



EmbASP, un framework per l'integrazione di moduli di ragionamento dichiarativo in sistemi esterni

Laureando

Marco Bellizzi 176239

Relatori

Prof. Francesco Calimeri

Dott. Stefano Germano

Sommario. Questo lavoro di tesi si colloca nell'ambito dell'Intelligenza Artificiale. Analizzeremo i principi base della programmazione logica, concentrandoci poi sui linguaggi logici ASP e PDDL. Presenteremo EmbASP, un framework che supporta l'integrazione di programmi logici ASP e PDDL in sistemi esterni, analizzandone la struttura e il funzionamento. Analizzeremo infine le attività da me svolte durante il periodo di tirocinio, durante il quale ho collaborato con il team di EmbASP per la sua evoluzione.

Contesto e Motivazioni

- > Programmazione logica: un problema viene descritto con un insieme di formule della logica, dunque in forma dichiarativa.
- > Answer Set Programming (ASP): È un linguaggio caratterizzato dalla semantica fissa dei programmi.
 - Elementi base: fatti, regole, vincoli.
 - Estensioni: disgiunzione, funzioni aggregate, vincoli deboli, altre.
- > Planning Domain Definition Languages (PDDL): È un linguaggio che tratta problemi di pianificazione. Problemi di questo tipo si compongono di tre elementi: uno stato iniziale, uno stato finale e un insieme di azioni che, quando applicate, provocano un cambiamento di stato.

Funzionalità Sistema

- EmbASP è un framework che integra moduli di ragionamento dichiarativo in sistemi esterni. L'integrazione avviene attraverso l'esecuzione di un solver, uno strumento informatico in grado di risolvere un programma logico. EmbASP svolge una funzione di intermedizione tra l'utente e il solver, gestendone l'input e l'output.
- > È implementato in 3 linguaggi di programmazione (Java, Python e C#).
 - > Supporta le piattaforme Desktop e Android.
 - > Supporta i linguaggi di programmazione logici ASP e PDDL.
 - > Supporta i solver DLV, DLV2, DLVHEX e Clingo per il linguaggio ASP, e il solver SPD per il linguaggio PDDL.

Architettura

- EmbASP è diviso in 4 moduli:
- > Base: sono presenti le classi fondamentali del framework. L'Handler colleziona l'input, fornitogli attraverso l' InputProgram. Il Service avvia l'esecuzione del solver, e l'output viene poi modellato grazie alla classe Output o Callback.
 - > Platform: sono presenti le estensioni delle classi Handler e Service che gestiscono le piattaforme Desktop e Android.
 - > Languages: sono presenti le estensioni delle classi InputProgram e Output, le quali hanno lo scopo di gestire i differenti solver.
 - > Specialization: in questo modulo sono contenute le estensioni delle classi Service e Output per poter gestire l'esecuzione dei differenti solver.

Conclusione

- Attraverso questo elaborato si è voluto mettere in risalto come la logica abbia caratterizzato i linguaggi di programmazione per l'Intelligenza Artificiale. Abbiamo visto come EmbASP favorisca l'integrazione di moduli di ragionamento dichiarativo in sistemi esterni, attraverso l'esecuzione di un solver in grado di risolvere un problema logico. Abbiamo infine analizzato le attività svolte durante il periodo di tirocinio, le quali riguardano la cura di tutti quegli aspetti pertinenti le fasi di
- > testing
 - > debugging
 - > pubblicazione del software
 - > rilascio del software