di formazione nel frattempo inglobate nell'ERP (*Enterprise Resource Planning*). Il grado di concentrazione del mercato dell'E-learning è guidato dai processi di omogeneizzazione culturale e dipende in qualche modo dall'emergenza delle politiche internazionali.

Nel complesso nel corporate e-learning non sono verificate le condizioni che caratterizzano l'e-learning. Più che un mezzo per imparare,

l'e-learning aziendale è uno strumento necessario per la protezione del know-how delle aziende, che dai sistemi informativi mainframe migrano verso i datawarehouse distribuiti.

L'E-learning in Italia

In Italia, l'E-learning si trova in una situazione di caos primordiale. Superato il primo momento, che potremmo definire di "consapevolezza" (*awareness*), l'industria della formazione europea deve risolvere, in primo luogo, il problema della lingua: molti operatori hanno, infatti, compreso che la qualità della loro offerta dipende anche dalla capacità di renderla disponibile in più lingue. Ecco perché molti stanno attrezzandosi per tradurre, ma soprattutto "localizzare", i contenuti formativi. Sarà necessario, infine, favorire l'interazione all'interno dei corsi, passando dagli ipertesti (corsi privi di link) ad ambienti in grado di riprodurre -con la simulazione- gli ambienti di apprendimento reale.

Le ultime rilevazioni hanno rivelato che il business statunitense legato all'E-learning si aggira intorno ai 66 miliardi di dollari all'anno. In particolare, quello d'oltreoceano è un mercato legato alle aziende del settore privato. Mentre le società americane hanno certamente già compreso quanto l'E-learning sia vantaggioso, quelle europee, invece, stentano ad adottare nuovi standard. Il problema è soprattutto politico-culturale, ma anche infrastrutturale, come ha sottolineato lo stesso commissario europeo all'istruzione Viviane Reding, che ha già avviato un programma di adeguamento. Il *digital divide*, che negli stati uniti sembra legato all'ambiente sociale, in Europa è caratterizzato dalla geografia, se si considera che nei Paesi del nord Europa la

media delle connessioni a Internet hanno prestazioni molto superiori a quelle dei Paesi nell'area mediterranea. Supportati dalla rete Internet sarà possibile prendere parte a lezioni in "virtual classroom", intervenire nella discussione, chiedere chiarimenti e approfondimenti, interagire con il docente e gli altri partecipanti al corso, ricevere un servizio di tutorato. Il vantaggio è adeguare il corso ai propri ritmi di apprendimento ed alle proprie esigenze organizzative (tempo e luogo), sostenendo costi globali più bassi rispetto alla formazione tradizionale.

Lo standard SCORM

L'analisi orientata agli oggetti e la modellazione unificata del software (UML) hanno diffuso i Learning Object nella comunità degli studiosi dell'apprendimento. Come è accaduto per la programmazione orientata agli oggetti, che ha permesso la modellazione su larga scala dei sistemi software, così i Learning Object si propongono di dare una risposta "complessa" al problema, egualmente complesso, della riusabilità dei materiali didattici per la costruzione di strutture cognitive sempre più dense di contenuti.

Questo processo è favorito dall'omologazione degli standard, sia nel campo dello sviluppo dei componenti che in quello dei protocolli di comunicazione.

In modo analogo alla programmazione orientata agli oggetti il paradigma dei Learning Object si fonda su due caratteristiche essenziali: *l'incapsulamento e l'astrazione*; in questo modo l'utilizzatore di un Learning Object ha l'opportunità di riferirsi alle sue sole caratteristiche di utilizzazione (interfaccia) senza essere obnubilato da dettagli

interni che per lui potrebbero essere irrilevanti.

Se invece dell'intero Learning Object se ne considera la sola descrizione simbolica -l'interfaccia -si è più liberi di costruire strutture con la granularità che si vuole.

Per la costruzione di strutture complesse è necessario frammentare i contenuti didattici in unità semplici ma esaustive e riassemblabili; ma probabilmente questo non basta: c'è bisogno di un meccanismo in grado di creare connessioni di tipo omeostatico tra le unità di conoscenza.

Dal campo di ricerca della *knowledge engineering* è derivato il concetto di "*Ontologia*", fino alla tarda modernità appartenuto essenzialmente alla tradizione filosofica. Un'Ontologia è un grafo in cui i nodi sono concetti interconnessi da relazioni epistemiche. A ciascun nodo si associa un Learning Object. In ciascuna Ontologia emerge uno specifico dominio di conoscenza e l'interazione con ciascun utente -docente, tutore o allievo- dà luogo ad una corrispondente interpretazione dell'Ontologia, che -dal filosofo tedesco Gadamer- prende il nome di *ermeneutica*. In sostanza, un'ermeneutica corrisponde a un percorso didattico tracciato nel dominio di conoscenza.

Per la omologazione dei Learning Object si stanno mettendo a punto degli standard (ARIADNE, LOM, IEEE) che fanno capo a una "visione comune" proposta dal consorzio AdlNet. Questa visione si chiama SCORM (Sharable Content Object Reusable Model) e nel campo del WebCt sostituisce lo standard AICC affermatosi in precedenza sul mercato delle tecnologie didattiche.

Lo standard SCORM si propone di rappresentare univocamente un

Learning Object con una lista di nove classi di attributi codificati in XML. Nell'ambito delle architetture Web che aderiscono a questo progetto, un Learning Object è rappresentato da un insieme di attributi scelto tra le classi specificate dallo SCORM.

1.4. I Learning Object

Tra le diverse proposte di definizione di Learning Object, la più rilevante sembra essere quella di Wiley [89].

Definizione 1:

Un Learning Object è un'entità digitale o non digitale che può essere usata, riusata o referenziata.

Si osservi che, per la definizione di Learning Object, l'autore ritiene necessaria, e anche sufficiente, la possibilità di riusare la risorsa didattica. Questa definizione ha dei limiti di natura tecnologica perché in Computer Science tutto è "discreto" e di conseguenza tutto è banalmente riutilizzabile.

Ma non tutto quello che è riutilizzabile è dotato di valenza didattica. Ad esempio, consideriamo un file audio contenente un brano del **Canon in D maggiore** di Johan Pachelbel e un file grafico che contiene l'animazione di una spirale. Secondo la definizione di Wiley, il Canon è un Learning Object, ma per un docente di matematica il Canon in D maggiore potrebbe non esserlo. Se si mettono insieme i due file si percepisce che lo schema del Canone è un ciclo involutorio: i due oggetti diventano didatticamente rilevanti.

L'esempio rileva che i Learning Object - come tutti i sistemi complessi - sono didatticamente interessanti per via di proprietà che non si riscontrano nelle singole componenti.

Da questo esempio si deduce, quindi, che la definizione di Lear-

ning Object proposta da Wiley è alquanto limitata ed è necessaria una definizione migliore. Si ritiene di dover dare una definizione più esaustiva, rispetto a quella fornita da Wiley, partendo dalla considerazione che un Learning Object è formato da un insieme di "asset" che costituiscono i "mattoni" per la sua costruzione, così come gli atomi costituiscono i mattoni della materia. Soffermiamoci quindi sulla definizione di asset.

Definizione 2:

Gli asset sono unità minimali di conoscenza.

Nell'esempio, gli asset sono assimilabili ai singoli file e possono essere paragonati agli atomi che compongono una struttura molecolare. Sotto questo aspetto ci troviamo in accordo con la definizione data in [Hodgin 2003]. Assemblando gli asset con l'intento di favorire specifiche forme di apprendimento, emergono, poi, delle proprietà didatticamente significative. Tali proprietà emergenti consistono in una forma di "chiusura" che viene applicata al Learning Object, costituito, appunto, da un insieme minimale di asset che fa emergere queste nuove proprietà.

Definizione 3:

Un Learning Object è la più piccola unità di apprendimento indivisibile rispetto alla sua valenza didattica.

Naturalmente, questa definizione non è buona per l'implementazio-

ni del learning object in computer science, dove un learning object è considerato essenzialmente una sequenza di valori chiamati "metadata".

Dato l'insieme $I = \{1, \dots, m\}$, fissiamo una sequenza di etichette $\{a_1, a_2, \dots, a_n\}$ e una sequenza di insiemi $(D_i)_{i \in I}$, che chiamiamo insiemi di domini di attributi o insiemi di metadata; per esempio D_i può essere l'insieme di metadata per lo standard SCORM. Chiamiamo *standard Learning Object* una sequenza $(x_i)_{i \in I}$ di valori in un insieme fissato di metadati D_i

Consideriamo un sottoinsieme di attributi AT di $\{a_1, a_2, \dots, a_n\}$. Per ciascun $a_i \in \{a_1, a_2, \dots, a_n\}$ chiamiamo *tipo* di a_i l'insieme D_i . Chiamiamo *classe* un sottoinsieme di elementi di AT .

Per ciascuna classe $C = \{a_{i1}, \dots, a_{im}\} \subseteq AT$, chiamiamo *istanza* una sequenza (a_{ij}) tale che $a_{ij} \in A_{ij} \forall j = 1, \dots, m$. Una *sottoclasse* di C è un sottoinsieme di C . Intuitivamente una classe è uno stampo con cui si forgiavano gli oggetti. Nel nostro caso, l'insieme degli attributi è fissato secondo uno standard di cui rimandiamo alla letteratura (Wiley). Uno standard Learning Object è un oggetto definito selezionando gli attributi che si ritengono rilevanti per l'apprendimento e la didattica.

Diciamo che $d \leq d'$ se d, d' sono rispettivamente istanze di C e C' , $C \subseteq C'$. In altri termini $d \leq d'$ se d è un'istanza di una sottoclasse di d' .

Questa definizione diventa forse più chiara se consideriamo questo esempio:

$$C = \{modulo.obiettivi_didattici, interattività, granularità\}$$

$$C' = \{autore, università, corso.obiettivi_didattici, modulo.obiettivi_didattici, interattività, granularità\}$$

$$d' = (autore, università_di_Dublino_programmazione)$$

Si ha che $d \leq d'$ per ciascuna degli standard learning object costituiti dalle righe della matrice seguente

Babbage	università_di_Dublino	Linguaggio Ada	programmazione	bassa	alta
Babbage	università_di_Dublino	Periferica Input	programmazione	bassa	alta
Babbage	università_di_Dublino	Periferica Output	programmazione	bassa	alta
Babbage	università_di_Dublino	Memoria	programmazione	bassa	alta
Babbage	università_di_Dublino	Unità di calcolo	programmazione	bassa	alta

CAPITOLO 2: L'E-learning

2.1. Un framework di rappresentazione per oggetti 3D nel Web Semantico

Oggi il Web è costruito in modo tale che la maggior parte dei dati che costituiscono i suoi contenuti sono rappresentati affinché possano essere letti e compresi da utenti umani, ma non sono progettati per essere processati da componenti automatizzati (agenti software): ad esempio, un browser può, seguendo precise istruzioni, disporre e visualizzare le informazioni, ma non ha modo di riconoscerle o attribuire loro un particolare significato al fine di elaborarle.

Lo sviluppo della comunicazione attraverso il World Wide Web ha comportato la necessità di rendere attività, quale ad esempio la ricerca di informazioni, adeguate alle esigenze.

In un sistema di gestione della conoscenza, si possono essenzialmente individuare cinque processi fondamentali:

1. Acquisizione della conoscenza,
2. Rappresentazione della conoscenza,
3. Elaborazione della conoscenza,
4. Condivisione della conoscenza,
5. Utilizzo della conoscenza.

I problemi principali a cui si viene incontro quando si naviga nel

Web sono essenzialmente il fatto che la ricerca dell'informazione è basata su sistemi che operano mediante parole-chiave (*keyword*), producendo risultati affetti da "rumore" o da "silenzio" (per *rumore* si intende un certo numero di documenti non pertinenti o con informazioni poco importanti rispetto alla ricerca effettuata, mentre per *silenzio* si intende una ricerca che non restituisce alcun documento). I risultati sono singole pagine Web o singoli documenti, a parte l'esistenza di link che stabiliscono i collegamenti tra i documenti, e l'estrazione di informazioni rilevanti è dominio quasi esclusivo degli essere umani, mediante la lettura dei documenti reperiti. Inoltre i risultati sono sensibili alla differente terminologia adoperata per la ricerca: spesso le parole-chiave utilizzate sono differenti da vocaboli adoperati in documenti rilevanti. Infine la manutenzione dell'informazione è un'attività che richiede notevole investimento di tempo e di risorse umane, soprattutto per documenti di grandi dimensioni.

L'ostacolo principale è che, attualmente, il contenuto del Web non è "*machine accessible*"; o meglio, quando si tratta di interpretare sentenze ed estrarre informazione utile per l'utente, le capacità degli odierni software sono limitate. Un approccio alternativo è rappresentare il contenuto del Web in una forma più facilmente processabile dagli agenti software e adoperare tecniche che sfruttino in modo vantaggioso questa rappresentazione.

Nella visione di Tim Berners-Lee la sfida del futuro è il "Semantic Web", la cui filosofia di base è quella di far sì che il Web divenga uno spazio informativo universale, un'architettura per lo scambio di informazione e conoscenza, un'infrastruttura contenente grande quantità

di informazioni (simile ad un database globale) collegate in modo che sia i codici che i loro significati siano facilmente interpretabili dalle macchine in un ambiente in cui si specifica il significato dei dati e non il modo in cui si intende utilizzarli. Esso involverà gradualmente il web di oggi.

In una recente pubblicazione Tim Berners-Lee [23] ha tracciato una "roadmap" allo scopo di trasformare l'attuale struttura del World Wide Web in un sistema in cui i dati siano comprensibili contemporaneamente dagli umani e dalle macchine. Le linee di sviluppo del Web sono coordinate dal World Web Consortium (W3C, costituito nel 1994 con lo scopo di sviluppare al massimo il potenziale del World Wide Web, fondato e guidato da Tim Berners-Lee) [43], consorzio che sviluppa tecnologie (specifiche, linee guida, software e strumenti) all'interno del Web, definendo protocolli comuni che ne assicurino evoluzione ed interoperabilità, ossia scambio di dati e servizi efficace che permette la comunicazione tra piattaforme hardware e software eterogenee.

Tra gli obiettivi principali del W3C è lo sviluppo del Semantic Web:

"The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation" [24].

Esso rappresenta l'evoluzione del Web da semplice sistema di comunicazione e di recupero di documenti ad un sistema in cui l'informazione è compresa da software in grado di assistere l'utente ad acquisirla ed a rielaborarla, grazie all'impiego di "Metadata" associati alle informazioni, con i quali alle informazioni si possono attribuire nuovi significati espressi in modo tale che siano comprensibili anche ai

calcolatori. Il W3C considera l'ideale evoluzione del Web da "*machine-representable*" a "*machine-processable*", ossia da *rappresentabile dalle macchine* ad *elaborabile dalle macchine*.

Attualmente, quando si reperisce un documento (o un oggetto) sul Web, è possibile che insieme ad esso il server invii alcune informazioni sulla risorsa, quali il suo autore, la data di aggiornamento, eccetera; quindi il Web, come insieme di risorse e di informazioni sulle risorse, cioè "Metadata", è attualmente già a nostra disposizione. I Metadata, quindi, non sono altro che informazioni sulle risorse comprensibili, oltre che dagli umani, anche dalle macchine, e di conseguenza costituiscono un tipo di informazione che può essere utilizzata dagli agenti software, oppure processata in modo distribuito in vari nodi della rete differenziando le funzioni dei componenti del World Wide Web.

Secondo l'idea di Tim Berners-Lee, quando si guarda ad una possibile formulazione di un Web Semantico universale, il principio minimo richiesto è che esso si basi su un modello comune di grande generalità; nella sua visione, il Web Semantico deve essere strutturato secondo un'architettura a livelli (*layer cake*), definendo un modello stratificato che parte dai dati e si sviluppa su vari piani, rappresentando le semantiche dei dati e l'apparato logico che permette la comunicazione all'interno del Semantic Web:

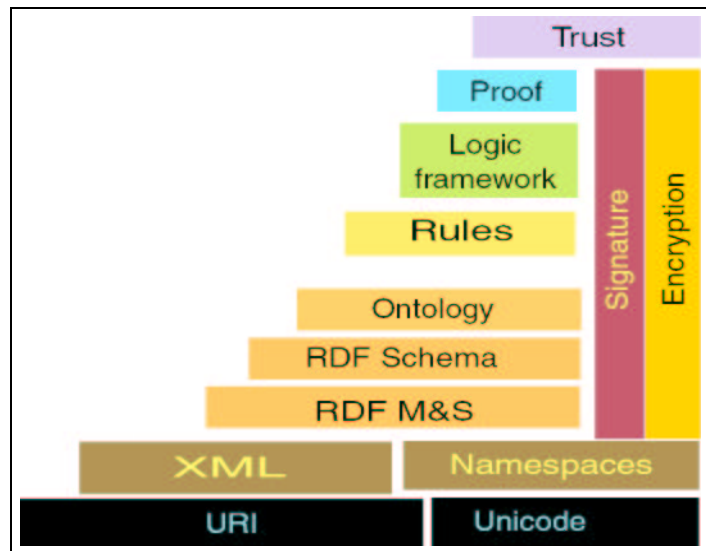


Figura 2.1: *Web Semantico*
<http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html>

Difatti, il Semantic Web può funzionare solo se le macchine possono accedere ad un insieme univocamente strutturato di regole e di informazioni da utilizzare per descrivere un linguaggio comprensibile ad esse.

Il Web Semantico si propone come una soluzione al problema del sovraccarico cognitivo del World Wide Web. Attualmente l'informazione disponibile sul Web risulta difficilmente reperibile perché viene memorizzata sotto forma di stringhe -intese come pura e semplice concatenazione di caratteri- prescindendo del tutto dal suo significato. Il Web Semantico, in sintonia con gli approcci del AI "debole", non si fa carico di cercare un modello computazionale per la rappresentazione dei significati, aspetto questo che riguarda l'AI "forte", ma propone un modello di tipo simulativo. Essenzialmente il web semantico si presenta come una gerarchia di livelli di astrazione che, pur partendo dalla specifica del codice dei caratteri, arriva poi alla definizione di

regole per la rappresentazione dei concetti e la negoziazione dei loro significati sul world wide web.

Lo scopo del Web Semantico è di permettere la costruzione di strumenti avanzati per la gestione di informazioni quali:

- * l'organizzazione della conoscenza in strutture cognitive come le reti semantiche o gli spazi concettuali [Gardenfors 2000];

- * lo sviluppo di strumenti automatici che si occupano della qualità dei dati e di agenti "intelligenti" che estraggono informazioni personalizzate a ciascun utente;

- * tools di ricerca basati non più su parole chiave ma su linguaggi naturali di interrogazione;

- * possibilità di gestire l'accesso personalizzato alle informazioni, o parte di esse, e persino al singolo documento.

Il Web Semantico è una "vision" la cui realizzazione richiede tecnologie esistenti: una soluzione parziale alle sue problematiche essenziali già esiste. [28, 40, 42]. C'è da dire che lo stesso Web Semantico risulta idoneo per la verifica e la messa a punto di nuovi modelli e soluzioni di agenti intelligenti e di applicazioni per il *datawarehousing*.

Come è avvenuto per il World Wide Web, possiamo ritenere che la realizzazione del Web Semantico possa avvenire per auto-organizzazione, in cui le strutture superiori emergono dall'interazione delle singole parti componenti in modo non pre-ordinato.

Descriviamo ora alcuni possibili scenari che illustrano come il Web Semantico può essere utilizzato nell'ambito dell'E-learning.

Esempio 1: I filtri collaborativi

Immaginiamo un docente di matematica che voglia disporre di un buon ambiente virtuale per l'apprendimento della geometria. Attiva una ricerca utilizzando un agente del Web Semantico - installato di recente dal CD Rom nel suo Pc- per chiedere programmi basati sulla "turtle logic" che insegnano a disegnare figure nel piano. L'esperienza accumulata negli anni dal docente consente a questi di evitare di indicare le caratteristiche del software; si limita a scrivere una query del tipo "cerco un ambiente di apprendimento virtuale come il linguaggio Logo" specificando che il suo corso dura due mesi ed ha un budget di 3000 euro.

In breve tempo l'agente recupera un elenco di comunità di apprendimento che hanno provveduto a filtrare le risorse didattiche, designate il letteratura come "Learning Object", tramite un processo collaborativo di valutazione. Il docente scorre la lista e si sofferma sulla comunità di Scuole dell'Infanzia del comune di Reggio Emilia attratto dal prestigio che gode la comunità in oggetto. Allora il suo agente negozia le ricerche con il suo omologo della Comunità e ritorna con un ulteriore elenco di risorse didattiche che sono state valutate positivamente.

Dopo aver provveduto ad acquistare ed utilizzare le risorse che ritiene adeguate al suo scopo, può decidere a sua volta di dar vita ad una nuova comunità di valutazione dando origine a un "learning hub" (Un *hub* o *learning hub* è un nucleo intorno al quale si sviluppa una comunità di apprendimento).

Esempio 2: Virtual Class Room

Un docente di ingegneria strutturale illustra ai suoi studenti un processo di equilibrio del tensore degli sforzi nelle volte a mattoni, servendosi di un modello in scala situato nel cortile dell'Università. Gli studenti possono osservare sul maxi-schermo la distribuzione di un tensore lungo una superficie -che rappresenta una sollecitazione- e azionare dei cursori a rondella per far scorrere le putrelle delle rotaie e variare i punti di tensione del carico.

Per calcolare il punto di equilibrio della superficie, il docente definisce un funzionale da minimizzare e con un software matematico fornisce una soluzione numerica che non è accettabile. Per migliorare la soluzione il professore decide allora di utilizzare un Agente del web semantico per connettere il proprio Pc in un grid computing e negoziare una superficie NURBS con un nodo contenente una repository di modelli 3D. L'agente si serve di un algoritmo di approssimazione per fornire la soluzione che, senza minimizzare il funzionale, simula l'equilibrio con una superficie NURBS.

L'agente fornisce al professore una lista di nodi del grid-computing in cui le NURBS sono state già usate per quel problema. A questo punto l'agente connette il Pc del professore con il nodo remoto del grid computing contenente la NURBS. Gli studenti spostano le putrelle e seguono le evoluzioni della NURBS che simulano le catastrofiche rotture della volta.

Analizzando l'architettura del Web Semantico secondo la vision di Tim Berners Lee, il Web semantico è un modello gerarchico che prevede vari strati, rappresentati nella forma di "torta a strati" o "layer cake", che ricordano un po' l'architettura onion-leaves tipica di tutti i sistemi informatici, introdotti per motivi essenzialmente pragmatici. Uno dei principali vantaggi della stratificazione del Web Semantico è che una volta che uno standard è stato stabilito su un livello, tutti gli altri che vi stanno lavorando -sviluppatori di tool e ricercatori- si adeguano e lo standard accettato "ricade" sul software e sui modelli elaborati per i livelli sottostanti. In questo modo si permette alla struttura del Web Semantico di essere costruita "in itinere" senza dover attendere che l'intera visione si materializzi.

Mentre finora il Web è stato essenzialmente un insieme di standard come l'*HTTP* (*HyperText Transfer Protocol*), l'*HTML* (*HyperText Markup Language*), e *URI* (*Uniform Resource Identifiers*), usati per trasmettere e ricevere documenti ipertestuali e multimediali, nella rappresentazione stratificata di Berners-Lee del Web Semantico, si nota, oltre all'*Unicode*, codifica per l'assegnazione di un unico numero ad ogni carattere, indipendentemente dalla piattaforma, dal programma e dal linguaggio, e all'*URI*, insieme di tutti i nomi e gli indirizzi che fanno riferimento ad una risorsa, il ruolo base giocato dal linguaggio *XML* (*eXtensible Mark-up Language*).

Il primo livello del Web Semantico è praticamente un livello rappresentativo, e coloro che lavorano a questo *layer* devono essenzialmente preoccuparsi di trovare algoritmi, standard e strumenti per la codifica delle componenti che costituiscono i livelli superiori. Una caratteristi-

ca importante del primo livello, ancora lontana dall'essere compiuta, è la mappatura tra una risorsa Web e la locazione geografica della macchina dove viene collocata. L'URI è stato introdotto per far corrispondere univocamente ad ogni risorsa del web semantico il riferimento fisico, o geografico, della sua collocazione.

XML si è sviluppato sulla scia del successo di HTML. Dalla sintassi, complicata, di *SGML* (*Standard Generalized Markup Language*) è stato derivato un linguaggio particolarmente semplice per il World Wide Web. La struttura rigida della sintassi del linguaggio, e la mancanza di una standardizzazione rigorosa, ne hanno richiesto l'estensione ad un formalismo che permette di specificare le risorse sul Web.

In XML il *markup* (insieme dei "marcatori" o "tag", alla base del linguaggio stesso) dei contenuti risulta separato da quello della forma: in questo modo le informazioni non sono semplicemente formattate per facilitarne il reperimento da parte di un utente umano, ma anche facilmente elaborabili da agenti software. Le caratteristiche vantaggiose del linguaggio XML possono essere illustrate con un semplice esempio; la sua sintassi usa tag di inizio e fine che servono a marcare i campi informativi, e marcare in una pagina web le parole "*Gioacchino Rossini*" con il tag `<compositore>` è utile al fine di limitare le ricerche riguardo solo ai documenti di interesse, escludendo eventualmente quelli in cui non si parla del compositore, ma di un suo omonimo. XML è un linguaggio sufficientemente flessibile che fornisce uno strato sintattico comune per rappresentare oggetti, regole e relazioni [64].

Per esempio, XML permette di rappresentare la copia di un libro depositata in una biblioteca:

```

<?xml version="1.0" encoding="UTF-16"?>
<libro>
<autore>Alfred Gray </ autore >
<lingua> inglese </lingua>
<titolo> Modern Differential Geometry of Curves and Surfaces
</titolo>
<editore> CRC </editore>
<anno> 1993 </anno>
<densita_semantica> alta </densita_semantica>
<livello_interattivo> alto</livello_interattivo>
</libro>

```

XML-schema -XMLs- è una raccolta di schemi di sintassi XML che standardizzano l'uso del sottostante livello XML. In generale, XML è un linguaggio per la rappresentazione dei dati che, per essere visualizzati ed elaborati, hanno bisogno di programmi che funzionano solo se il codice XML è "validato" rispetto ad un insieme di regole prestabilito. Nel caso del libro di Alfred Gray, le regole sono rappresentate in figura 2. Nel linguaggio tecnico, le regole che danno luogo allo schema in figura si chiamano *Document Type Definition -DTD-* e sono implementate come una grammatica riconosciuta da un automa finito.

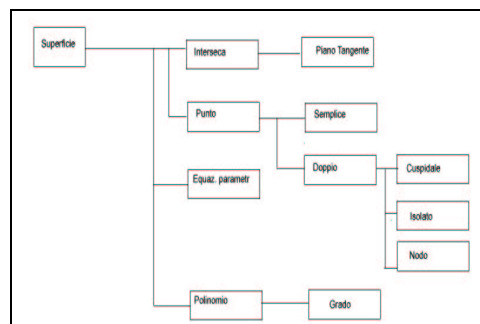


Figura 2.2: DTD per la rappresentazione di un libro in XML

Un ulteriore vantaggio degli schemi XMLs è la componibilità. E' possibile integrare sorgenti di informazione di varia natura dando luogo a strutture informative più complesse. Tecnicamente questo si ottiene usando il costrutto *<include>* per concatenare le DTD e usando il meccanismo del *namespace (XMLns)* che permette di usare lo stesso termine specificandone di volta in volta il significato con un riferimento alla locazione -URI- del documento dove è memorizzato il contesto corrispondente [59]. L'XMLns fornisce un metodo per identificare in maniera non ambigua la semantica e le convenzioni che regolano l'utilizzo delle proprietà identificando l'authority che gestisce il vocabolario.

Uno degli esempi più noti è la *Dublin Core Initiative* [52]; si può utilizzare un namespace XML per identificare in maniera non ambigua lo schema per il vocabolario Dublin Core puntando alla risorsa Dublin Core che ne definisce la semantica.

Per esempio si può comporre lo schema "libro" con lo schema "autore", dando luogo a uno schema più complesso in cui il termine *indirizzo* viene usato sia per indicare l'ubicazione dell'editore che per indicare dove si trova l'autore:

```

<?xml version = "1.0" encoding="UTF-16"?>
<include schemaLocation = "http://www.unina.it/book.xsd" >
<include schemaLocation = "http://www.unina.it/author.xsd" >
<libro: editore
      xmlns:libro = "http://www.unina.it/2000/10/XMLSchema"
      xmlns:autore = "http://www.unina.it/2000/20/XMLSchema" >
<nome> CRC </nome>
<libro indirizzo> Random House, 20 Vauxhall Bridge Road, London
SW1V 25A </indirizzo>
</editore>
<autore> Alfred Gray </autore>
<lingua> inglese </lingua>
<anno> 1995 </anno>
<capitolo numero_capitolo = "1"> Curves in the Plane </capitolo>
<capitolo numero_capitolo = "2">
Studying Curves in the plane with Mathematica
</capitolo>
<capitolo numero_capitolo = "3"> Famous plane curves
</capitolo>
</parte>
      <parte numero_parte = "2"> The Tales </parte>
<autore
      xmlns:libro = "http://www.unisa.it/2000/10/XMLSchema"
      xmlns:autore =" http://www.unisa.it/2000/20/XMLSchema" >
<nome> Alfred Gray </nome>
<nazionalita> Americana <nazionalita>
<autore indirizzo> University of Utah </indirizzo>
<opere>
<opera> Ordinary Differential Equations with Mathematica. </opera>
<opera> Tubes </opera >
</opere>
</autore>
</libro>

```

L'XML, tuttavia, pur consentendo la creazione di documenti strutturati, non gestisce la semantica dei contenuti e non fornisce meccanismi di classificazione, per cui un programma, anche se può riconoscere i contenuti, non è in grado di attribuire loro un significato. L'uso efficace dei metadati richiede che vengano stabilite delle convenzioni per la *semantica*, la *sintassi* e la *struttura*. Di conseguenza, la descrizione tramite marcatori dell'XML necessita di ulteriori strumenti che permettano di definire la semantica dei tag XML: *RDF*.

Sulla base del successo dell'analisi orientata agli oggetti [30] e dell'introduzione delle reti connessioniste [54], il *Resource Description Framework* (RDF) è la novità fondamentale introdotta nel Web Semantico. L'RDF è un linguaggio che specifica formalmente la semantica delle risorse e consente l'interoperabilità tra le applicazioni che si scambiano sul Web informazioni machine-understandable; l'RDF non descrive la semantica ma fornisce una base comune per poterla esprimere, un modello che -a basso livello- viene specificato con la rappresentazione dei metadata. L'RDF può essere usato in una grande varietà di aree applicative, per esempio nella scoperta di risorse per fornire migliori prestazioni ai motori di ricerca, nel catalogare per descrizione il contenuto e le relazioni tra i contenuti disponibili in un particolare sito Web, una pagina, o una libreria digitale, etc.

L'RDF è costituito da due componenti: *Model and Syntax Specification* (W3C Recommendation del 25 Marzo 2002), che espone la struttura fondamentale del modello RDF ("Data Model") e descrive

una possibile sintassi basata su XML, e *RDF Schema* (W3C Candidate Recommendation del 27 Marzo 2000) che espone la sintassi per definire schemi e vocabolari per i metadata.

L'RDF Data Model definisce un modello per descrivere relazioni all'interno delle risorse ed è basato su tre tipi di oggetti:

i) le *risorse*, che possono essere pagine Web, parti o insiemi di esse, o qualsiasi altra cosa individuata da un URI;

ii) le *proprietà*, ossia attributi, relazioni, caratteristiche che descrivono le risorse, che sono individuate da nomi che assumono dei valori;

iii) gli *statement* (o *istanze*), ossia asserzioni costituite da triple (*Soggetto, Proprietà, Oggetto*), dove il Soggetto è una risorsa e l'Oggetto è il valore assunto dalla proprietà, che può a sua volta essere un'altra risorsa.

Il Data Model è rappresentato mediante un grafo, che schematizza la tripla: i nodi del grafo costituiscono le risorse descritte da RDF (gli ovali rappresentano le risorse e i rettangoli le istanze), mentre gli archi, la cui direzione va dal soggetto all'oggetto, etichettati con il nome della proprietà relativa, identificano la proprietà stessa. Così l'espressività dell'RDF include la classe dei diagrammi entità-relazione.

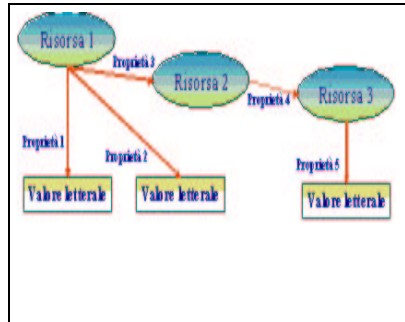


Figura 2.3: *RDF*

Esistono diversi livelli di astrazione in cui concetti espressi in RDF vengono scritti. Al livello più astratto c'è il formalismo delle reti semantiche: ad esempio, per esprimere il concetto "*Azione di un gruppo abeliano di uno spazio vettoriale su un insieme*", nella terminologia delle reti semantiche *KL-ONE* (cfr. ad esempio [51]) graficamente si rappresenta nel modo seguente:

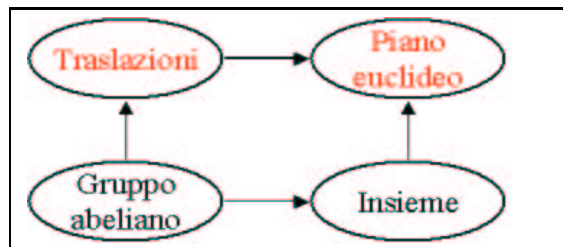


Figura 2.4: *Esempio di rete semantica KL-ONE*

Nel formalismo *KL-ONE*, una rete semantica si costruisce con un linguaggio L munito di

- * un insieme C i cui elementi sono le classi o concetti
- * un insieme T di costanti individuali che esprime le istanze dei concetti

* un insieme F di relazioni binarie che esprime le proprietà o ruoli dei concetti

* una relazione binaria \subseteq in C chiamata sussunzione. Intuitivamente $a \subseteq b$ (diremo che b sussume a) se ogni ruolo per il concetto a è anche un ruolo per b .

Dato un insieme F di ruoli ed una relazione n -aria r , esiste un'estensione F' di F che rappresenta r come composizione di ruoli [cfr. Frixione 2002]. La relazione di sussunzione funge da regola di inferenza per le reti di tipo KL-ONE.

Il formalismo KL-ONE e la relativa definizione di RDF sono alla base per la definizione di linguaggi per costruire il livello immediatamente superiore del Web Semantico.

A livello più basso, mediante il linguaggio XML, utilizzando il costrutto linguistico del namespaces, il concetto espresso in precedenza diventa:

```
<rdf:Description rdf:about="www.dma.unina.it/websemantico/geometria">
<azione_di_gruppo xmlns:geometria =
"http://www.unina.it/websemantico/geometria">
  <geometria:azione>
    <geometria:gruppo> Traslazioni </geometria:gruppo>
    <geometria:insieme> Piano euclideo </geometria:insieme>
  </geometria:azione>
</azione_di_gruppo>
```

Nel caso della didattica, le risorse sono i Learning Objects e l'RDF si propone come un modello di rappresentazione che sostituisce gli attuali "metadata". I metadata sono specificati in XML e sono rappresentati come una sequenza di attributi. Il formalismo RDF -derivato dalla

tradizione logicista dell'intelligenza artificiale - si propone di andare un po' più avanti, sostituendo l'unità di informazione -l'oggetto- con la nozione di *modello*; un modello è un'informazione completa.

Ad esempio consideriamo, ancora, il seguente file di metadata XML compatibile con lo standard SCORM, che rappresenta il libro di Alfred Gray come Learning Object:

```
<?xml version="1.0" encoding="UTF-16"?>
  <autore>Alfred Gray </ autore >
  <lingua> inglese </lingua>
  <titolo> Modern Differential Geometry of Curves and Surfaces
</titolo>
  <editore> CRC </editore>
  <anno> 1993 </anno>
  <densita_semantica> alta </densita_semantica>
  <livello_interattivo> alto</livello_interattivo>
```

Non è difficile comprendere che questo file di metadata specifica un oggetto in modo che -ancorché sia compatibile con lo standard SCORM - è essenzialmente incompleto per uno studioso che volesse conoscere il titolo dei racconti contenuti nel libro. L'inadeguatezza del paradigma dei metadata sopraggiunge perchè noi percepiamo gli oggetti non tanto come un insieme di attributi, ma come una serie di informazioni che all'occorrenza siamo in grado di completare. Nel caso del libro, le informazioni fornite nell'esempio possono essere specificate con delle asserzioni di tipo generale

1. "parte di" è una relazione tra due oggetti
 2. "in alto a" è una relazione tra due oggetti
 3. "a destra di" è una relazione tra due oggetti
- completate dalle seguenti informazioni

- 1.' Un capitolo è una parte del libro
- 2.' La prefazione è una parte del libro che sta immediatamente dopo la copertina
- 3.' Le illustrazioni sono una parte del libro che si trova in alto a destra
- 4.' Vi è una sola illustrazione per ogni pagina.

Queste relazioni, a loro volta, si completano con un insieme di affermazioni del tipo

- 1." Il primo capitolo si chiama "curves in the plane"
- 2." A pagina 5 del libro si trova un'illustrazione di una cicloide
- 3." A pagina 31 c'è la formula di curvatura gaussiana.

Quanto sopra costituisce un'informazione completa e pertanto è un modello. Le asserzioni 1, 2, 3 sono informazioni di carattere generale e corrispondono all'RDF. Le asserzioni 1', 2', 3' costituiscono un insieme di vincoli che completano l'informazione. Le asserzioni 1", 2", 3" sono il vocabolario.

La descrizione RDF, unita ai vincoli e al vocabolario, costituisce l'*RDF-schema* o *RDFs*. L'RDF Data Model non fornisce meccanismi per dichiarare proprietà, o per definire le relazioni tra proprietà ed altre risorse, compito dell'RDF Schema. Per descrivere risorse bibliografiche, per esempio, si usano comunemente attributi descrittivi del tipo "autore", "titolo", "soggetto"; per certificazioni digitali, sono richiesti attributi del tipo "certificazione": il significato dei termini utilizzati in uno statement RDF è importante al fine di stabilire che tipo di elaborazione si deve effettuare sull'informazione.

La dichiarazione di queste proprietà e le loro corrispondenti seman-

tiche sono definite nel contesto dell'RDF mediante l'RDF Schema; uno Schema definisce non solo le proprietà della risorsa (Titolo, Autore, Soggetto, Taglia, Colore, eccetera), ma anche definisce il tipo di risorse che si sta descrivendo (libri, pagine Web, persone, e così via). Esso specifica i meccanismi necessari a definire tali elementi, a definire le classi di risorse con cui possono essere usati, a limitare possibili combinazioni di classi e relazioni, a rilevare violazioni a queste restrizioni.

In definitiva, l'RDF schema serve a definire uno schema di specifiche del linguaggio.

Al di sopra della strato RDF+RDF Schema risiede il livello dedicato all'Ontologia che nell'ambito del Web Semantico viene utilizzato per definire la descrizione formale dei concetti di un dato dominio. In pratica rappresenta il dominio della conoscenza nell'ambito del quale ci si sta muovendo.

RDF ed RDF Schema non consentono di specificare le proprietà delle proprietà; le condizioni necessarie e sufficienti per l'appartenenza alle classi non permettono di specificare meccanismi di ragionamento. Un'Ontologia è una formalizzazione di un determinato dominio di conoscenza che propone di rendere condivisibile e riutilizzabile. Le Ontologie definiscono i termini usati per descrivere e rappresentare un'area di conoscenza e sono comprensibili da umani e da applicazioni che necessitano di condividere il dominio d'informazione (un dominio è proprio un'area di argomento specifico o un campo di conoscenza, ad esempio la medicina, il commercio di automobili, etc). Includono definizioni, utilizzabili da computer, dei concetti basi del dominio, le relazioni tra di essi e i loro attributi; sono di solito espresse in lin-

guaggio logico (è conveniente creare un'Ontologia indipendentemente dal linguaggio con cui verrà in seguito realizzata, per non vincolarla a scelte d'implementazione), dettagliato, accurato, consistente, valido, cosicché distinzioni comprensibili possono essere fatte tra le classi, le proprietà e le relazioni.

Le Ontologie rappresentano, nell'emergente Web Semantico, un modo per rappresentare tout court le connessioni tra concetti e per essere usate da applicazioni e agenti software. Nella rappresentazione della conoscenza -così come nei fenomeni percettivi - è di grande importanza il discernimento della "parte" rispetto al tutto [Arnheim 2000]. Né la topologia né l'algebra riescono a modellare questa capacità che è diventata ambito di studio di una nuova disciplina, la mereotopologia [Casati e Varzi 1999].

Per evitare di usare un termine che non possiede ancora una definizione chiara, in questa sede l'Ontologia sarà identificata con il concetto di "*rete semantica*".

OWL (*Ontology Web Language*) è un formalismo più ricco di RDF in grado di rappresentare meta-proprietà di oggetti descritti da RDF. Il formalismo RDF/RDFs è sufficientemente potente per rappresentare modelli, ma diventa inadeguato quando si richiede di rappresentare relazioni non binarie. RDF non può essere usato per rappresentare "meta-proprietà" quali l'equivalenza di concetti. Pertanto si sono costituiti dei gruppi di ricerca in America ed Europa con il compito di sviluppare un'estensione dell'RDF chiamata DAML-ONT che ha dato origine allo standard euro-americano DAML+OIL. Quest'ultimo è stato assunto come punto di partenza del linguaggio OWL che si prefi-

gura come lo standard d'adozione per lo sviluppo di ontologie nel Web Semantico.

Il gruppo di lavoro sull'*Ontology Web Language* -a tutt'oggi- non ha raggiunto uno standard universalmente accettato. Ciò nonostante, diversi software sono stati elaborati per lo sviluppo di ontologie e per la loro traduzione automatica in XML; questo consente la condivisione di risorse didattiche e la creazione di comunità di apprendimento senza la necessità di conoscere i dettagli di XML. Particolarmente diffuso è il software *Cmap*.

Cmap, abbreviativo di *Cognitive-Maps*, è un tool integrato per la costruzione e la condivisione di risorse didattiche in una comunità distribuita di apprendimento. Cmap è una collezione di strumenti che comprende un editor di ontologie -visualizzato in Figura 4- con la possibilità di esportare in formato XML, un server per la condivisione delle ontologie sul Web, e diversi plugin per la fruizione interattiva delle risorse didattiche. A questi si aggiungono delle funzionalità di ricerca e un canale per la comunicazione sincrona docente-studenti.

Con Cmap è possibile progettare e realizzare esperienze didattiche collaborative in progetti che coinvolgono sia le singole scuole che istituzioni internazionali di istruzione e di ricerca.

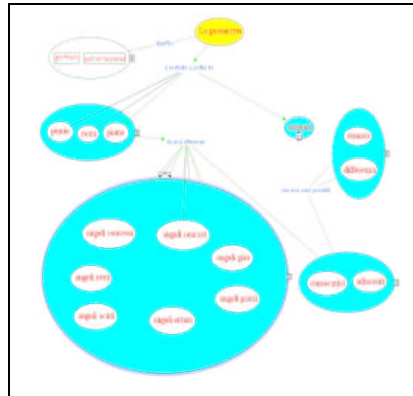


Figura 2.5: Il tool di sviluppo delle ontologie di Cmap.

Lo strato immediatamente superiore del layer cake del Web semantico è quello logico. Fino al livello ontologico c'è solo rappresentazione della conoscenza, non *inferenza* (per *inferenza* si intende un ragionamento deduttivo mediante il quale, partendo da premesse, si ricava, per via logica, una conclusione). E' necessario avere modi di scrivere logica nei documenti per permettere delle cose come, per esempio, regole di deduzione di un tipo di documento da documenti di altro tipo, o la risoluzione di una query convertita da termini sconosciuti a termini conosciuti. Un semplice esempio d'applicazione di questo livello è quando due database, costruiti indipendentemente e poi messi sul web, sono collegati da link semantici che permettono che query su uno siano convertite in query sull'altro.

I sistemi deduttivi normalmente non sono interoperabili, per cui invece di progettare un unico sistema per supportare il ragionamento, si potrebbe pensare di definire un linguaggio universale per rappresentare le dimostrazioni. Una dimostrazione (*Proof*) è una sequenza di formule, ciascuna delle quali è derivata da assiomi, definizione, altre formule;

un "Proof Language" può consentire ad agenti software di scambiarsi delle asserzioni a partire da altre asserzioni ritenute accettabili da parte dell'agente ricevente.

La *Firma Digitale* (*Digital Signature*) è un importante pezzo della struttura del Web Semantico operante su vari livelli del modello. Nella visione di Tim Berners-Lee essa costituisce una struttura che permette di riconoscere parti come credibili o non credibili anche solo in specifici domini. Dal momento in cui si iniziano a realizzare inferenze, bisogna prendere in considerazione la veridicità delle informazioni, e la loro affidabilità. Una fragilità fondamentale dovuta alla totale libertà di espressione del Web Semantico potrebbe esser data dal fatto che non solo le affermazioni contraddittorie non determinano nuova informazione, ma possono essere usate per generare qualunque inferenza.

Il passo successivo allora è creare una rete di affermazioni di affidabilità e fiducia (*Trust*) sulle collezioni, in cui viene espresso il valore di affidabilità delle affermazioni contenute, e attraverso sistemi di sicurezza e crittografia rendere più affidabile il sistema di affermazioni. Le asserzioni RDF dovranno essere contrassegnate con la firma digitale di chi le ha prodotte, lasciando all'utente la facoltà di scegliere di quali firme digitali fidarsi. Il Web of Trust è allora il passo ultimo per permettere la creazione di significato utile, automatico, affidabile.

2.2. L'E-learning e la didattica della geometria

Come anticipato nel capitolo primo, l'apprendimento nei corsi basati sul web ha come condizione necessaria la possibilità, da parte dell'utente, di costruire percorsi didattici personalizzati.

Questa condizione - che poi diventa un processo chiamato "decostruzione" - vale, naturalmente, anche nella didattica della matematica e nell'ambito che andiamo a trattare, la geometria.

Per dare all'utente la possibilità di creare il proprio percorso, consideriamo la geometria come l'intersezione di piani di costruzione e linee di tradizione [57] e vediamo come deve essere costruito il *framework* in cui i concetti debbono ricombinarsi nell'interazione tra ciascun utente e l'Ontologia. Essa deve tener conto dell'evoluzione della disciplina e dei diversi modi di trattare la geometria nei corsi che la utilizzano.

La geometria è attualmente una parte istituzionale dei corsi di Matematica, Fisica, Ingegneria e non solo; studiare geometria vuol dire occuparsi sempre meno di strutture e proprietà astratte del mondo reale e sempre più della costruzione di "modelli" che, secondo i cognitivisti e i logici, sono mondi possibili che interpretano in modo coerente il linguaggio che li descrive.

Dunque, la geometria diventa uno strumento di indagine scientifica di uso generale e territorio condiviso anche da biologi, economisti e, di recente, dagli studiosi della complessità.

La decostruzione di un corso di geometria è un processo che in

principio non termina mai e, nella pratica, è circoscritto dagli interessi di chi si occupa della stesura dell'Ontologia e dalla memoria del computer.

E' interessante osservare che il concetto di decostruzione della geometria può essere chiarito senza usare linguaggi più potenti della geometria stessa. Le tassellazioni aperiodiche, come quelle di Penrose, consentono di disegnare strutture che, al crescere della complessità, non assumono una forma categorica ma si ricombinano in tassellazioni che emergono dalle diverse prospettive con cui l'osservatore guarda la figura:

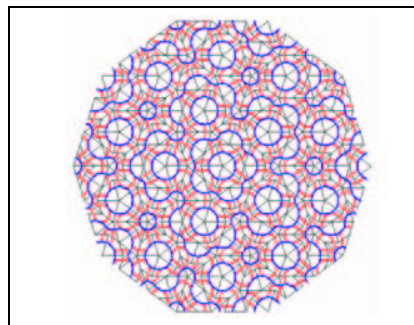


Figura 2.6: *Decostruzione di forme geometriche in una tassellazione di Penrose.*

L'implementazione di un corso on line richiede due tipologie di insegnanti: l'*istruttore*, o *instructor*, che, in qualità di esperto, costruisce l'Ontologia, e il *tutor*, che funge da mediatore didattico. Il ruolo di quest'ultimo è quello di filtrare i Learning Object che l'utente si annette nel processo di decostruzione dei concetti geometrici.

Nel caso della geometria, abbiamo ristretto l'ambito di indagine ai soli aspetti di nostra competenza -matematica, ingegneria, fisica e

applicazioni alle scienze- tralasciando quanto si lega con le discipline umanistiche -storia, arte e antropologia-, di cui non si può negare l'importanza.

Nel ventesimo secolo la geometria si è affermata come lo studio di strutture geometriche caratterizzate da un insieme strutturato di trasformazioni. I nuovi modelli in fisica, ingegneria, biologia, computer science e nella stessa matematica dimostrano che questa definizione, oggi, non regge se si considera fondare la geometria sulla teoria degli insiemi e della logica del primo ordine, rivelatasi incompleta. L'aritmetizzazione della geometria ha permesso ai logici di ridurre i fondamenti della geometria euclidea a quelli dell'aritmetica per la quale, svanite le speranze dei fisici in una *Theory Of Everything*, non resta che fare affidamento alle teorie proposte dalla teoria dei giochi oppure, sbagliando forse di meno, al senso comune.

2.3. Contenuti e metodologie didattiche per un corso di Geometria sul Web: le lezioni sulle quadriche di Luigi Campedelli

Come specificato da Maragliano [66], l'apprendimento testuale opera per astrazione e l'apprendimento multimediale per immersione. Emma Castelnuovo, su ispirazione dello stesso Campedelli, ha seguito quest'ultimo approccio, ritenendo che "far comprendere la matematica forte che regola l'equilibrio di certe forme (nella realtà) è stata la motivazione per lo studio delle quadriche".

La "matematica nella realtà" diventa, nel contesto dell'E-learning, la matematica dei mondi possibili laddove la "realtà" sostenuta dalla Castelnuovo diventa -adesso-il "virtuale"

Descriviamo la trasposizione nell'E-learning di un corso di Geometria tenuto nello stile del professor Luigi Campedelli.

Come anticipato nel Capitolo 1, il processo si articola in tre fasi non necessariamente disgiunte, che coinvolgono le tre tipologie di ruoli (l'istruttore -Instructor-, lo sviluppatore di contenuti -Content Developer-, il tutore -Tutor):

- i) Progettazione del corso - Course Design,
- ii) Sviluppo dei contenuti - Content Development,
- iii) Tutoraggio -Tutoring.

Fase 1: La progettazione del corso.

La progettazione del corso, a carico di un instructor, consiste nell'elaborazione di un learning object con la granularità di un corso monografico e nella codifica ad alto livello nel formalismo RDFs.

Per elaborare il corso si è seguito il testo di Campedelli [33]: a partire dalla definizione di superficie del secondo ordine, si è preparato un asset sulla classificazione delle quadriche a partire dalle intersezioni con il piano improprio.

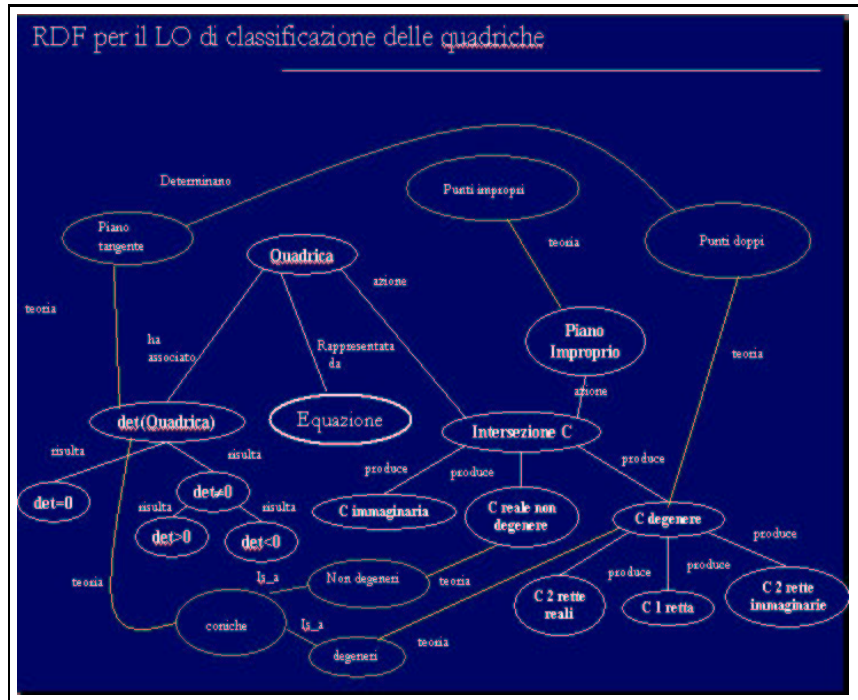


Figura 2.7: *RDF per un Learning Object sull'insegnamento delle quadriche.*

A ogni aspetto teorico relativo alla geometria delle superfici quadriche, corrisponde un asset nel grafo RDF, che affronta in modo esauriente l'argomento affrontato. L'RDF costruito segue la tratta-

zione sulla classificazione delle quadriche, così come illustrata da Luigi Campedelli.

Fase 2: Lo sviluppo dei contenuti.

A differenza che in passato, dove le capacità di un singolo artigiano erano sufficienti a produrre uno strumento didattico, la complessità del RDFs ha richiesto che lo sviluppo degli asset deve essere un attività distribuita e -in qualche modo auto-organizzata- di un gruppo di sviluppatori, ciascuno con strumenti a loro volta "complessi".

Gli asset sono stati elaborati con tre tipi di strumenti

- i) sistemi autore
- ii) ambienti di sviluppo integrati 3D
- iii) tool di software matematico

I sistemi autore sono ambienti di sviluppo integrati che consentono di costruire asset a chi non è un programmatore esperto . Basati sul paradigma della programmazione orientata agli oggetti, i sistemi autore presentano ogni componente come una classe identificata da un insieme di attributi. Lo stato dell'arte nell'ingegneria del software fornisce sistemi autore per coprire l'intera attività di progettazione e sviluppo di contenuti per corsi on-line. Secondo il modello Scorm-CAM, i sistemi autore dedicati al Learning Management System devono includere gli asset in un contenitore -detto "package"- che contiene tutte le informazioni relative al corso, allocate in un file chiamato *manifest*.

Il manifest di un corso è esportabile in tutti gli ambienti che sono compatibili con lo standard SCORM. Nel nostro caso si è usato il *Web Authoring Tool di IBM Lotus Learning Management System*. Questo

sistema autore lavora in due modalità: planner e editor. Nella prima viene definita la struttura del corso e nella seconda si creano gli asset che lo costituiscono.

Per costruire lo strato RDF, sul top del LMS abbiamo inserito nel planner i soli file xml, contenenti le istanze xmls delle RDF degli asset, ciascuno riferito all'Uniform Resource Identifier (URI) della risorsa.

Sulla piattaforma LMS UFO 2 è stato anche montata la versione 1.0 dell'algoritmo di approssimazione NURBS, scritto in Java ed esteso a una classe applet su UFO.

Gli asset 3D sono stati realizzati come di ambienti virtuali di apprendimento e sviluppati con strumenti di software matematico e modellatori 3D, usando il linguaggio VRML come substrato software comune

Mathematica, fin dalla pubblicazione della sua prima versione, nel 1988, ha dimostrato di essere un programma utile per il calcolo scientifico e per la didattica della matematica, in particolare per le computazioni simboliche e grafiche.

L'architettura di Mathematica si compone di server di calcolo - il kernel- e di un client. I protocolli di chiamata variano in base all'architettura della piattaforma. Nel caso di windows il protocollo DDE consente di invocare gli oggetti nel server/kernel tramite un client visuale. Nel caso di Unix/Linux si estendono le librerie grafiche contenute nell'Xserver.

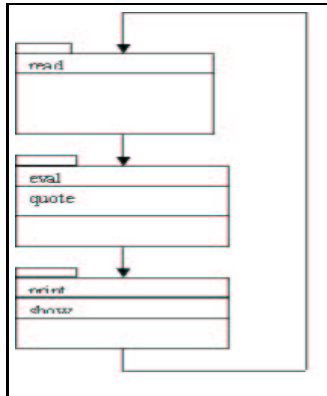


Figura 2.8: *La sessione di Mathematica.*

Mathematica è un ambiente che cresce per accumulazione in una comunità di apprendimento distribuita, pertanto è un linguaggio di programmazione in cui convergono costrutti provenienti dalle comunità degli utenti che lo utilizzano. Vi si trovano elementi del paradigma procedurale, funzionale ed equazionale. Dal paradigma procedurale sono stati ereditati le usuali strutture di controllo. Dal paradigma funzionale è stata ereditato la valutazione delle lambda-espressioni, che consente la definizione di funzioni con tipi di ordine superiore, mentre dal paradigma equazionale è stata ereditata la valutazione per riscrittura contestuale.

Le librerie di Mathematica sono divise in package, che possono essere sia in formato source-code che in eseguibile. E' sufficiente inserire un file sorgente o compilato in una directory di Mathematica per estendere lo scope delle procedure all'ambiente di programmazione. Naturalmente esistono delle istruzioni che consentono al programmatore di proteggere il codice incapsulando le variabili nei package.

Le istruzioni di Mathematica sono eseguite da un interprete dotato di garbage collection per la deallocazione dinamica della memoria.

Mathematica è dotato di un ambiente di sviluppo SDK con chiamate in ambienti CORBA -MathLink-e J2EE, -JLink- che consentono di scrivere estensioni dell'ambiente.

Uno di questi è MathGL3D, che aggiunge a Mathematica un modellatore 3D basato sul modello di Phong.

MathGL3D è stato usato in configurazione pipeline con le primitive grafiche di Mathematica. In questo modo, dapprima si sono calcolati gli shaders delle quadriche a partire dalle equazione

Per ottenere un iperboloide rigato, si sono dapprima considerate due schiere di rette, parametrizzate con Mathematica dall'equazione, poi si è ottenuto lo shader con Mathematica,

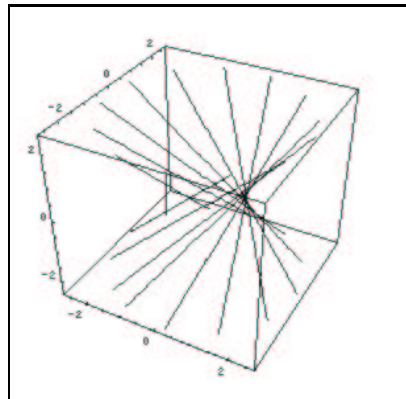


Figura 2.9: Shader di una delle due schiere di rette generatrici dell'Iperboloide ottenuto con Mathematica.

a cui è seguito un rendering con MathGl3d che ha trasformato le rette in cilindri

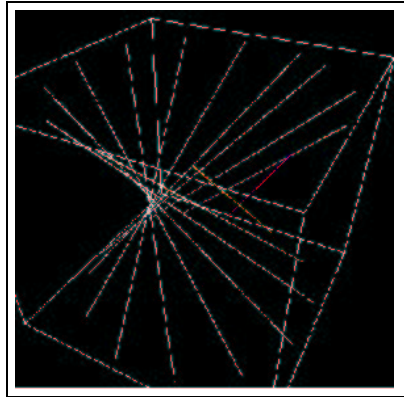


Figura 2.10: *Ray-tracing dell'Iperboloide rigato con MathGL3D.*

Si sono poi importati i due file *dxf* in Caligari TrueSpace, dove si è applicata l'unione booleana alle due schiere di rette generatrici, completando l'iperboloide. Si è eseguito un rendering con un filtro anti-aliasing -per migliorare il rendering ai bordi-, ottenendo questa figura

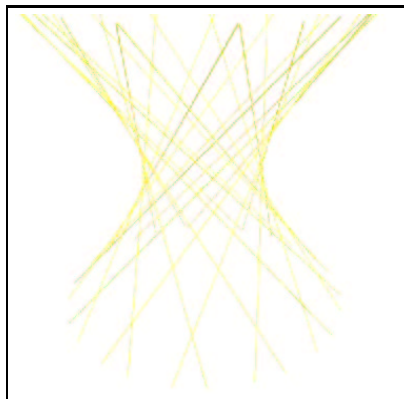


Figura 2.11: *Il rendering con Anti-aliasing dell'Iperboloide rigato con Caligari TrueSpace 3.*

Infine, l'iperboloide è stato esportato nel formato COB e convertito in VRML97. Dopo aver preparato tutti gli oggetti, si è costruito un ambiente 3D con Internet Space Builder, un programma con

interfaccia visuale in grado di costruire ambienti 3D attraverso la combinazione di primitive poligonali.

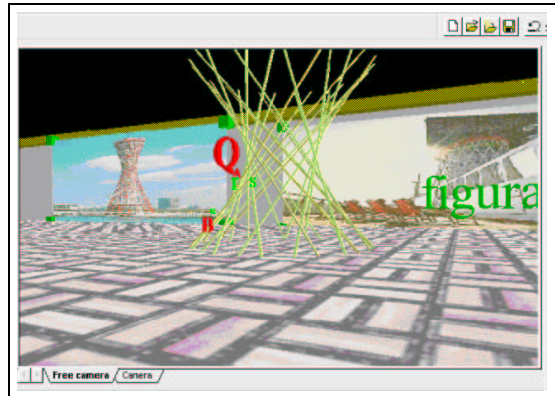


Figura 2.12: L'assemblaggio dell'iperboloide rigato in un ambiente 3D con Internet Space Builder.

Per la modellazione delle quadriche con le Nurbs, si è utilizzato Mathematica per generare un insieme di punti di campionamento e poi - con l'algoritmo di approssimazione Nurbs del Capitolo 1 e si è ottenuta una versione NURBS codificata in un PROTO vrml di Blaxxun

(<http://www.blaxxun.com/vrml/protos/nurbs.wrl>)



Figura 2.13: Approssimazione NURBS di un paraboloide a sella.

Infine, per evitare incompatibilità con i browser vrml che non supportano il proto di blaxxun, si è convertito il proto vrml in insieme di poligoni, che è stato assemblato nella scena con Internet Scene Assembler

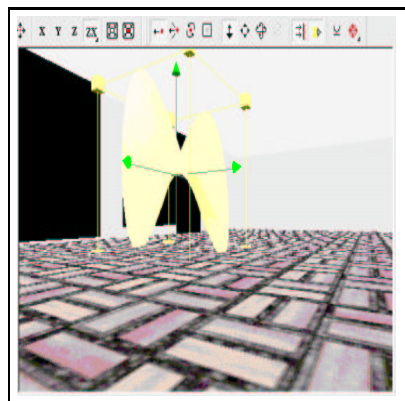


Figura 2.14: *Assemblaggio del paraboloide a sella NURBS in un ambiente 3D con Internet Scene Assembler.*

Fase 3: Il ruolo del tutor.

Per un corso sulle superfici del secondo ordine, si è dapprima costruita l'Ontologia, una RDF per un Learning Object completo per l'insegnamento delle quadriche. Dopo che l'instructor ha preparato l'RDF schema e gli asset, il tutor ha due ruoli essenziali da svolgere,

Piano tangente:	http://www.dma.unina.it/~nicla.palladino/LO/LO.wrl
Punto improprio:	http://www.dma.unina.it/~nicla.palladino/LO/Punti_Impropri.htm
Punti doppi:	http://www.dma.unina.it/~nicla.palladino/LO/Geometria_solida.htm
Quadrica:	$x^2 - y^2 - z^2 = a$
Coniche:	http://www.dma.unina.it/~nicla.palladino/LO/coniche.htm
Degeneri	http://www.dma.unina.it/~nicla.palladino/LO/coniche.htm
Non degeneri	http://www.dma.unina.it/~nicla.palladino/LO/coniche.htm

2.4. *Modelli di E-learning con filtri collaborativi*

Partendo dalle definizioni del Capitolo 1, è possibile costruire modelli computabili di Learning Object sufficientemente potenti per le implementazioni del livello applicativo.

Riprendiamo la definizione di *computable Learning Object* come coppia di operatori algebrici H e K data nel capitolo 1. Si è visto che uno standard Learning Object può considerarsi come una sequenza di *feature* $(d_i)_{i \in I}$ dove I è un insieme di caratteristiche standard. Con questa definizione si possono costruire modelli per applicazioni di E-learning complesse che, per avere senso, debbono essere preceduti da una descrizione "tout court" del processo di apprendimento a distanza.

L'E-learning coinvolge essenzialmente tre tipologie di attori non necessariamente disgiunte:

- i fornitori di contenuto (*Content provider*),
- i fornitori di servizi di E-learning (*Learning service provider*)
- le comunità di apprendimento (*Learning community*).

I *fornitori di contenuto* possono essere le università e le aziende che fanno formazione che, come descritto nel capitolo 1, possono organizzarsi in comunità virtuali distribuite; il loro compito è quello di produrre materiale didattico -appunti, esercizi, note - e allocarlo nelle repository web, come ad esempio la MIT OCW (<http://ocw.mit.edu>), oppure le pagine web dei corsi del Geometry Center.

I *Learning service provider* possono essere le stesse università, come accade per la Global Virtual University, oppure società di distribuzione di Learning Object, come i *Learning Mart*, tra cui la norvegese GRID-Arendal (<http://www.grida.no>) e l'italiana Sfera (www.sfera.it).

Infine, le comunità di apprendimento; esse possono vedersi come l'estensione del core business dell'università, in questo caso costituiscono gli studenti on-line e tradizionali, oppure sono le reti di scuole, oppure ancora, nel caso della formazione professionale, le *community of practice*.

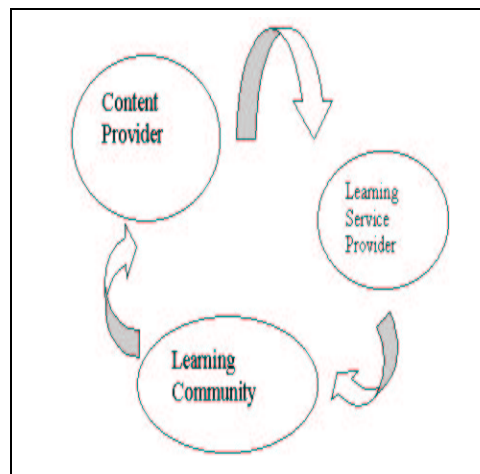


Figura 2.16: Un framework generalizzato per l'E-learning in the large.

La complessità del sistema rende necessario che ciascuna parte sia interfacciata con un filtro che elimina i Learning Object “spuri”, fornendo alla comunità di apprendimento i soli Learning Object che rispondono agli obiettivi didattici. Questi filtri sono chiamati *Web Recommender*, e si dividono in tre categorie: *information filtering*, *collaborative filtering*, *content-based filtering*.

La complessità del Web rende sempre più difficile all’utente la possibilità di trovare informazioni di qualità che siano rilevanti per i suoi interessi: una soluzione è il *recommender system*. Un *recommender system* (o *sistema di raccomandazione*) è un sistema che prende in input il profilo dell’utente -ovvero quali sono gli interessi dell’utente- e poi gli raccomanda le risorse web; esso è quindi, fondamentalmente, un filtro adattivo che legge lo schema di navigazione dello studente e gli fornisce quanto lo studente non ha ancora visto. In base al tipo di filtro usato, il modello di E-learning assume una particolare configurazione.

Per comprendere le condizioni in cui si sta attuando l’E-learning, giova forse la citazione di una parte -sempre meno inattuale- del celeberrimo articolo "*As we may think*" scritto da Vannevar Bush al MIT verso la fine della Seconda guerra mondiale:

...Professionally our methods of transmitting and reviewing the results of research are generations old and by now are totally inadequate for their purpose. If the aggregate time spent in writing scholarly works and in reading them could be evaluated, the ratio between these amounts of time might well be startling. Those who conscientiously attempt to keep abreast of current thought, even in restricted fields, by close and continuous reading might well shy away from an exami-

nation calculated to show how much of the previous month's efforts could be produced on call. Mendel's concept of the laws of genetics was lost to the world for a generation because his publication did not reach the few who were capable of grasping and extending it; and this sort of catastrophe is undoubtedly being repeated all about us, as truly significant attainments become lost in the mass of the inconsequential.

The difficulty seems to be, not so much that we publish unduly in view of the extent and variety of present-day interests, but rather that publication has been extended far beyond our present ability to make real use of the record. The summation of human experience is being expanded at a prodigious rate, and the means we use for threading through the consequent maze to the momentarily important item is the same as was used in the days of square-rigged ships.

- **Learning Object Information Filtering (LOIF)**

Nel *Learning Object information filtering*, il sistema di raccomandazione è basato essenzialmente sul comportamento dell'utente, pertanto la predizione del comportamento futuro dell'utente si basa solo sul suo comportamento passato. Un esempio di sistema LOIF è il sistema esperto della piattaforma IWT.

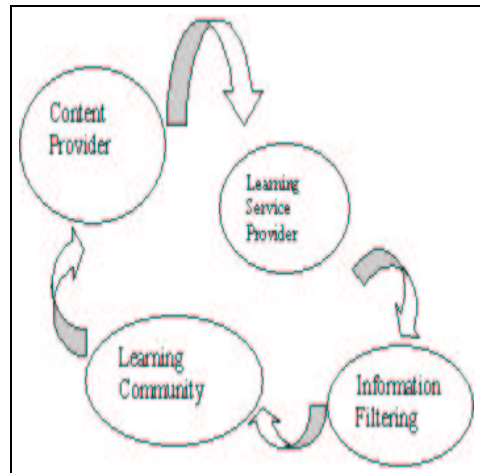


Figura 2.17: *Information Filtering.*

• Learning Object Collaborative Filtering

Nel filtraggio collaborativo, il sistema compara gli utenti per trovare utenti simili, come nel caso del sito *Amazon*. Il filtro collaborativo si basa sulla seguente inferenza

Users who choose Resource A also choose Resource B, C, and D

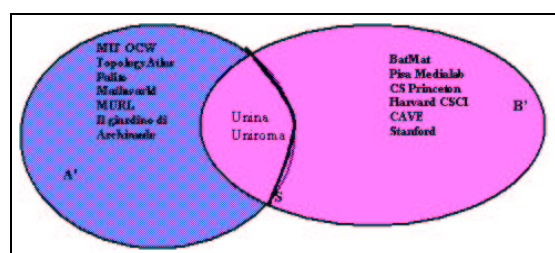


Figura 2.18: *Esempio di filtraggio collaborativo.*

Per chiarire il problema, supponiamo che due studenti *A* e *B* sono entrambi interessati all'argomento *T*. Supponiamo che *A* e *B* abbiano esplorato lo stesso insieme di risorse *S*, con la differenza che *A* ha visto

i Learning Object $S \cup A'$ e B ha visto i Learning Object $S \cup B'$, con A' e B' Learning Object non condivisi.

E' chiaro che le tecniche di filtraggio collaborativo sono più potenti di quelle della semplice information filtering, perché possono raccomandare pagine che l'utente non ha ancora visto. Comunque, le tecniche di filtraggio collaborativo hanno alcune limitazioni: nella maggior parte dei sistemi è richiesta la votazione esplicita tramite un sistema di ranking. In questo modo non si raccolgono, in generale, voti sufficienti per ottenere risultati significativi. Un altro svantaggio è che le scale sono soggettive. Può accadere che un corso di analisi matematica abbia ricevuto un voto da 2 a 10 e ci si chiede se i criteri di valutazione sono simili a quelli di chi lo deve usare. Un altro problema è che le risorse con un numero basso di voti possono essere ingiustamente penalizzate. Accade infatti che un nuovo learning object, siccome è nuovo, non accumula un numero sufficiente di votazioni per essere considerato una risorsa didattica di alta qualità.

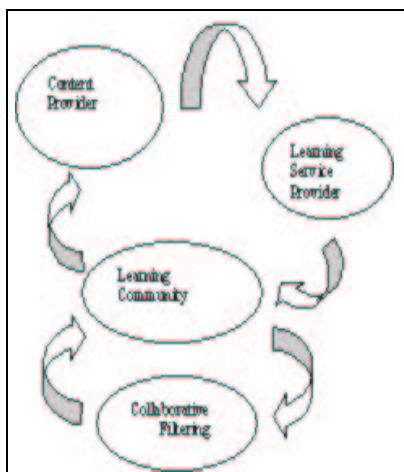


Figura 2.19: Collaborative Filtering.

- **Content-based Filtering**

Nel filtraggio basato sui contenuti, la similarità dei LO è basata sul loro contenuto. Si usano diverse tecniche, dal conteggio delle occorrenze di parole chiave alla valutazione della similarità semantica.

- **Approccio Ibrido**

I problemi delle tecniche di *information filtering*, *collaborative filtering*, o *content-based filtering* hanno determinato un approccio ibrido al problema del *learning object web recommending*.

2.5. Principali tecniche di Web Recommending

2.5.1. Prodotto scalare di attributi

Il prodotto scalare di *features* (attributi) si ottiene definendo l'utente come una coppia ordinata (F, S) , dove F è una sequenza di attributi rilevanti $(f_i)_{i \in I}$ che caratterizzano l'utente e S è un insieme di Learning Object conosciuti dall'utente. Dati due utenti (F, S) e (F', S') , si assume il coseno dell'angolo del prodotto scalare $F \bullet F'$ come grado di similarità tra i due utenti.

Quindi, per ciascuna coppia di sequenze di features F ed F' , si ha

$$R(F, F') = \frac{F \bullet F'}{\|F\| \|F'\|};$$

se F e F' si considerano come vettori di variabili casuali, $R(F, F')$ coincide con il *grado di correlazione lineare di Pearson*. Siccome $R(F, F') \in [-1, 1]$, i due utenti F ed F' sono tanto più simili quanto più $R(F, F')$ si avvicina ad 1. Fissato $\lambda \in [-1, 1]$ come valore di soglia, si ottiene il seguente algoritmo:

Filtraggio collaborativo di Learning Object tramite grado di correlazione
--

```
for i=1 to N do
  for j=1 to N do
    if  $R(F[i], F[j]) > \lambda$  then
       $S[i] = S[i] \cup S[j]$ 
       $S[j] = S[i] \cup S[j]$ 
    end if
  end for
end for
```

2.5.2. La Graph HIT distance

Per risolvere il problema della rilevazione delle feature degli utenti, si è determinata una similarità tra gli utenti U e U' in base alla distanza dei link segnati da U ed U' nel loro grafo di Iperlink. In questo caso la similarità non è definita tra utenti, ma tra i nodi di un grafo.

L'idea di base è la seguente, ripresa da Chen [1]. Quando un utente cerca una specifica informazione, potrebbe chiedersi "Quali altre pagine possono essere state esplorate da persone con le stesse esigenze?"

Assumiamo, come nel caso precedente, che gli utenti A e B sono entrambi interessati alla disciplina T . Supponiamo che entrambi abbiano fruito dei Learning Object $\{P1, P2, P3\}$, che chiameremo *common set*, che contengono informazioni su T .

Supponiamo inoltre che A e B hanno visto, ciascuno, altri LO dif-

ferenti: A ha fruito dei Learning Object $P4$ e $P5$ e B ha fruito di $P6$ e $P7$. Tutti questi contengono informazioni su T e li collezioneremo in un insieme chiamato *different set*.

In questo caso il filtro collaborativo analizza la similarità tra i Learning Object del *common set* e in seguito esplora il *different set* per cercare elementi simili a qualche elemento nel *common set*. Se questo è il caso, il *common set* è esteso e il nuovo Learning Object è fruibile a tutta la comunità di apprendimento.

Partiamo dalla definizione della *distanza Graph Hit (HG)*. HG è definita nel modo seguente:

Definizione 1:

Sia $G = (V, E)$ un grafo e $P, P', P'' \in V$. Poniamo

$$HG(P, P') = \begin{cases} 0 & \text{se } PP' \in E \\ n & \text{se } P''P' \in E \text{ e } HG(P, P'') = n - 1 \end{cases}.$$

HG si chiama *distanza Graph Hit* o *Graph-Hit distance*.

CAPITOLO 3: Le superfici NURBS

3.1. 3D-Resource brokering con algoritmi basati su Nurbs

Gli algoritmi di resource brokering devono soddisfare le specifiche piuttosto critiche per i *core services* nel modello a clessidra del Grid [49]. Il rendimento - *throughput* - dell'intero sistema è tanto maggiore quanto più piccolo è il bundle di standard presenti nei core services. Per garantire l'adattività dei core services sia verso l'alto -nella direzione delle risorse- che verso il basso- nella direzione delle applicazioni- la risposta degli algoritmi implementativi non deve essere di tipo "tutto o niente" ma deve essere in qualche modo flessibile. L'E-learning fa un uso estensivo della grafica 3D -come nel caso della modellazione 4D- e questo ha determinato la creazione di numerose repository di modelli grafici -superfici, curve, animazioni- che, per la mancanza di tool di Service Discovery, sono difficilmente accessibili a chi li cerca e, pertanto, inducono alla costruzione di ulteriore materiale che, a sua volta ignorato dai più, diventa un processo per diffondere il caos sull'intero sistema. In questo capitolo proponiamo un algoritmo utilizzabile per il 3D-resource brokering in cui le risorse sono negoziate con processi di discovery "semanticamente ricchi". Per minimizzare gli standard

del core service usiamo una shell semantica -costruita sul modello del layer cake di Tim Berners Lee - in cui XML è il solo standard di comunicazione. XML è facilmente integrabile con gli standard *PVM* e *MP*.

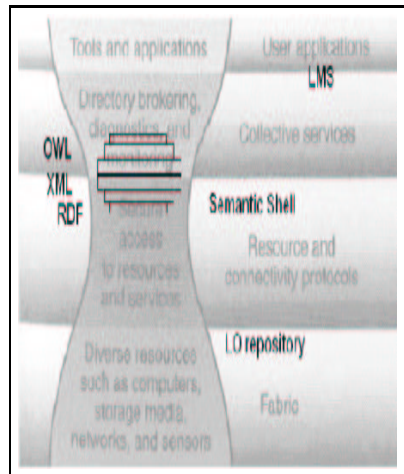


Figura 3.1: Integrazione di una shell semantica nell'Hourglass model.

Descriviamo un service broker che negozia le richieste di Learning Object da parte di un *learning management system* intelligente. La *shell semantica* fornisce un substrato comune XML che connette il *service broker* con il *bottom level* della clessidra -tramite il Resource Description Framework (RDF) delle superfici- e con il Web Ontology Language usato dai LMS dotati di modello utente [35]. L'integrazione con i protocolli PVM e MPI è sufficientemente garantita dalla progressiva affermazione degli standard *Open Grid Software Architecture* (*OGSA*).

3.1.1. Funzioni di base B-Spline

Sia U una sequenza strettamente crescente di m numeri reali u_i , detti *nod*i

$$u_0 < u_1 < \dots < u_m.$$

Si definiscono *funzioni di base B-Spline* di grado h sul vettore dei nodi U , le funzioni $N_{i,h}(u)$ costruite mediante la formula ricorrente

$$N_{i,h}(u) = \frac{u-u_i}{u_{i+h}-u_i} N_{i,h-1}(u) + \frac{u_{i+h+1}-u}{u_{i+h+1}-u_{i+1}} N_{i+1,h-1}(u)$$

con

$$N_{i,0} = \begin{cases} 1 & \text{se } u_i \leq u < u_{i+1} \\ 0 & \text{altrimenti.} \end{cases}$$

Si noti che, poichè per calcolare la funzione di base $N_{i,1}(u)$ si usano le funzioni $N_{i,0}(u)$ e $N_{i+1,0}(u)$ che risultano diverse da zero rispettivamente negli intervalli $[u_i, u_{i+1})$ e $[u_{i+1}, u_{i+2})$, allora la funzione $N_{i,1}(u)$ è diversa da zero nell'intervallo $[u_i, u_{i+2})$. Allo stesso modo, poichè per calcolare la funzione di base $N_{i,2}(u)$ si usano le funzioni $N_{i,1}(u)$ e $N_{i+1,1}(u)$ che risultano diverse da zero rispettivamente negli intervalli $[u_i, u_{i+2})$ e $[u_{i+1}, u_{i+3})$, allora $N_{i,2}(u)$ è diversa da zero nell'intervallo $[u_i, u_{i+3})$.

In sintesi, si può concludere che la funzione di base $N_{i,h}(u)$ è diversa da zero nell'intervallo $[u_i, u_{i+h+1})$ (proprietà del *supporto locale*).

L'insieme delle funzioni di base B-Spline gode, inoltre, di altre proprietà rilevanti:

- $N_{i,h}(u)$ è un polinomio di grado h .
- Per ogni i , h ed u , $N_{i,h}(u)$ è non negativa.
- La somma di tutte le funzioni di base B-Spline diverse da zero di grado h sull'intervallo $[u_i, u_{i+1})$ è uguale ad 1 (proprietà di *partizione dell'unità*).
- La funzione $N_{i,h}(u)$ è continua di classe C^h .

3.1.2. Curve B-Spline e curve NURBS

Si definisce curva B-Spline di grado h una funzione la cui rappresentazione parametrica in R^2 è

$$C(u) = \sum_{i=0}^{n-1} p_i N_{i,h}(u)$$

con $u \in [a, b] \subseteq R$, dove

- u è il parametro della rappresentazione parametrica;
- $p_i = (x_i, y_i) \in R^2$, $i = 0, \dots, n$ sono detti punti di controllo;
- $N_{i,h}(u)$ sono le funzioni di base B-Spline sul vettore dei nodi $U = (u_0, \dots, u_m)$.

Sussiste poi una relazione importante che lega il numero dei nodi m , il grado della curva h ed il numero dei punti di controllo n ; deve infatti essere:

$$m = n + h + 1.$$

Le curve B-Spline godono di alcune importanti proprietà

- Una curva B-spline $C(u)$ è una curva a tratti, ogni componente della quale è una curva di grado h ;
- Una curva B-Spline è contenuta nella parte di piano convessa formata dal polinomio i cui vertici sono i suoi punti di controllo. Nello specifico, se u appartiene all'intervallo $[u_i, u_{i+1})$, allora $C(u)$ è contenuta nella parte di piano convessa delimitata dai punti di controllo $p_{i-h}, p_{i-h+1}, \dots, p_i$;
- Se si cambia la posizione del generico punto di controllo p_i , allora il cambiamento modifica solo la parte di curva $C(u)$ contenuta nell'intervallo $[u_i, u_{i+h+1})$;
- $C(u)$ è una funzione continua di classe C^h ;
- Se ad una curva B-Spline è applicata una trasformazione affine, il risultato può essere ricostruito a partire dall'immagine affine dei suoi punti di controllo. In pratica, se si vuole applicare una trasformazione geometrica od affine ad una curva B-Spline, è sufficiente applicare la trasformazione ai suoi punti di controllo (cosa più semplice), e calcolare la curva B-Spline definita da questi nuovi punti.

I punti di controllo possono essere rappresentati anche in coordinate omogenee; riscrivendoli come vettore colonna, si ha, per il generico punto p_i

$$p_i = \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}.$$

Poichè moltiplicando le coordinate (omogenee) di un punto per uno stesso numero reale diverso da zero, la sua posizione non cambia, possiamo moltiplicare le coordinate di p_i per un reale w_i che chiameremo peso, ottenendo

$$p_i^w = \begin{bmatrix} w_i x_i \\ w_i y_i \\ w_i \end{bmatrix}.$$

Sostituendo le nuove coordinate del punto di controllo nell'equazione della curva B-Spline, si ha

$$C^w(u) = \sum_{i=0}^{n-1} p_i^w N_{i,h}(u) = \sum_{i=0}^{n-1} N_{i,h}(u) \begin{bmatrix} w_i x_i \\ w_i y_i \\ w_i \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^{n-1} N_{i,h}(u) w_i x_i \\ \sum_{i=0}^{n-1} N_{i,h}(u) w_i y_i \\ \sum_{i=0}^{n-1} N_{i,h}(u) w_i \end{bmatrix}$$

A questo punto si può riconvertire l'equazione in coordinate cartesiane, dividendo $C^w(u)$ per la sua terza coordinata:

$$C(u) = \begin{bmatrix} \frac{\sum_{i=0}^{n-1} N_{i,h}(u)w_i x_i}{\sum_{i=0}^{n-1} N_{i,h}(u)w_i} \\ \frac{\sum_{i=0}^{n-1} N_{i,h}(u)w_i y_i}{\sum_{i=0}^{n-1} N_{i,h}(u)w_i} \\ 1 \end{bmatrix} = \sum_{i=0}^{n-1} \frac{N_{i,h}(u)w_i}{\sum_{j=0}^{n-1} N_{j,h}(u)w_j} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix};$$

In definitiva, si ha

$$C(u) = \frac{1}{\sum_{i=0}^{n-1} N_{i,h}(u)w_i} \sum_{i=0}^{n-1} N_{i,h}(u)w_i p_i$$

La curva $C(u)$ di grado h definita dai punti di controllo p_i , dal vettore dei nodi U e dai pesi w_i , $i = 0, \dots, n-1$, è detta *Curva NURBS*.

Naturalmente, se tutti i pesi sono uguali ad 1, la curva NURBS si riduce ad una curva B-Spline.

Si può indicare

$$R_{i,h}(u) = \frac{N_{i,h}(u)w_i}{\sum_{j=0}^{n-1} N_{j,h}(u)w_j},$$

dove le funzioni $R_{i,h}(u)$ sono dette *funzioni B-Spline razionali* di grado h sul vettore dei nodi U con pesi w_j , e riscrivere l'equazione della curva NURBS nel modo seguente

$$C(u) = \sum_{i=0}^{n-1} R_{i,h}(u)p_i.$$

Le curve NURBS e le funzioni B-Spline razionali godono di alcune proprietà:

- $R_{i,h}(u)$ è una funzione razionale di grado h in u ;
- per ogni i ed h , $R_{i,h}(u)$ è non negativa;
- su ogni intervallo $[u_i, u_{i+1})$, al più $h+1$ funzioni B-Spline razionali di grado h sono diverse da zero:

$$R_{i-h,h}(u), R_{i-h+1,h}(u), R_{i-h+2,h}(u), \dots, R_{i,h}(u);$$

- la somma di tutte le funzioni B-Spline razionali di grado h diverse da zero nell'intervallo $[u_i, u_{i+1})$ è 1 (proprietà di partizione dell'unità);
- la funzione $R_{i,h}(u)$ è continua di classe C^h ;
- se $w_i = c \forall i$, (dove c indica una costante diversa da zero), allora $R_{i,h}(u) = N_{i,h}(u)$, cioè le funzioni di base B-Spline sono casi speciali delle funzioni razionali.

3.1.3. *Superfici B-Spline e superfici NURBS*

Assegnati $m \times n$ punti di controllo $p_{i,j} = (x_i, y_i, z_i) \in R^3$, un vettore dei nodi $U = (u_0, u_1, \dots, u_p)$, un vettore dei nodi $V = (v_0, v_1, \dots, v_q)$, un grado h ed un grado k , si può definire, a partire da tali parametri, una *Superficie B-Spline*, data dalla seguente rappresentazione parametrica:

$$S(u, v) = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} N_{i,h}(u) N_{j,k}(v) p_{i,j}$$

dove

- u e v sono i parametri della rappresentazione,
- le $N_{i,h}(u)$ sono le funzioni di base B-Spline di grado h definite sul vettore dei nodi U ,
- le $N_{j,k}(v)$ sono le funzioni di base B-Spline di grado k definite sul vettore dei nodi V .

Come per le curve B-Spline, anche nel caso delle superfici sussistono delle relazioni che legano tra loro il grado, il numero dei punti di controllo ed il numero dei nodi; deve infatti essere

$$p = m + h + 1$$

$$q = n + k + 1 .$$

Le superfici B-Spline godono di alcune proprietà importanti:

- $N_{i,h}(u)$ e $N_{j,k}(v)$ sono non negative per ogni h, k, i, j, u e v ;
- la somma di tutte le $N_{i,h}(u)$ e $N_{j,k}(v)$ è 1 per $u, v \in [0, 1]$;
- se $(u, v) \in [u_i, u_{i+1}] \times [v_j, v_{j+1}]$, allora la superficie $S(u, v)$ giace nella parte di spazio convessa delimitata dai punti di controllo $p_{l,q}$ dove $i - h \leq l \leq i$ e $j - k \leq q \leq j$;
- $N_{i,h}(u) N_{j,k}(v)$ vale zero se $(u, v) \notin [u_i, u_{i+h+1}] \times [v_j, v_{j+k+1}]$;

- $S(u, v)$ è continua di classe C^h nella direzione u , ed è continua di classe C^k nella direzione v ;
- se ad una superficie B-Spline è applicata una trasformazione affine, il risultato può essere ricostruito a partire dall'immagine affine dei suoi punti di controllo. In pratica, se si vuole applicare una trasformazione geometrica od affine ad una superficie B-Spline, è sufficiente applicare la trasformazione ai suoi punti di controllo (cosa più semplice), e calcolare la superficie B-Spline definita da questi nuovi punti.

Analogamente a quanto fatto nel caso bidimensionale, si possono definire anche le *Superfici NURBS*, date dalla rappresentazione parametrica

$$S(u, v) = \frac{1}{\sum_{i=0}^{n-1} N_{i,h}(u)w_i} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} N_{i,h}(u)N_{j,k}(v)w_{i,j}p_{i,j}$$

dove $w_{i,j} \in R$, $i = 0, \dots, m-1$, $j = 0, \dots, n-1$ sono i pesi.

3.1.4. B-Spline Approximation nel senso dei minimi quadrati

Il problema dell'approssimazione mediante superfici B-Spline può essere formulato come segue:

assegnati $m \times n$ punti $Q_{i,j} = (a_{i,j}, b_{i,j}, c_{i,j}) \in R^3$, con $i = 0, \dots, m-1$ e $j = 0, \dots, n-1$, bisogna determinare una superficie B-Spline

$$S(u, v) = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} N_{i,h}(u) N_{j,k}(v) p_{i,j}$$

di gradi h e k con punti di controllo opportuni $p_{i,j} = (x_{i,j}, y_{i,j}, z_{i,j}) \in R^3$, per $i = 0, \dots, m-1$ e $j = 0, \dots, n-1$, ed opportuni vettori dei nodi $U = (u_0, u_1, \dots, u_p)$ e $V = (v_0, v_1, \dots, v_q)$, tale che, per opportuni valori s_i e t_j dei parametri u e v , risulti minima la distanza tra i punti assegnati $Q_{i,j}$ e la superficie B-Spline determinata.

Per risolvere il problema, si può applicare la tecnica di approssimazione mediante curve B-Spline, suddividendo il problema nel modo seguente:

Passo 1. Considerata la matrice di dimensioni $m \times n$ costituita dai punti $Q_{i,j}$ da approssimare, $i = 0, \dots, m-1$, $j = 0, \dots, n-1$, si applica l'algoritmo di approssimazione mediante curve B-Spline alle colonne di punti $Q_{i,j}$ ottenute fissando l'indice j (ad esempio, per $j=0$, si ha la colonna $(Q_{00}, Q_{10}, \dots, Q_{m-10})$); facendo variare j tra 0 ed $n-1$, si effettuano in tutto n approssimazioni mediante curve B-Spline di grado h .

I risultati ottenuti formano colonne di punti intermedi $P_{i,j}$, con $i = 0, \dots, m-1$, $j = 0, \dots, n-1$, che costituiscono una matrice di dimensioni $m \times n$.

Passo 2. Si applica l'algoritmo di approssimazione mediante curve B-Spline alle righe di punti $P_{i,j}$ ottenute fissando l'indice i (ad esempio, per $i=0$, si ha la riga $(P_{00}, P_{01}, \dots, P_{0,n-1})$); facendo variare i tra 0 ed $m-1$, si effettuano in tutto m approssimazioni mediante curve B-Spline di

grado k .

I risultati ottenuti formano le righe dei punti di controllo cercati $p_{i,j}$, $i = 0, \dots, m-1$, $j = 0, \dots, n-1$, che costituiscono una matrice di dimensioni $m \times n$.

Essendoci ricondotti ad approssimazioni dei minimi quadrati mediate curve B-Spline, passiamo a formulare tale problema:

dati un insieme di n punti $Q_i = (a_i, b_i) \in R^2$, $i = 0, \dots, n-1$, ed assegnato un grado h , si cercano n punti di controllo $p_i = (x_i, y_i) \in R^2$, $i = 0, \dots, n-1$ tali che sia minima la distanza euclidea tra i punti assegnati e la curva B-Spline

$$C(u) = \sum_{i=0}^{n-1} p_i N_{i,h}(u)$$

definita dai punti di controllo calcolati e da un opportuno vettore dei nodi $U = (u_0, u_1, \dots, u_{n+h+1})$.

Si richiede, dunque, che sia minima la distanza

$$f = \sum_{k=0}^{n-1} |Q_k - C(t_k)|^2$$

dove i t_k sono i valori del parametro u associati ai corrispondenti punti Q_k .

Il minimo della funzione f si ha per quei valori di p_j per i quali risulta

$$\frac{\partial f}{\partial p_j} = 0.$$

Sviluppando il quadrato del binomio si ha

$$\begin{aligned} |Q_k - C(t_k)|^2 &= \left| Q_k - \left(\sum_{i=0}^{n-1} N_{i,h}(t_k) p_i \right) \right|^2 = \\ &= Q_k \cdot Q_k - 2 \left(\sum_{i=0}^{n-1} N_{i,h}(t_k) p_i \right) + \left(\sum_{i=0}^{n-1} N_{i,h}(t_k) p_i \right) \cdot \left(\sum_{i=0}^{n-1} N_{i,h}(t_k) p_i \right), \end{aligned}$$

da cui, derivando ogni termine rispetto a p_j e ponendo a zero la derivata, si ha:

$$\sum_{i=0}^{n-1} N_{j,h}(t_k) \sum_{i=0}^{n-1} N_{i,h}(t_h) p_i = \sum_{k=0}^{n-1} N_{j,h}(t_k) Q_k.$$

In forma matriciale, il sistema può essere riscritto come

$$N^T N P = N^T Q,$$

dove

$$P = \begin{bmatrix} p_0 \\ p_1 \\ \vdots \\ p_{n-1} \end{bmatrix}, \quad Q = \begin{bmatrix} Q_0 \\ Q_1 \\ \vdots \\ Q_{n-1} \end{bmatrix} \quad \text{e}$$

$$N = \begin{bmatrix} N_{0,h}(t_0) & N_{1,h}(t_0) & \dots & N_{n-1,h}(t_0) \\ N_{0,h}(t_1) & N_{1,h}(t_1) & \dots & N_{n-1,h}(t_1) \\ \vdots & \vdots & & \vdots \\ N_{0,h}(t_{n-1}) & N_{1,h}(t_{n-1}) & \dots & N_{n-1,h}(t_{n-1}) \end{bmatrix}_{(n \times n)}.$$

Indicando con

$$\begin{cases} X(u) = \sum_{i=0}^{n-1} x_i N_{i,h}(u) \\ Y(u) = \sum_{i=0}^{n-1} y_i N_{i,h}(u) \end{cases}$$

le equazioni parametriche della B-Spline $C(u)$, e ricordando che $Q_i = (a_i, b_i)$, $i = 0, \dots, n-1$, i sistemi da risolvere sono:

$$N^T N x = N^T a \quad \text{e} \quad N^T N y = N^T b,$$

dove

$$x = \begin{bmatrix} x_0 & x_1 & \dots & x_{n-1} \end{bmatrix}^T, \quad y = \begin{bmatrix} y_0 & y_1 & \dots & y_{n-1} \end{bmatrix}^T,$$

$$a = \begin{bmatrix} a_0 & a_1 & \dots & a_{n-1} \end{bmatrix}^T, \quad b = \begin{bmatrix} b_0 & b_1 & \dots & b_{n-1} \end{bmatrix}^T.$$

L'algoritmo per l'approssimazione di punti mediante curve B-Spline, risulta così abbastanza semplice. Per assicurare l'esistenza e l'unicità della soluzione del problema, si adopera una scelta opportuna dei valori t_i del parametro u da utilizzare e del vettore dei nodi U , in base ai seguenti teoremi:

Teorema 1:

f^ è una migliore approssimazione dei minimi quadrati di g nello spazio X se e solo se $f^* \in X$ e la funzione $f^* - g$ è ortogonale ad X , cioè*

$$\langle f^* - g, f \rangle = 0 \quad \forall f \in X.$$

In particolare, g ha un'unica migliore approssimazione dei minimi quadrati f^* in X , se e solo se $\|\cdot\|_2$ è una norma in X , cioè se e solo se

$$\|f\|_2 = 0 \Rightarrow f = 0.$$

Occorre quindi dimostrare che la norma utilizzata è effettivamente una norma nello spazio $S_h(U)$ delle funzioni spline polinomiali definite sul vettore dei nodi U . Sussiste il seguente

Teorema 2:

La seminorma

$$\|f\|_2 = \sqrt{\sum_{j=0}^{n-1} f(t_j)^2}$$

con $t_i \leq t_{i+1}$, $i = 0, \dots, n-2$, è una norma sullo spazio $S_h(U)$ se

e solo se per qualche sequenza di indici $0 \leq j_0 < \dots < j_{n-1} \leq n-1$ risulta

$$u_i < t_{j_i} < u_{i+h}, \quad i = 0, \dots, n-1.$$

Un metodo per costruire una parametrizzazione soddisfacente a tali requisiti, è quello detto *Chord Length*; tale metodo impone che si scelgano

$$\begin{aligned} t_0 &= 0, \\ t_{n-1} &= 1, \\ t_i &= t_{i-1} + \frac{|Q_i - Q_{i-1}|}{\sum_{j=1}^{n-1} |Q_j - Q_{j-1}|}, \quad i = 1, \dots, n-2. \end{aligned}$$

Il vettore dei nodi U può essere poi generato a partire dalla seguente formula:

$$\begin{aligned} u_0 &= u_1 = \dots = u_h = 0, \\ u_{j+h} &= \frac{1}{h} \sum_{i=j}^{j+h-1} t_i, \quad j = 1, \dots, n-h \\ u_{n+1} &= \dots = u_{n+h+1} = 1, \end{aligned}$$

Con la scelta dei parametri e dei nodi così fatta, si verifica che la matrice N costituita dalle funzioni base B-Spline è una matrice a banda con ampiezza di banda non maggiore del grado h della curva B-Spline approssimante. La matrice $N^T N$ risulta così simmetrica, definita positiva, e a banda, con ampiezza di banda non superiore a $2h-1$.

Per la costruzione della matrice

$$N = \begin{bmatrix} N_{0,h}(t_0) & N_{1,h}(t_0) & \dots & N_{n-1,h}(t_0) \\ N_{0,h}(t_1) & N_{1,h}(t_1) & \dots & N_{n-1,h}(t_1) \\ \vdots & \vdots & & \vdots \\ N_{0,h}(t_{n-1}) & N_{1,h}(t_{n-1}) & \dots & N_{n-1,h}(t_{n-1}) \end{bmatrix}_{(n \times n)}$$

bisogna determinare, per ognuno dei valori $u = t_i, i = 0, \dots, n-1$, l'indice l tale che $t_i \in [u_l, u_{l+1})$. Una volta determinato l , si osserva che, per la proprietà di supporto locale delle funzioni di base B-Spline, l'equazione

$$C(u) = \sum_{i=0}^{n-1} p_i N_{i,h}(u)$$

si riduce a

$$C(u) = \sum_{i=l-h}^l p_i N_{i,h}(u),$$

per cui bisogna calcolare solo le funzioni di base B-Spline

$$N_{l-h,h}(u), \dots, N_{l,h}(u).$$

L'algoritmo utilizzato per il calcolo delle basi, è mostrato di seguito:

```

for i=0,h-1
     $N_i = 0$ 
endfor
 $N_l = 1$ 
for j=1,h
    for k=l-j, l
        
$$N_k = \frac{u-u_k}{u_{k+j}-u_k} N_k + \frac{u_{k+1+j}-u}{u_{k+1+j}-u_{k+1}} N_{k+1}$$

    endfor
endfor.

```

La complessità di tempo dell'algoritmo così sviluppato è

$$T_{BASI} = O\left(\frac{3}{2}h^2\right)$$

Infine, l'algoritmo per l'approssimazione nel senso dei minimi quadrati mediante curve B-Spline può essere riassunto nei seguenti passi:

Passo 1. Si costruisce un'opportuna parametrizzazione $(t_0, t_1, \dots, t_{n-1})$ mediante il metodo Chord Length;

Passo 2. Si costruisce il vettore dei nodi $U = (u_0, u_1, \dots, u_{n+h})$;

Passo 3. Si costruisce la matrice dei coefficienti $N = (N_{j,h}(t_i))_{i,j=0,\dots,n-1}$;

Passo 4. Si calcola il prodotto $N^T N$;

la complessità di tempo dell'algoritmo è:

$$T_{MATMAT}(h, n) = O(2hn^2).$$

Passo 5. Si applica l'algoritmo di Cholesky alla matrice $N^T N$ ottenendo una matrice triangolare inferiore L tale che $N^T N = LL^T$;

la complessità di tempo dell'algoritmo è:

$$T_{CHOLBAND}(h, n) = O(h^2 n).$$

Passo 6. Si calcolano i prodotti $N^T a$ ed $N^T b$, costituenti i vettori dei termini noti del sistema, dove $a = (a_0, a_1, \dots, a_{n-1})^T$ e $b = (b_0, b_1, \dots, b_{n-1})^T$;

la complessità di tempo per questi prodotti è:

$$2 * T_{MATVET}(h, n) = 2 * O(hn).$$

Passo 7. Si risolvono i sistemi finali $LL^T x = N^T a$ e $LL^T y = N^T b$ mediante forward e back substitution.

la complessità di tempo è:

$$2 * [T_{FORWARDBAND}(2h-1, n) + T_{BACKBAND}(2h-1, n)] = 4 * O(4hn).$$

Tornando all'approssimazione dei minimi quadrati mediante superfici B-spline, bisogna specificare una parametrizzazione generalizzata ottenuta dalla parametrizzazione *Chord Length*. Difatti, assegnati $m \times n$ punti $Q_{i,j} = (a_{i,j}, b_{i,j}, c_{i,j}) \in R^3$, con $i = 0, \dots, m-1$ e $j = 0, \dots, n-1$, bisogna determinare una superficie B-Spline

$$S(u, v) = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} N_{i,h}(u) N_{j,k}(v) p_{i,j}$$

di gradi h e k con punti di controllo opportuni $p_{i,j} = (x_{i,j}, y_{i,j}, z_{i,j}) \in R^3$, per $i = 0, \dots, m-1$ e $j = 0, \dots, n-1$, ed opportuni vettori dei nodi $U = (u_0, u_1, \dots, u_p)$ e $V = (v_0, v_1, \dots, v_q)$, tale che, per opportuni valori

s_i e t_j dei parametri u e v , risulti minima la distanza tra i punti assegnati Q_{ij} e la superficie B-Spline determinata.

Bisogna, quindi, prima di tutto, definire i valori s_i e t_j dei parametri, per $i = 0, \dots, m - 1$ e $j = 0, \dots, n - 1$:

fissato l'indice di colonna j della matrice dei punti $Q_{i,j}$, si calcolano, mediante il metodo Chord Length, m parametri provvisori $s_{0,j}, s_{1,j}, \dots, s_{m-1,j}$. Facendo variare j tra 0 ed n , si ricavano $m \times n$ parametri provvisori; i parametri desiderati s_0, s_1, \dots, s_{m-1} sono semplicemente la media effettuata su ogni riga dei parametri provvisori, come è mostrato nell'immagine seguente:

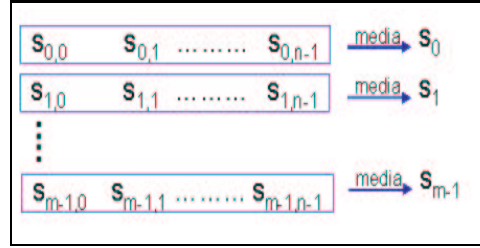


Figura 3.1: Calcolo dei parametri.

Il calcolo dei valori t_j , $j = 0, \dots, n - 1$ per il parametro v è simile:

fissato l'indice di riga i della matrice dei punti $Q_{i,j}$, si calcolano, mediante il metodo *Chord Length*, n parametri provvisori $t_{i,0}, t_{i,1}, \dots, t_{i,n-1}$. Facendo poi variare i tra 0 ed m , si ricavano $m \times n$ parametri provvisori; i parametri desiderati t_0, t_1, \dots, t_{n-1} sono semplicemente la media effettuata su ogni colonna dei parametri provvisori.

La procedura per il calcolo dei parametri s_i , $i = 0, \dots, m - 1$ è

riportata di seguito:

```

for j=0,n-1
    s0,j=0
    for i=1,m-2
         $s_{i,j} = s_{i-1,j} + \frac{|Q_{i,j} - Q_{i-1,j}|}{\sum_{k=1}^{m-1} |Q_{k,j} - Q_{k-1,j}|}$ 
    endfor
    sl-1,j = 1
endfor
s0 = 0;
for i=1,m-2
     $s_i = \frac{1}{n} \sum_{j=0}^{n-1} s_{i,j}$ 
endfor
sm-1 = 1.

```

Dopo aver costruito i vettori dei nodi U e V in modo analogo al caso delle curve B-Spline, si procede adoperando la tecnica di approssimazione mediante curve B-Spline. Si consideri la matrice $Q = (Q_{i,j})$ costituita dai punti $Q_{i,j} = (a_{i,j}, b_{i,j}, c_{i,j})$, $i = 0, \dots, m-1$, $j = 0, \dots, n-1$ da approssimare. Su ogni colonna j , per $j = 0, \dots, n-1$, della matrice Q si effettua l'approssimazione mediante curve B-Spline, costruendo la matrice N di dimensioni $m \times m$ contenente le funzioni base B-Spline di grado h , con i parametri s_i , $i = 0, \dots, m-1$, ed il vettore dei nodi U ; si risolvono, per ogni colonna j , i tre sistemi

$$N^T N \bar{x} = N^T a, \quad N^T N \bar{y} = N^T b \quad \text{e} \quad N^T N \bar{z} = N^T c,$$

dove

$$N = \begin{bmatrix} N_{0,h}(s_0) & N_{1,h}(s_0) & \dots & N_{n-1,h}(s_0) \\ N_{0,h}(s_1) & N_{1,h}(s_1) & \dots & N_{n-1,h}(s_1) \\ \vdots & \vdots & & \vdots \\ N_{0,h}(s_{n-1}) & N_{1,h}(s_{n-1}) & \dots & N_{n-1,h}(s_{n-1}) \end{bmatrix}_{(n \times n)},$$

$$\bar{x} = \begin{bmatrix} \bar{x}_{0,j} \\ \bar{x}_{1,j} \\ \vdots \\ \bar{x}_{n-1,j} \end{bmatrix}, \bar{y} = \begin{bmatrix} \bar{y}_{0,j} \\ \bar{y}_{1,j} \\ \vdots \\ \bar{y}_{n-1,j} \end{bmatrix}, \bar{z} = \begin{bmatrix} \bar{z}_{0,j} \\ \bar{z}_{1,j} \\ \vdots \\ \bar{z}_{n-1,j} \end{bmatrix},$$

$$a = \begin{bmatrix} a_{0,j} \\ a_{1,j} \\ \vdots \\ a_{n-1,j} \end{bmatrix}, b = \begin{bmatrix} b_{0,j} \\ b_{1,j} \\ \vdots \\ b_{n-1,j} \end{bmatrix}, c = \begin{bmatrix} c_{0,j} \\ c_{1,j} \\ \vdots \\ c_{n-1,j} \end{bmatrix}$$

ottenendo le coordinate di punti provvisori

$$\bar{p}_{i,j} = (\bar{x}_{i,j}, \bar{y}_{i,j}, \bar{z}_{i,j})_{i=0,\dots,m-1,j=0,\dots,n-1}.$$

Utilizzando poi i parametri t_i ed il vettore dei nodi V , si costruisce la matrice M di dimensioni $m \times m$ di funzioni di base B-Spline di grado k ; su ogni riga i , per $i = 0, \dots, m-1$, della matrice $P = (\bar{p}_{i,j})_{i=0,\dots,m-1,j=0,\dots,n-1}$ si effettua l'approssimazione mediante curve B-Spline, con i parametri t_j , $j = 0, \dots, n-1$, ed il vettore dei nodi V ; si risolvono, per ogni riga, i tre sistemi

$$M^T Mx = M^T \bar{x}, M^T My = M^T \bar{y} \text{ e } M^T Mz = M^T \bar{z},$$

dove

$$M = \begin{bmatrix} M_{0,k}(t_0) & M_{1,k}(t_0) & \dots & M_{m-1,k}(t_0) \\ M_{0,k}(t_1) & M_{1,k}(t_1) & \dots & M_{m-1,k}(t_1) \\ \vdots & \vdots & & \vdots \\ M_{0,k}(t_{m-1}) & M_{1,k}(t_{m-1}) & \dots & M_{m-1,k}(t_{m-1}) \end{bmatrix}_{(m \times m)},$$

$$\bar{x} = \begin{bmatrix} \bar{x}_{i,0} & \bar{x}_{i,1} & \dots & \bar{x}_{i,m-1} \end{bmatrix}^T, \bar{y} = \begin{bmatrix} \bar{y}_{i,0} & \bar{y}_{i,1} & \dots & \bar{y}_{i,m-1} \end{bmatrix}^T,$$

$$\bar{z} = \begin{bmatrix} \bar{z}_{i,0} & \bar{z}_{i,1} & \dots & \bar{z}_{i,m-1} \end{bmatrix}^T$$

$$x = \begin{bmatrix} x_{i,0} & x_{i,1} & \dots & x_{i,m-1} \end{bmatrix}^T, y = \begin{bmatrix} y_{i,0} & y_{i,1} & \dots & y_{i,m-1} \end{bmatrix}^T,$$

$$z = \begin{bmatrix} z_{i,0} & z_{i,1} & \dots & z_{i,m-1} \end{bmatrix}^T,$$

ottenendo le coordinate di punti di controllo desiderati

$$p_{i,j} = (x_{i,j}, y_{i,j}, z_{i,j})_{i=0,\dots,m-1,j=0,\dots,n-1}.$$

3.1.5. NURBS Approximation nel senso dei minimi quadrati

Il problema dell'approssimazione mediante superfici NURBS può essere formulato come segue:

assegnati $m \times n$ punti $Q_{i,j} = (a_{i,j}, b_{i,j}, c_{i,j}) \in R^3$, e $m \times n$ pesi $r_{i,j} \in R$, con $i = 0, \dots, m-1$ e $j = 0, \dots, n-1$, bisogna determinare una superficie NURBS

$$S(u, v) = \frac{1}{\sum_{i=0}^{n-1} N_{i,h}(u) w_i} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} N_{i,h}(u) N_{j,k}(v) w_{i,j} p_{i,j}$$

di gradi h e k con punti di controllo opportuni $p_{i,j} = (x_{i,j}, y_{i,j}, z_{i,j}) \in R^3$, pesi associati $w_{i,j} \in R$, per $i = 0, \dots, m-1$ e $j = 0, \dots, n-1$, ed opportuni vettori dei nodi $U = (u_0, u_1, \dots, u_p)$ e $V = (v_0, v_1, \dots, v_q)$, tale che, per opportuni valori s_i e t_j dei parametri u e v , risulti minima la distanza tra i punti assegnati $Q_{i,j}$ e la superficie NURBS $S(u, v)$ determinata.

E' possibile ricondurre il problema a quello dell'approssimazione dei minimi quadrati mediante superfici B-Spline, come riassunto di seguito:

Passo 1. Si calcolano le coordinate omogenee dei punti $Q_{i,j} = (a_{i,j}, b_{i,j}, c_{i,j}) \in R^3$ da approssimare, ottenendo:

$$Q_{i,j}^r = (a_{i,j} r_{i,j}, b_{i,j} r_{i,j}, c_{i,j} r_{i,j}, r_{i,j}) \quad i = 0, \dots, m-1, \quad j = 0, \dots, n-1.$$

Passo 2. Utilizzando l'algoritmo di approssimazione mediante superfici B-Spline, si calcolano i punti di controllo della superficie B-Spline la cui distanza dai punti da approssimare sia minima:

$$p_{i,j}^w = (x_{i,j}^w, y_{i,j}^w, z_{i,j}^w, w_{i,j}^w) \quad i = 0, \dots, m-1, \quad j = 0, \dots, n-1.$$

Passo 3. Si proiettano i punti di controllo nello spazio R^3 :

$$p_{i,j} = (x_{i,j}, y_{i,j}, z_{i,j}) = \left(\frac{w_{i,j}^w}{w_{i,j}^w}, \frac{y_{i,j}^w}{w_{i,j}^w}, \frac{z_{i,j}^w}{w_{i,j}^w} \right) \quad i = 0, \dots, m-1, \quad j = 0, \dots, n-1.$$

3.2. Dettagli implementativi dell'algoritmo di approssimazione

In questo paragrafo viene descritto in dettaglio l'algoritmo di approssimazione, implementato in linguaggio Java.

L'algoritmo prende in input, in lettura da un file, i seguenti parametri:

```
int m, int n, dove m*n è il numero
dei punti dello spazio da approssimare (sampling points)

double[ ][ ] coord_x_sample, matrice contenente
le coordinate x dei punti da approssimare

double[ ][ ] coord_y_sample, matrice contenente
le coordinate y dei punti da approssimare

double[ ][ ] coord_z_sample, matrice contenente
le coordinate z dei punti da approssimare

int h, grado delle funzioni di base B-Spline lungo la direzione u

int k, grado delle funzioni di base B-Spline lungo la direzione v

double[ ][ ] weight_sample, matrice dei pesi
associati ai sampling points.
```

I primi metodi nell'algoritmo servono a calcolare i parametri s_i , $i = 0, \dots, m-1$ e t_j , $j = 0, \dots, n-1$ e i vettori dei nodi $U = (u_0, u_1, \dots, u_{m+h})$ e $V = (v_0, v_1, \dots, v_{n+k})$, e si basano sulle procedure viste nella sezione precedente:

Calcolo dei parametri
<pre> for j=0,n-1 s_{0,j}=0 for i=1,m-2 $s_{i,j} = s_{i-1,j} + \frac{ Q_{i,j} - Q_{i-1,j} }{\sum_{k=1}^{m-1} Q_{k,j} - Q_{k-1,j} }$ endfor s_{l-1,j} = 1 endfor s₀ = 0; for i=1,m-2 $s_i = \frac{1}{n} \sum_{j=0}^{n-1} s_{i,j}$ endfor s_{m-1} = 1. </pre>

Calcolo dei nodi
$u_0 = u_1 = \dots = u_h = 0,$ $u_{j+h} = \frac{1}{h} \sum_{i=j}^{j+h-1} s_i, \quad j = 1, \dots, n-h$ $u_{n+1} = \dots = u_{n+h+1} = 1.$

```

Compute_parameter_u
(double[][] coord_x_sample, double[][] coord_y_sample,
double[][] coord_z_sample, double[][] weight_sample,
int m, int n),
metodo per calcolare i parametri lungo la direzione u,
usando le coordinate dei sampling points, le dimensioni
della griglia e i pesi;

Compute_parameter_v
(double[][] coord_x_sample, double[][] coord_y_sample,
double[][] coord_z_sample, double[][] weight_sample,
int m, int n),
metodo per calcolare i parametri lungo la direzione v,
usando le coordinate dei sampling points, le dimensioni
della griglia e i pesi;

Compute_knot(int dimension, int degree, double[] Parameter),
metodo per calcolare il vettore dei nodi
lungo una qualsiasi delle due direzioni, usando i parametri,
il grado
delle funzioni di base B-Spline e le dimensioni della griglia.

```

Seguono poi i metodi per il calcolo delle matrici $N(m \times m)$ ed $M(n \times n)$ delle funzioni di base B-Spline; per ogni valore del parametro u , il ciclo è il seguente:

Calcolo delle funzioni di base B-Spline

<pre> for j=1,h for k=1-j,1 $N_k = \frac{u-u_k}{u_{k+j}-u_k} N_k + \frac{u_{k+1}+j-u}{u_{k+1}+j-u_{k+1}} N_{k+1}$ endfor endfor. </pre>
--

I metodi, in dettaglio, sono:

```

row_basis_construction (int degree, int u, int dimension,
double[ ] knot),
metodo per valutare le funzioni di base B-Spline nel parametro  $u$ ,
a partire dal grado delle funzioni, dal parametro, dal vettore
dei nodi
e dalle dimensioni della griglia;

Basis_assembler (double[ ] parameter, int degree, int dimension,
double[ ] knot),
metodo per calcolare la matrice delle funzioni di base B-Spline,
a partire dal vettore dei parametri, dal grado
delle funzioni di base B-Spline, dalle dimensioni della griglia
e dal vettore dei nodi.

```

L'algoritmo effettua poi la fattorizzazione di Cholesky sulla matrice ottenuta calcolando il prodotto righe per colonne tra la trasposta della matrice delle funzioni di base B-Spline e la matrice stessa, al fine di ottenere due matrici, una triangolare superiore e l'altra triangolare inferiore, in modo da poter usare la Back substitution e la Forward Substitution per risolvere sistemi lineari:

```

cholesky (double[ ][ ] matrix, int dimension),
metodo per effettuare la decomposizione di Cholesky della matrice
matrix;

back_substitution(double[][ ] A, double[] b, int dimension),
metodo che calcola la soluzione del sistema  $Ax = b$ 
quando A è una matrice triangolare superiore;

forward_substitution(double[][ ] A, double[] b, int dimension),
metodo che calcola la soluzione del sistema  $Ax = b$ 
quando A è una matrice triangolare inferiore.

```

3.3 Risultati dell'algoritmo

3.3.1 Approssimazione di superfici del secondo ordine

L'algoritmo di approssimazione di superfici mediante superfici NURBS è stato implementato su un nodo del cluster Beowulf installato presso il dipartimento di Matematica e Applicazioni dell'Università di Napoli "Federico II".

Le superfici utilizzate per lo sviluppo del Learning Object sono essenzialmente superfici quadriche, approssimate mediante superfici NURBS; di seguito vengono quindi riportati i tempi impiegati ed i risultati dell'algoritmo.

Superficie NURBS costituita da 41×41 punti di controllo, di grado 3×3 ;

tempo di implementazione: 0.41 secondi:

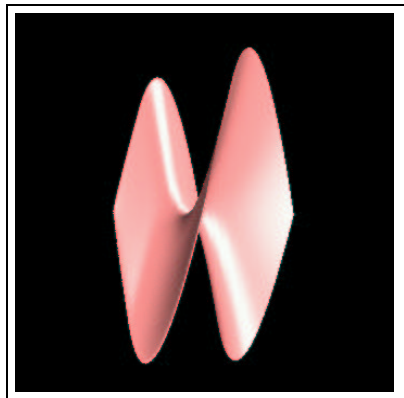


Figura 3.3: Superficie NURBS approssimante un paraboloide iperbolico.

Superficie NURBS costituita da 5×5 punti di controllo, di grado 3×3 ;

tempo di implementazione: 0.00075 secondi:

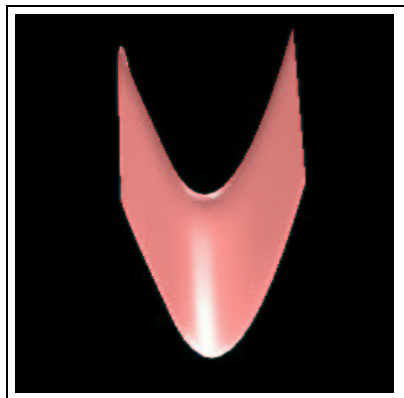


Figura 3.4: Superficie NURBS approssimante un paraboloide iperbolico.

Superficie NURBS costituita da 9×9 punti di controllo, di grado 3×3 ;

Tempo di implementazione: 0.41 secondi:

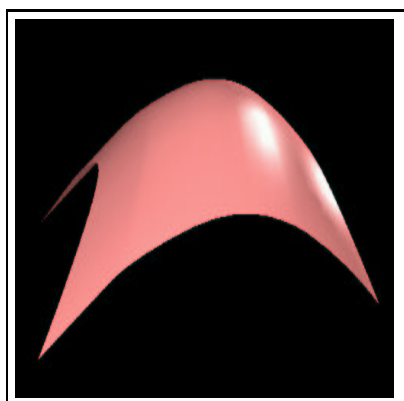


Figura 3.5: Superficie NURBS approssimante un iperboloide ellittico.

Superficie NURBS costituita da 30×30 punti di controllo, di grado 3×3 ;

Tempo di implementazione: 0.16 secondi:



Figura 3.6: *Superficie NURBS approssimante una sfera.*

Superficie NURBS costituita da 6×6 punti di controllo, di grado 3×3 ;

Tempo di implementazione: 0.0012 secondi:



Figura 3.7: *Superficie NURBS approssimante una sfera.*

3.3.2 RDF di una superficie quadrica

In questa sezione descriviamo l'implementazione del 3D-resource broker, descritto nella sezione precedente, nel framework del Semantic Web. Il modello hourglass suggerisce di adottare un bundle di pro-

toccolli limitati che nel caso del nostro broker si riduce al solo XML. Consideriamo, pertanto, una gerarchia di livelli del Web semantico che si estende sino al livello OWL nella parte superiore della clessidra, e fino al livello RDF nella sua parte inferiore.

Il tipo di negoziazioni che possono essere trattate dal singolo broker dipende essenzialmente dal suo profilo, specificato dal tipo di input ed output accettati, che si riflette sulle richieste effettuate dal livello applicativo e sulla tipologia delle risorse negoziate. Nel nostro caso l'output è una superficie NURBS e di conseguenza il livello RDFs della parte superiore deve contenere le informazioni sufficienti a definirla univocamente.

Nella parte inferiore del layer cake, che corrisponde all'input dell'algoritmo che implementa il broker, la definizione dell'RDF dipende dal tipo di risorsa da negoziare e quindi dalla query formulata nel livello delle applicazioni.

Il caso che consideriamo è quello in cui al livello applicativo si richiede il calcolo di una superficie a partire da un insieme di punti di campionamento:

RDF1, input:

```
<input: schema xmlns:xsd="http://www.dma.unina.it/Xmls">
<input: element name = "sampling_set" type = "input:string"/>
<output:attribute name = "degree_u" type = integer/>
<output: attribute name = "degree_v" type = integer/>
<output :sequence name ="punti" type = "in-put:ListOfPoints">
<output:element name = "elementi" type = "output:string">
<output:element/>
<output: sequence/>
</input:schema>
```

RDF2, output:

```
<output:schema xmlns:xsd="http://www.dma.unina.it/Xmls">
<output: element name = "nurbs" type ="input:string" />
<output: element name = "degree" >
<output:attribute name = "degree_u" type = integer/>
<output: attribute name = "degree_v" type = integer/>
<output:element/>

<ouptut:sequence name = "knot_spanning_u" type ="input:list">
<output: attribute name = "length_u" type = integer/>
<ouput:element name ="knot_spanning_item_u"
minOccurs = 1 type =real/>
<output:sequence/>

<ouptut:sequence name = "knot_spanning_v" type ="input:ListOfReals">
<output: attribute name = "length_v" type = integer/>
<ouput:element name ="knot_spanning_item_v"
minOccurs = 1 type =real/>
<output:sequence/>

<output :sequence name ="control_point_matrix"
type = "input:ListOfPoints">
<output:element name = "knot" type = "output:string">
<output:attribute name = "weight" type =real/>
<output:element/>
<output: sequence/>

</output:schema>
```

Input dell'algoritmo:

Sampling set	$(-2 \ -1 \ 0) \ (1 \ 2 \ -2) \ (-1 \ 0 \ 1) \ (2 \ -2 \ -1) \ (0 \ 1 \ 2),$ $(-2 \ -1 \ 0) \ (1 \ 2 \ -2) \ (-1 \ 0 \ 1) \ (2 \ -2 \ -2) \ (-2 \ -2 \ -2),$ $(-1 \ -1 \ -1) \ (-1 \ -1 \ 0) \ (0 \ 0 \ 0) \ (0 \ 1 \ 1) \ (1 \ 1 \ 1),$ $(2 \ 2 \ 2) \ (2 \ 2 \ 8) \ (5 \ 4 \ 5) \ (8 \ 5 \ 2) \ (1 \ 2 \ 5),$ $(4 \ 1 \ 0) \ (1 \ 4 \ 5) \ (2 \ 1 \ 2) \ (5 \ 8 \ 5) \ (4 \ 5 \ 8)$
Grado nella direzione u	3
Grado nelle direzione v	3
Pesi	Tutti uguali ad 1

File XML del Sampling Set (RDF):

```
<?xml version= "1.0" >
<sampling_set>
<degree_u> 3</degree_u>
<degree_v> 3</degree_v>
<punti>
<elemento>(-2 -1 0)</elemento>
<elemento> (1 2 -2)</elemento>
<elemento>(-1 0 1) </elemento>
<elemento>(-1 0 1) </elemento>
<elemento>(2 -2 -1) </elemento>
<elemento>(0 1 2) </elemento>
<elemento> (-2 -1 0) </elemento>
<elemento> (1 2 -2)</elemento>
<elemento> (-1 0 1)</elemento>
<elemento> (2 -2 -2)</elemento>
<elemento> (-2 -2 -2) </elemento>
<elemento>(-1 -1 -1) </elemento>
(<elemento>(1 1 1) </elemento>
<elemento>(-1 -1 0) </elemento>
<elemento>(0 0 0) </elemento>
<elemento> (0 1 1)</elemento>
<elemento>(2 2 2) </elemento>
<elemento> (2 2 8) </elemento>
<elemento> (5 4 5)</elemento>
<elemento>(8 5 2) </elemento>
<elemento> (1 2 5)</elemento>
<elemento> (4 1 0) </elemento>
<elemento> (1 4 5) </elemento>
<elemento> (2 1 2)</elemento>
<elemento> (5 8 5) </elemento>
<elemento>(4 5 8) </elemento>

</punti>
</set>
```

Output dell'algoritmo:

Vettore dei nodi nella direzione u	(0, 0, 0, 0, 0.5, 1, 1, 1, 1)
Vettore dei nodi nella direzione v	(0, 0, 0, 0, 0.5, 1, 1, 1, 1)
Coordinate dei punti di controllo	(-2.00 -2.00 8.00) (-2.24 -2.00 8.45) (-0.00 -1.99 -0.45) (2.24 -2.00 8.45) (1.99 -1.99 7.99) (-2.00 -2.24 8.45) (-2.24 -2.24 8.91) (-0.00 -2.24 -0.00) (2.24 -2.24 8.91) (1.998331 -2.24 8.49) (-1.99 -0.00 -0.45) (-2.24 -0.00 -0.00), (-0.00 -0.00 -8.91), (2.24 -0.00 -0.00) (1.99 -0.00 -0.45) (-2.00 2.24 8.47) (-2.24 2.24 8.91) (-0.00 2.24 -0.00) (2.24 2.24 8.91) (1.99 2.24 8.44) (-1.99 1.99 7.99) (-2.24 1.99 8.44) (-0.00 1.99 -0.45) (2.242615 1.99 8.44) (1.99 1.99 7.98)
Pesi	Tutti uguali ad 1

File XML della nurbs (RDF):

```

<?xml version = "1.0"?>
<nurbs>
<degree> 3</degree>
<degree_u> 3</degree_u>
<degree_v>3 </degree_v>
<knot_spanning_u length_u= "9">
<knot_spanning_item_u>0</knot_spanning_item_u>
<knot_spanning_item_u>0</knot_spanning_item_u>
<knot_spanning_item_u>0</knot_spanning_item_u>
<knot_spanning_item_u>0.5</knot_spanning_item_u>
<knot_spanning_item_u>1</knot_spanning_item_u>
<knot_spanning_item_u>1</knot_spanning_item_u>
<knot_spanning_item_u>1</knot_spanning_item_u>
<knot_spanning_item_u>1</knot_spanning_item_u>
<knot_spanning_item_u>1</knot_spanning_item_u>
</knot_spanning_u>
<knot_spanning_v>
<knot_spanning_item_v>0</knot_spanning_item_v>
<knot_spanning_item_v>0</knot_spanning_item_v>
<knot_spanning_item_v>0</knot_spanning_item_v>
<knot_spanning_item_v>0</knot_spanning_item_v>
<knot_spanning_item_v>0.5</knot_spanning_item_v>
<knot_spanning_item_v>1</knot_spanning_item_v>
<knot_spanning_item_v>1</knot_spanning_item_v>
<knot_spanning_item_v>1</knot_spanning_item_v>
<knot_spanning_item_v>1</knot_spanning_item_v>
</knot_spanning_v>
<control_point_matrix>
<knot weight = "1"> (-2.00 -2.00 8.00)</knot>
<knot weight = "1"> (-2.24 -2.00 8.45)</knot>
<knot weight = "1"> (-0.00 -1.99 -0.45)</knot>
<knot weight = "1"> (2.24 -2.00 8.45)</knot>
<knot weight = "1"> (1.99 -1.99 7.99)</knot>
<knot weight = "1"> (-2.00 -2.24 8.45)</knot>
<knot weight = "1"> (-2.24 -2.24 8.91)</knot>
<knot weight = "1"> (-0.00 -2.24 -0.00)</knot>
<knot weight = "1"> (2.24 -2.24 8.91)</knot>
<knot weight = "1"> (1.998331 -2.24 8.49)</knot>
<knot weight = "1"> (-1.99 -0.00 -0.45)</knot>
<knot weight = "1"> (-2.24 -0.00 -0.00)</knot>
<knot weight = "1"> (-0.00 -0.00 -8.91)</knot>
<knot weight = "1"> (2.24 -0.00 -0.00)</knot>
<knot weight = "1"> (1.99 -0.00 -0.45)</knot>
<knot weight = "1"> (-2.00 2.24 8.47)</knot>
<knot weight = "1"> (-2.24 2.24 8.91)</knot>
<knot weight = "1"> (-0.00 2.24 -0.00)</knot>
<knot weight = "1"> (2.24 2.24 8.91)</knot>
<knot weight = "1"> (1.99 2.24 8.44)</knot>
<knot weight = "1"> (-1.99 1.99 7.99)</knot>
<knot weight = "1"> (-2.24 1.99 8.44)</knot>
<knot weight = "1"> (-0.00 1.99 -0.45)</knot>
<knot weight = "1"> (2.242615 1.99 8.44)</knot>
<knot weight = "1"> (1.99 1.99 7.98)</knot>
</control_point_matrix>
</nurbs>

```

3.4. Una shell semantica per il 3D-resource broker

Attualmente, le collezioni di modelli matematici nel mondo sono poco accessibili. Ci proponiamo di definire un modello di Learning Management System che usa la shell del web semantico per riusare, in maniera distribuita, le rappresentazioni 3D dei modelli matematici nelle repository Web.

Nel nostro modello, che ricalca in qualche modo la struttura a clessidra del GRID, individuiamo tre tipologie di attori: l'applicazione utente, la risorsa e il resource broker.

Con la prima categoria identifichiamo tutte le applicazioni che richiedono risorse, come ad esempio i filtri collaborativi. Le risorse sono costituite essenzialmente da asset e learning object presenti sul Web. Compito del resource broker è fornire risorse alle applicazioni che le richiedono.

In questo capitolo proponiamo un resource broker che usa le NURBS per mediare in qualche modo tra le query del livello applicativo e le risorse disponibili sul web.

Il matching tra query e risorsa è modellato con il meccanismo dei 3D shape descriptor, che noi trattiamo come classi RDF. Infine, proponiamo un modo di implementare questo modello in un'architettura compatibile con lo standard SCORM.

Descriviamo come gli algoritmi di approssimazione mediante NURBS

si contestualizzano nell'ambito generale della ricerca e del reuso delle risorse sul Web, e come alcune tecniche elaborate nel campo della 3D Pattern Recognition si inquadrano in una teoria matematica più generale.

La ricerca di shell semantiche 3D per condividere le risorse in rete è un campo molto studiato [61, 67]. Lo sviluppo esponenziale del Web ha reso necessari, in tutta la Computer Science, lo sviluppo di sistemi di pattern mining basati su una gerarchia di livelli di rappresentazione. Non è difficile generalizzare quanto osservato in [61] nel caso della grafica 3D: "La proliferazione delle risorse sulla rete sta spostando l'attenzione dalla produzione al reuso".

Le repository di oggetti 3D, dai video-game ai siti a carattere didattico, stanno diventando sempre più numerose. Gli esempi che forniamo intendono dare l'idea di quanto sia difficile reperire informazione che, in mancanza di strumenti per il filtraggio delle risorse, si fa sempre più ridondante e caotica e sempre meno reperibile.

Storicamente, si è partiti con shell semantiche di tipo testuale, in cui il matching avveniva tra keyword inserite dall'utente e un array di file indicizzati, ciascuno contenente informazioni significative di un oggetto 3D. Alcuni sistemi, come WordNet, erano dotati di una tabella di sinonimi che consentivano di cercare anche negli alias. Naturalmente, l'attributo più importante di un oggetto 3D è la sua forma. Uno dei metodi più naturali da adottare è lo *shape matching*, un algoritmo che stabilisce la corrispondenza tra due forme. Un'alternativa è la proiezione ortogonale di un oggetto su tre piani e un linguaggio di query che permette all'utente di abbozzare l'oggetto cercato su ciascuno dei

tre piani di proiezione. I risultati non sono stati molto soddisfacenti e si è proceduto alla definizione di un'interfaccia di Query strutturata.

Il meccanismo delle query strutturate si basa sul fatto che l'estrazione delle informazioni di un oggetto -Pattern Mining -è un processo molto costoso da farsi a run-time. Pertanto, si preferisce trasformare l'oggetto in uno shape descriptor che viene confrontato con il corrispondente in linguaggio di query. Intuitivamente, uno shape descriptor è una rappresentazione astratta di proprietà rilevanti di un oggetto. L'idea di shape descriptor -figura 3.17 -non è nuova né in matematica -dove può essere fatta risalire a Galois- né nella visione artificiale, ed è basata sulla relazione tra similarità e distanza.

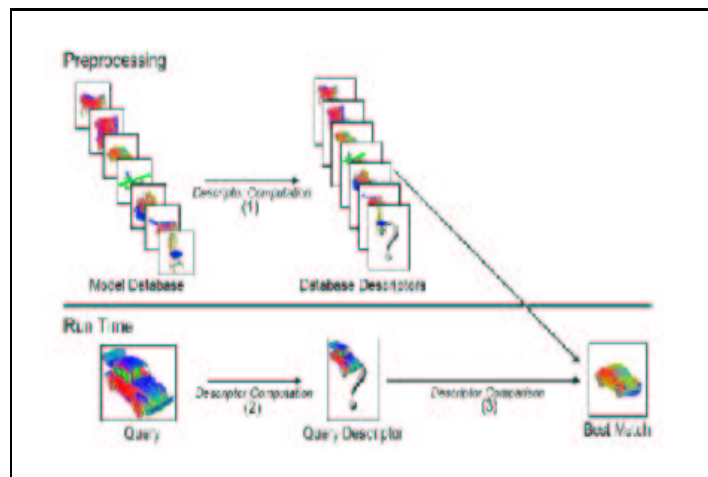


Figura 3.8: 3D Shape descriptor.

Definizione:

Uno *shape descriptor* è un'applicazione Φ di uno spazio metrico (S, d) in uno spazio di Banach (S', d') a dimensione finita.

Naturalmente, sarebbe opportuno che uno shape descriptor fosse un'isometria, in modo da aversi $d(s, s') = d'(\Phi(s), \Phi(s'))$. In questo modo il problema del pattern matching viene ricondotto allo studio delle distanze tra shape descriptors. La relazione tra distanza e similarità è stata molto studiata. Dato un insieme S , una similarità in S è un' applicazione $R : S \times S \rightarrow [0, 1]$ tale che, per ogni $x, y, z \in S$

$$\text{i) } R(x, x) = 1$$

$$\text{ii) } R(x, y) = R(y, x)$$

$$\text{iii) } R(x, z) \geq R(x, y) \wedge R(y, z)$$

dove \geq è l'operazione di minimo usuale sui numeri reali.

Più in generale, se il minimo è generalizzato in una norma triangolare, una \otimes -similarità in S è un' applicazione $R : S \times S \rightarrow [0, 1]$ tale che valgono i), ii), e iii) $R(x, z) \geq R(x, y) \otimes R(y, z)$.

Similmente, un \otimes -sottogruppo del gruppo delle trasformazioni biunivoche di un insieme S in se stesso, denotato con Σ_S , è un' applicazione $G_S : \Sigma_S \rightarrow [0, 1]$ tale che

$$G_S(id_S) = 1$$

$$G_S(f) \geq G_S(f^{-1})$$

$$G_S(f \bullet g) \geq G_S(f) \otimes G_S(g)$$

per ogni $f, g \in \Sigma_S$.

Si è indicato con id_S l'applicazione identica di S .

Kazhdan [61] ha definito la similarità tra due oggetti S ed S' come “il lavoro compiuto per trasformare un oggetto S in un oggetto S' ”. Questa definizione, intuitivamente evidente, resta probabilmente nel vago se non si considera il seguente risultato dimostrato in [48]:

Teorema 1:

Dato un insieme S , sia Σ_S il gruppo delle trasformazioni biunivoche di S in se. Allora, dato un \otimes -sottogruppo G di Σ_S ed una funzione quasi-invertibile $f : [0, 1] \rightarrow [0, +\infty)$, esistono una distanza d ed una \otimes -similarità R tali che

$$d = f(R(x, y))$$

$$R(x, y) = \sup_{g \in G(g)} \{G(g), g(x) = y\} \quad (*)$$

La (*) afferma in maniera formale che la similarità tra due figure è il grado della loro reciproca deformabilità.

Kazahan ha definito una relazione di similarità che si può estendere ad un insieme di punti dello spazio. Chiamiamo *voxel* un'unità elementare di volume. I

Dato un insieme S di punti di R^3 , si definiscono due matrici

1) La Matrice $Raster(R_S)$ è una matrice di ordine 3 ottenuta ponendo

$$a_{ijk} = \begin{cases} 1 & \text{se il voxel di coordinate } (i, j, k) \text{ interseca la frontiera di } S \\ 0 & \text{altrimenti} \end{cases}$$

2) La trasformata della distanza euclidea rispetto ad S (EDT_S)
 $EDT_S : R \rightarrow [0, +\infty)$ è definita ponendo

$$EDT_S(X) = \min_{Y \in \Delta(S)} d^2(X, Y)$$

Possiamo dare a questo punto la seguente definizione

Definizione:

Dati due sottoinsiemi S ed S' di R^3 , poniamo

$$R(S, S') = Raster(R_S) \bullet EDT_S(S') + Raster(R_{S'}) \bullet EDT_{S'}(S)$$

Questa definizione consente di definire un grado di similarità tra il sampling set dell'algoritmo di una Nurbs e una figura 3D. In questo modo si definisce una distanza tra una figura e la NURBS approssimante.

A questo punto abbiamo gli strumenti per formulare un modello distribuito che -con la mediazione di un resource broker basato su NURBS- rende accessibili i modelli matematici alle applicazioni utente che li richiedono.

Definito un linguaggio di query basato sull'RDF di un sampling set, il modello opera in questo modo:

- 1) si considera la RDF di un learning object LO che si vuole costruire, riutilizzando come asset dei modelli matematici allocati sul Web;
- 2) il learning object LO è composto da asset o learning object 3D di cui è specificata la RDF;
- 3) dalla RDF degli asset, si ricava un insieme di punti che viene convertito in uno shape descriptor;

4) con l'algoritmo di approssimazione NURBS, viene individuato uno shape descriptor per l'insieme dei punti;

5) tramite il resource broker, questo Shape descriptor è matchato con il corrispettivo dei modelli 3D allocati in una repository di modelli matematici, applicando una formula di tipo (*);

6) se il matching ha successo -la distanza tra le due superfici è all'interno di una soglia accettabile- allora il broker attiva un canale di comunicazione di tipo "late binding" tra la query e la risorsa nella repository 3D.

In questo modo il "late binding" determina la costruzione dinamica del learning object LO.

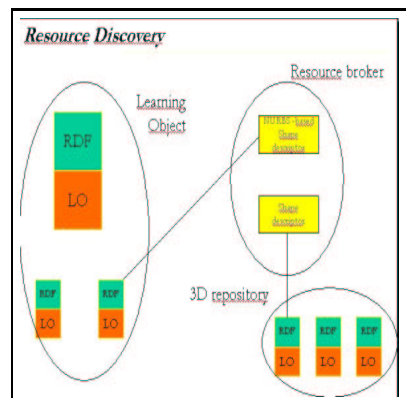


Figura 3.9: Resource brokering in una collezione distribuita di modelli matematici.

Riferimenti bibliografici

- [1] *3D Cafe*:
<http://www.3dcafe.com>.
- [2] *3D Model Retrieval System*:
<http://3d.csie.ntu.edu.tw/~dynamic/>.
- [3] *Content-based Classification of 3D-models by Capturing Spatial Characteristics*:
<http://merkur01.inf.uni-konstanz.de/>.
- [4] *De Espona 3D Models Enciclopedia*:
<http://www.deespona.com>.
- [5] *Extensible Markup Language (XML)*:
<http://www.w3.org/XML>.
- [6] *Introduction to Computing with Geometry Notes*:
<http://www.cs.mtu.edu/~shene/COURSES/cs3621/NOTES/notes.html>.
- [7] *Le Raccolte Museali Italiane di Modelli per lo Studio delle Matematiche Superiori*:
<http://www.dma.unina.it/~nicla.palladino/catalogo/>.
- [8] *Mathesis, Società italiana di scienze matematiche e fisiche*:
<http://www.mathesisnazionale.it/PagineHTML/index.htm>
- [9] *Mr. Furniture*:
<http://www.mr-cad.com>.
- [10] *Namespace in XML - World Wide Web Consortium*:
<http://www.w3.org/TR/REC-xml-names/>.

- [11] *Princeton 3D Model Search Engine*:
<http://shape.cs.princeton.edu>.
- [12] *Protein Data Bank*:
<http://www.rcsb.org>.
- [13] *ShapeSifter*:
<http://www.shapesearch.net/>.
- [14] *The Dublin Core Home Page*:
<http://dublincore.org/>.
- [15] *Viewpoint Data Labs*:
<http://www.viewpoint.com>, 2001.
- [16] *World Wide Web Consortium Home Page*:
<http://www.w3.org>.
- [17] Ankerst M. -Kastenmuller G. -Kriegel H. -Seidl T., *3d shape histograms for similarity search and classification in spatial databases*, in R. H. Güting, D. Papadias, and F. H. Lochovsky editors, *Advances in Spatial Databases, 6th International Symposium, SSD99, Hong Kong, China, proceedings, volume 1651 of Lecture Notes in Computer Science*, Springer, 1999.
- [18] Arcelli Fontana F. -Formato F. -Gerla G., *Similarity based unification, Logic Programming and Soft Computing*, J. Baldwin & Trevor Martin Eds.
- [19] Arcelli Fontana F. -Formato F., *Towards a Flexible Information Retrieval Approach Based on the Context*, *Proceedings 7th Workshop on Multimedia Information Systems*.

- [20] Arnheim R., *Il pensiero visivo*, Trad. It. Einaudi, Milano, 1969
- [21] Asperti -Padovani L. - Coen S. -Schena I., *Towards a Library of Formal Mathematics*, Proceedings of TPHOLS 2000.
- [22] Barendregt H.P., *The lambda Calculus*, Chapter 9 of the Handbook of Theoretical Computer Science - Vol. A, Model Theory and Semantics Edited by J. van Leeuwen, Albert R. Meyer, M. Nivat, Matthew Paterson and D. Perrin, 1992.
- [23] Berners Lee T., *A roadmap to the Semantic Web*, Sept 1998. P. Hayes, RDF Semantics, published on-line at <http://www.w3.org/TR/rdf-mt/>, 2003.
- [24] Berners Lee T. -Hendler J. -Lassila O., *The Semantic Web*, Scientific American 284 (May 2001).
- [25] Berners Lee T., with M. Fischetti, *Weaving the Web*, San Francisco, Harper, 1999.
- [26] Berners Lee T., *What the Semantic Web Can Represent*, September 1998, at <http://www.w3.org/DesignIssues/RDFnot.html>.
- [27] Berners Lee T., *Metadata architecture*, (1997): <http://www.w3.org/DesignIssues/Metadata.html>.
- [28] Besl P., *Triangles as a primary representation*, Object Recognition in Computer Vision, LNCS 994:191-206, 1995.
- [29] Beth E.W. -Piaget J., *Epistémologie mathématique et Psychologie*, PUF, Paris, 1961.
- [30] Booch G., *Object Oriented Analysis*, Addison Welsey, 1985.

- [31] Brachmann, *Whats in a concept: structural foundations for semantic networks*, International Journal of Man-machine Studies, 1977.
- [32] Campedelli L., *Il centenario del programma di Klein*, Archimede 1972.
- [33] Campedelli L., *Esercitazioni complementari di geometria*, Cedam, Padova.
- [34] Campedelli L., *Fantasia e logica nella matematica*, Sienza Nuova, Feltrinelli, 1973.
- [35] Casadei G. -Magnani M. -Sirocchi F., *Design and Implementation of an Adaptive Learning Management System*, SSRR 2002 Int.l Conf. on Advances in Infrastructure for e-Business, e-Education, e-Science, and e-Medicine on the Internet, L'Aquila, July 29- August 4 2002.
- [36] Casati R. -Varzi A., *Parts and Places: The Structures of Spatial Representation*, MIT Press, 1999.
- [37] Castelnuovo E., *Documenti di un'esposizione di matematica*, 1972 editore Boringhieri.
- [38] Castelnuovo E. -Barra M., *La matematica nella realtà*, Boringhieri, 1975.
- [39] Cavallo D. and Papert S., *The Future of Learning: Mission Statement and Project Description*.
- [40] Damiano E., *I mediatori didattici*, Irsae , 1994.

- [41] Davies J. -Fensel D. -Van Harmelen F., *The Semantic Web: Ontology-Driven Knowledge Management*, New York, Wiley, 2002.
- [42] De Luca A. -Ricciardi L.M., *Introduzione alla Cibernetica*, Franco Angeli, Milano.
- [43] Delingette H. -Hebert M. -Ikeuchi K., *Shape representation and image segmentation using deformable surfaces*. Image and Vision Computing, 1992.
- [44] Dumbill E., *The Semantic Web: A Primer*,
<http://www.xml.com/pub/a/2000/11/01/semanticweb>, November 2000.
- [45] Ferrarese G. -Palladino F., *Sulle collezioni di modelli matematici dei dipartimenti di matematica dell'Università e del Politecnico di Torino*, Leo S.Olschki Editore.
- [46] Findler, *Associative Networks: Representation and Use of Knowledge by Computers*, Academic Press , New York 1979.
- [47] Fisher G., *Mathematische Modelle*, Vieweg.
- [48] Formato F. -Gerla G. -Scarpati L., *Similarity and Fuzzy Subgroups*, Soft Computing, 2000.
- [49] Foster I. and Kesselman C., *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann, 1999.
- [50] Franchetta -Mastrogiacomo, *Geometria analitica con elementi di proiettiva*, Liguori Editore.
- [51] Frixione M., *Logica, Significato e Intelligenza Artificiale*, Franco Angeli, Milano, 1994.

- [52] Funkhouser T. -Min P. -Kazhdan M. -Chen J. -Halderman A. -Dobkin D. -Jacobs D., *A search engine for 3d models*, ACM Transactions on Graphics, pages 83-105, 2003.
- [53] Gardenfors P., *Conceptual Spaces*, MIT Press, Cambridge, MA, 2000.
- [54] Goldberg A., *SmallTalk 80*, Mc Graw Hill, 1981.
- [55] Gray W.W., *The Living Brain*, W.W. Norton, New York, 1963.
- [56] Guarino N., *Formal Ontology, Conceptual Analysis and Knowledge Representation*, International Journal of Human and Computer Studies, 1995.
- [57] Hilbert D. -Vossen S.C., *Geometria intuitiva*, Paolo Boringhieri.
- [58] Hodgins W., *The premature arrival of the future in Proceedings of Techlearn 2002*, Orlando, 2002.
- [59] Horn B., *Extended Gaussian images*, in Proceedings of the IEEE, volume 72, 1984.
- [60] Iannucci M. -Imperiali M., *Semantic Web*, reperibile a:
<http://online.infomedia.it/riviste/dev/90/articolo23/articolo.htm>.
- [61] Kazhdan M., *Shape Representation and Algorithms for 3d model retrieval*, Phd Thesis, University of Princeton, 2004.
- [62] Kang S. -Ikeuchi K., *Determining 3d object pose using the complex extended Gaussian image*, CVPR, pages 580-585, June 1991.
- [63] Kolovski V. -Galletly J., *E-learning via the Semantic Web*, International Conference on Computer Systems and Technologies, CompSysTech2003.

- [64] Lèvy P., *Qu'est-ce que le virtuel?*, Paris, La Découverte, 1998.
- [65] Loncaric S., *A survey of shape analysis techniques*, Pattern Recognition, 1998.
- [66] Maragliano, *Manuale di didattica multimediale*, Laterza Bari 2000.
- [67] Marchiori M., *The Mathematical Semantic Web*, proceedings of the International Conference on Mathematical Knowledge Management, LNCS vol.2594, Elsevier, 2003.
- [68] Minichiello G., *Il doppio pensiero Morano*, Napoli 1981.
- [69] Minichiello G., *L'apprendimento come enazione*, da "Il Mondo Intrepretato", editrice La Scuola, 1995.
- [70] Minsky M. -Papert S., *LOGO the Language*, MIT Press , 1975.
- [71] Negroponte N., *Being multimedial*, MIT Press, trad. it. Essere Multimediali, Laterza, Bari.
- [72] Osada R. -Funkhouser T. -Chazelle B. -Dobkin D., *Matching 3d models with shape distributions*, in "Shape Modeling International", IEEE Computer Society, 2001.
- [73] Palladino F., *Il Fondo di modelli e strumenti matematici antichi dell'Università di Padova e l'iniziativa di Giuseppe Veronese per un laboratorio nazionale italiano*, Università degli studi di Padova, 1999.
- [74] Palladino F., Memorie dell'Istituto Lombardo, Accademia di Scienze e lettere Vol.XXIX-Memoria 5 *Uno Specimen dei gia-*

cimenti italiani di modelli e strumenti matematici: il nachlass dell'Università di Pavia, Istituto Lombardo di Scienze e Lettere.

- [75] Petrucco C., *Le prospettive didattiche del Semantic web*.
- [76] Poincaré H., *Analysis Situs*, 1895, in Journal de l'Ecole Polytechnique, Boringhieri.
- [77] Rendiconto dell'Accademia delle scienze fisiche e matematiche, Vol. LXV, (1997). *Estratto: "Le Collezioni di strumenti e modelli matematici del dipartimento di matematica e applicazioni "R.Caccioppoli" dell'Università degli Studi di Napoli "Federico II": Cataloghi ragionati*, Liguori Editore.
- [78] Resnick Turtle M., *Termites and Traffic Jams*, MIT Press, Cambridge, 1997.
- [79] Schilling M., *Catalog mathematischer Modelle*, Leipzig, Verlag von M. Schilling, 1911.
- [80] Signore O., *RDF per la rappresentazione della conoscenza*.
- [81] Silov G.E., *Matematicheskij Analiz MIR*, 1978, Mosca 1978.
- [82] Stojanovic L. -Staab S. -Studer R., *E-learning based on the Semantic Web*.
- [83] Tolman E.C., *A new formula for behaviourism*, Psychological Review.
- [84] Van der Veen R.G.W., *Community Development as Citizen Education*, International Journal of Lifelong Education Vol 22, No. 6.

- [85] Van der Veen R.G.W, *The many layers of learning*, International Journal of Lifelong Education.
- [86] Vanni A., *La pedagogia nell'era della tecnica*, Edisud, Salerno.
- [87] Von Bertalanffy L., *General System Theory*.
- [88] Vranic D. and Saupe D., *3d model retrieval with spherical harmonics and moments*, proceedings of the DAGM, 2001.
- [89] Wiley D.A., *The Instructional Use of Learning Objects*, 2002.
- [90] Winston P., *The semantics of Programming Languages*, MIT Press, 1991.
- [91] Zadeh L.A., *Similarity and Quasi-ordering*, Fuzzy set and systems, 1969.