Aspetti della modellazione e dell'analisi di dominio nell'ingegneria dei sistemi informativi

Ernesto Compatangelo* Francesco M. Donini† Giovanni Rumolo‡

Sommario

Questa presentazione illustra un approccio alla modellazione di dominio che permette forme di ragionamento tassonomico automatico sugli aspetti statici della conoscenza comportamentale e strutturale utilizzata nella specifica dei requisiti di un sistema informativo. Tale approccio è stato espressamente conceptito per lo sviluppo di una nuova generazione di ambienti di supporto intelligente alla modellazione e all'analisi della conoscenza di dominio nell'ingegneria dei sistemi informativi.

Inquadramento del problema

Per poter essere efficacemente utilizzata, la conoscenza di dominio deve essere rappresentata in modo formale ed elaborata automaticamente. La scelta di tecniche per la rappresentazione della conoscenza e per il ragionamento automatico non è infatti completamente nuova o recente nell'ingegneria dei sistemi informativi (ISI). Già all'inizio degli anni ottanta fu messa in luce l'importanza di una rappresentazione esplicita della conoscenza acquisita durante la specifica dei requisiti di un sistema informativo [1]. Da allora sono stati proposti vari linguaggi e sistemi per rappresentare questo tipo di conoscenza [2, 3]. Ciò nonostante, manca un approccio alla modellazione di dominio che permetta di catturare aspetti comportamentali, oltre che strutturali, utilizzando un unico schema di modellazione in grado di supportare forme specializzate di ragionamento automatico.

L'approccio delineato in questa presentazione e discusso in maggiore dettaglio in [4] rappresenta un tentativo di superare tali limitazioni basandosi su un quadro di riferimento ontologico, metodologico e operativo per la modellazione e l'analisi della conoscenza di dominio. Lo scopo principale di tale quadro di riferimento è quello di fornire all'analista di sistemi informativi un linguaggio, una metodologia e uno strumento di supporto che gli permettano di ragionare automaticamente con basi di conoscenza costituite da concetti di dominio in relazione tra loro e appartenenti a distinte categorie (dati, processi). Cioò vuol dire non solamente composizione gerarchica automatica dei concetti appartenenti a ogni singola categoria, ma anche controllo di consistenza della base di conoscenza, individuazione di errori semantici, derivazione di proprietà nascoste, risposte a query complesse. Il quadro di riferimento propostoè incentrato su $\mathcal{EDDL}_{DP}$, un linguaggio di concetti orientato agli analisti che permette la descrizione di dati e processi, considerati come fondamentali astrazioni di dominio in ambito ISI. La formalizzazione di $\mathcal{EDDL}_{DP}$ deriva dalla sua mappatura in $\mathcal{TDDL}$, un linguaggio terminologico di rappresentazione della conoscenza appartenente alla famiglia KL-ONE [5].

Il linguaggio $\mathcal{EDDL}_{DP}$ permette la rappresentazione integrata di due fondamentali categorie di concetti epistemologici: *meta-dato* e *meta-processo*. La loro semantica è caratterizzata in uno stile *set- theoretic* totalmente indipendente dalle categorie dei concetti rappresentati. Ogni espressione di uno specifico concetto epistemologico viene trasformata in un insieme di espressioni terminologiche attraverso un particolare gruppo di regole di riscrittura [4]. La rappresentazione epistemologica presuppone l'esistenza di una ontologia chiusa, la quale definisce le caratteristiche delle strutture introdotte per rappresentare i concetti presenti nel dominio applicativo. In generale queste strutture hanno descrizioni e regole di classificazione diverse. Per contro, la rappresentazione terminologica presuppone l'esistenza di una ontologia aperta. Quest'ultima, che è necessaria per descrivere gli operatori base utilizzati per la costruzione dei concetti, assume il ruolo di algebra di rappresentazione. Le espressioni terminologiche vengono quindi manipolate da un ragionatore automatico senza alcun riferimento al significato epistemologico del loro contenuto informativo.

Mentre il meta-dato è una specializzazione epistemologica utilizzata per descrivere concetti strutturali, il meta-processo è una specializzazione epistemologica utilizzata per descrivere svariati tipi di concetti comportamentali di mondo reale basati sull'idea di trasformazione. Il meta-processo è un meta-concetto fondamentale, le cui istanze assumono un significato che comprende sia la tradizionale interpretazione come elemento di elaborazione sia la piú ampia accezione di attività espletata nel dominio. Ciascun processo, istanza del meta-processo, è

[0] Istituto di Informatica della Facoltà di Ingegneria, Università di Ancona – email compatan@inform.unian.it
[0] Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza" – email donini@dis.uniroma1.it
[0] Dipartimento di Informatica ed Automazione, Università di Roma Tre – email rumolo@inf.uniroma3.it

un'astrazione di una collezione di trasformazioni tra individui autonomi che popolano l'UoD. Contrariamente ad alcuni approcci correntemente adottati nell'ingegneria dei sistemi informativi, che considerano i processi come entità comportamentali descritte indipendentemente dalle entità strutturali su cui essi operano, ogni processo in $\mathcal{EDDL}_{DP}$ viene definito in termini di componenti di ingresso e uscita del corrispondente *contesto*. Quest'ultimo rappresenta la componente *statica* della conoscenza "esterna" su un processo, descritto come un concetto che possiede due attributi funzionali obbligatori, detti rispettivamente ingresso (**input**) e uscita (**output**).

Risultati conseguiti

L'approccio proposto permette di lavorare con concetti di dominio utilizzando una sintassi vicina a quella degli analisti, fornendogli dei meta-concetti predefiniti con un sufficiente margine di libertà espressiva. L'esigenza di tale tipo di sintassi descrittiva, associata ad una certa rigidità nella capacità espressiva, è particolarmente sentita dagli utilizzatori. Il non soddisfacimento di tale esigenza in passato ha determinato la mancata valorizzazione delle logiche descrittive in ambito ISI. Allo stesso tempo, alla sintassi $\mathcal{EDDL}_{DP}$ si associa una semantica ben definita, che permette di operare su una base di conoscenza di dominio con tecniche di ragionamento automatico. La mappatura di una descrizione epistemologica in una terminologica consente infatti di sfruttare i servizi deduttivi offerti dai sistemi di rappresentazione terminologica della conoscenza.

Da un punto di vista implementativo, è in corso di sviluppo un ambiente intelligente di gestione della conoscenza, detto sistema $\mathcal{JANUS}$, che supporta gli aspetti ontologici e metodologici espressi nell'approccio $\mathcal{DD}$. I primi risultati d'uso di tale sistema, ancorché parziali, indicano che $\mathcal{JANUS}$ permette ad utenti modellisticamente non evoluti di costruire descrizioni di dominio estremamente complesse e non contraddittorie. Il prototipo di tale sistema, attualmente in fase di sviluppo, è realizzato in ambiente Linux integrando tre componenti eterogenee: un sottosistema di interfaccia con l'utente, un compilatore ed un sottosistema per il ragionamento automatico. L'interazione con l'utente avviene attraverso una applicazione Tcl/Tk in ambiente Xwindows, che oltre a gestire l'interfaccia grafica ha il ruolo di integratore e coordinatore delle altre sottocomponenti. Il compilatore, realizzato in C ANSI, permette la traduzione da e verso il sottosistema di ragionamento. In analogia con gli usuali compilatori per i linguaggi di programmazione di alto livello esso traduce le descrizioni di dominio nelle espressioni di basso livello necessarie al sottosistema di rappresentazione e gestione della conoscenza. Non essendo disponibile al momento attuale un sottosistema il cui linguaggio per la rappresentazione abbia lo stesso potere espressivo del linguaggio $\mathcal{TDDL}$ (che è la controparte terminologica del linguaggio epsitemologico $\mathcal{EDDL}_{DP}$) si è deciso di utilizzare il sistema Classic della AT&T. La corrispondente riduzione di potere espressivo è stata compensata introducendo opportuni meccanismi procedurali implementati come *rule* del linguaggio di Classic. Questa esperienza ha fatto maturare negli autori la convinzione che sia necessario studiare il problema di un'architettura aperta di riferimento per i sistemi di ragionamento e rappresentazione della conoscenza basati sulle logiche descrittive. Questa architettura deve garantire, attraverso l'utilizzo di componenti modulari ed integrabili, la realizzazione di progetti applicativi che facciano esplicito uso di tecniche per il ragionameto automatico e la rappresentazione della conoscenza.

L'esperienza finora portata avanti ci ha permesso di individuare un contesto metodologico entro il quale affrontare il porblema della analisi di dominio per i sistemi informativi. All'interno di questo contesto è stato possibile progettare un linguaggio per l'analisi di dominio di facile utilizzo per gli analisti e che nello stesso tempo fosse in grado di catturare gli schemi normalmente realizzati con gli approcci semiformali. Questo linguaggio permette di rappresentare, in modo integrato rispetto alla componente strutturale, anche gli aspetti statici della componente comportamentale di una analisi di dominio. Infine la nostra esperienza ha messo in luce l'esigenza di una architettura software di riferimento in grado di dare alla diffusione in campo industriale di prodotti applicativi basati sulla rappresentazione della conoscenza la stessa spinta che ha permesso ad analoghe proposte nel settore Object Oriented di ottenere una così vasta diffusione di prodotti e sistemi.

Riferimenti bibliografici

[1] A. Borgida, S. Greenspan, and J. Mylopoulos. Knowledge Representation as the basis for Requirements Specification. *IEEE Computer*, pages 82–91, Apr. 1985.

[2] J. Mylopoulos et al. Telos: Representing Knowledge About Information Systems. *ACM Transactions on Information Systems*, 8(4):325–362, Oct. 1990.

[3] S. Bergamaschi, S. Lodi, and C. Sartori. The E/S Knowledge Representation System. *Data & Knowledge Engineering*, 14:81–115, 1994.

[4] E. Compatangelo and G. Rumolo. $\mathcal{EDDL}_{DP} + \mathcal{TDDL}_{DP} =$ a double-level approach to Domain Knowledge Modelling. In H. Kangassalo and J.F. Nilsson, editors, *Information Modelling and Knowledge Bases VIII*. IOS Press, 1997.

[5] W. A. Woods and J. G. Schmolze. The KL-ONE family. *Computers and Mathematics with Applications*, 23(2-9):1–50, 1992.