

Sisteme multi-agenti

Litera A

Agent (computational / software) = Agent (computational / software)

O nouă paradigmă conceptuală și tehnologică se impune spectaculos la ora actuală în știința și ingineria calculatoarelor, cunoscută sub diverse nume: agenți autonomi, agenți inteligenți, agenți informaționali, sisteme multi-agent (SMA). Aceste denumiri, deși folosite de multe ori ca echivalente, împărtășesc elemente comune dar, în același timp, diferă prin diverse caracteristici.

Deși definiția noțiunii de agent este mult controversată, cele mai multe definiții converg, în general, spre ideea conform căreia un **agent** reprezintă o entitate care funcționează continuu și autonom într-un mediu în care se pot desfășura și alte procese și în care coexistă mai multi agenți. *Agenții software* se definesc prin următoarele caracteristici:

- *autonomie*: agenții operează fără intervenția directă a oamenilor și au control asupra acțiunilor și stărilor lor interne;
- *abilități sociale*: agenții interacționează cu alți agenți (artificiali sau umani) prin intermediul unui limbaj de comunicare;
- *reactivitate*: agenții percep mediul în care evoluează și răspund în timp util la schimbările acestuia;
- *pro-activitate*: agenții sunt capabili, pe lângă executarea acțiunilor ca răspuns la schimbările de mediu, să ia inițiative, deci să manifeste o comportare condusă de scopuri.

Caracteristicile identificate mai sus definesc categoria *agenților software în general*, numiți și agenți slabi, conform unei clasificări făcută de Wooldridge și Jennings. Exemple semnificative din această categorie sunt agenții informaționali, capabili să navigheze în Internet pentru a căuta, selecta, filtra și sorta informații în slujba utilizatorilor sau agenții asistenți digitali personali (PDA = Personal Digital Assistants) care ajută și cooperează cu utilizatorii în rezolvarea problemelor. PDA sunt un fel de secretare electronice care trebuie calificate astfel încât să reacționeze optim la cerințele utilizatorului.

Agent Communication = Comunicare între agenți

Comunicarea într-un SMA poate varia de la forme extrem de simple până la forme sofisticate de comunicare, cum ar fi cele bazate pe actele comunicării. O formă relativ simplă de comunicare este aceea bazată pe schimbul de informații printr-o *structură de tip tablă* ("blackboard") sau de resursă partajată. Tabla este de obicei împărțită în diferite nivele de abstractizare corespunzătoare diverselor nivele de rezolvare a problemelor iar agenții pot scrie sau citi informațiile de interes la nivelul corespunzător acțiunilor lor.

În comunitatea cercetătorilor din domeniul sistemelor multi-agent inteligente s-a admis faptul că într-un astfel de sistem comunicarea înseamnă mai mult decât comunicarea în cazul sistemelor distribuite. În timp ce comunicarea este standardizată în sistemele distribuite, nu există un acord asupra a ceea ce înseamnă comunicare într-un sistemelor multi-agent dar toți cercetătorii sunt de părere că aceasta reprezintă mai mult decât un simplu schimb de mesaje. Din acest motiv, este mai corect să se vorbească despre *interacțiune* între agenți decât despre comunicare.

Interacțiunea între agenți are loc pe cel puțin două nivele. Primul nivel corespunde conținutului informațional al comunicării iar cel de al doilea specifică intențiile agenților în momentul emiterii comunicării. Dacă interacțiunea între agenți este descrisă prin intermediul transmiției și primirii mesajelor, fiecare agent trebuie să fie capabil să deducă intenția emițătorului referitor la mesajul trimis.

În acest context, pornind de la cercetări din domeniul limbajului natural, s-a impus o formă mai sofisticată de comunicare, respectiv cea bazată pe *teoria actelor comunicării* ("theory of speech acts"). Într-un act de comunicare se face distincția între aspectele elocutionale (exprimarea unei propoziții), ilocutionale (efectul intenționat de cel ce comunică) și prelocutionale (efectul asupra destinatarului comunicării). Aceste modele ale dialogului uman au generat propuneri legate de structurarea mesajelor schimbate între agenți în funcție de tipul actului de comunicare și la specificarea protocoalelor adecvate acestor comunicări. În cazul sistemelor multi-agent, protocolul de comunicare nu desemnează un protocol de nivel scăzut ci un protocol ce stabilește acțiunile posibile ale agentului în fiecare moment al comunicării cu alți agenți. Astfel, un protocol de interacțiune restricționează diversele reacții pe care agenții le pot avea extern, în raport cu alți agenți din societate.

Limbajul care tinde să devină un standard pentru comunicarea și protocolul de comunicare între agenți este KQML (vezi Limbaj agent).

Agent Language = Limbaj agent

Cum pot fi programate sistemele multi-agent? O soluție evidentă este aceea a utilizării unui limbaj de programare de nivel înalt. În același timp, s-au dezvoltat însă o serie de limbaje special concepute pentru dezvoltarea și implementarea sistemelor multi-agent. *Un limbaj agent* este un limbaj de programare care încorporează noțiunea de agent.

Deși până la ora actuală nici unul din aceste limbaje nu s-a impus ca un standard (în afară de KQML și KIF dar care se referă preponderent la interacțiune), vom enumera o parte din aceste limbaje, importante în special prin conceptele noi pe care le-au impus și prin influența pe care o au și suntem siguri că o vor avea în modalitatea de abordare a programării în viitor. Corespunzător clasificării sistemelor multi-agent în sisteme cu agenți software (slabi) sau cu agenți cognitivi (tari), identificăm și limbaje agent slabe (ACTOR, TCL/TK, TELESCRIPT) sau limbaje agent tari (AGENT0/PLACA, Concurrent METATEM, KQML, KIF).

În limbajul ACTOR, Hewitt a pus bazele dezvoltării limbajelor orientate obiect concurente. Cercetările lui, efectuate împreună cu Gul Agha, au condus la ceea ce azi identificăm a fi limbajele ACTOR. Actorii sunt sociali, deci sunt capabili să trimită mesaje altor actori, și sunt reactivi, executând prelucrări ca răspuns la mesaje primite de la alți actori. Limbajul TCL (Tool Control Language) împreună cu TK oferă facilități de implementare a unor limbaje de comandă particulare. El este un limbaj interpretat, oferind o serie de primitive scrise în C++, disponibile în domeniul public și putând deci fi incluse în construcția diverselor aplicații. TELESCRIPT este un mediu de construire a SMA dezvoltat de General Magic, care conține două concepte cheie: locuri ("places") și agenți ("agents"). Locurile sunt locațiile virtuale ocupate de agenți, care pot corespunde unei mașini sau unei familii de mașini, iar agenții sunt furnizori sau consumatori de bunuri în piața electronică ce poate fi construită în mediul TELESCRIPT. Agenții sunt mobili și folosesc un bilet de călătorie (ce specifică destinația și timpul de sosire) care le permite mișcarea în rețea.

Primul și cel mai influent, din punct de vedere conceptual, limbaj de programare a agenților cognitivi este AGENT0 cu succesorul lui PLACA. AGENT0 a fost conceput de Shoham care a introdus, odată cu acest limbaj, o nouă paradigmă în programare: programarea orientată pe agenți (AOP=Agent-Oriented Programming). Programarea orientată pe agenți poate fi văzută ca o specializare a paradigmei programării orientate pe obiecte. Intuitiv, în timp ce programarea orientată pe obiecte propune considerarea unui sistem de prelucrare ca o colecție de module care comunică între ele prin transmitere de mesaje, AOP socializează acest model prin fixarea stărilor (numite acum stări mentale) acestor module (numite acum agenți), stări care sunt compuse din componente mentale (convingeri, intenții, obligații, angajamente, decizii, etc.). În acest context, obiectele sunt specializate în module intenționale. O prelucrare într-un astfel de model constă în acțiunile acestor agenți, acțiuni prin care agenții cer sau oferă informații, asistă sau intră în competiție cu alți agenți existenți în sistem. Comunicarea între agenți diferă în funcție de tipul de comunicare intenționată iar efectele acesteia sunt diferite în funcție de intențiile particulare.

Un alt limbaj agent tare este Concurrent METATERM în care comportarea agenților este modelată pe baza unei logici modale temporale. Un program METATERM este o mulțime de reguli de forma "trecut implică viitor" în care agenții sunt caracterizați prin operatori modali temporali de tipul "acum și întotdeauna", "cândva în viitor", "cândva în trecut", etc.

Unul dintre cele mai cunoscute limbaje și protocoale de comunicare bazate pe teoria actelor comunicării este KQML (Knowledge Query and Manipulation Language) propus de ARPA Knowledge Sharing Effort în 1992, cu ultima versiune standardizată apărută la începutul anului 1997. KQML folosește limbajul KIF (Knowledge Interchange Format) pentru a descrie conținutul informațional al mesajului. Limbajul KIF este un fel de reprezentare ASCII a logicii cu predicate de ordinul întâi, într-o sintaxă extrem de apropiată de cea a limbajului LISP. KQML și KIF (cu succesorul lui SIF - Simple Knowledge Interchange Format) tind să devină la ora actuală un fel de standard al comunicării în SMA cognitive. Urmează câteva exemple de primitive KQML:

- (ask A) - emițătorul întreabă receptorul dacă A este o convingere a receptorului;
- (tell A) - emițătorul comunică receptorului convingerea sa A;
- (achieve A) - emițătorul cere receptorului să includă A în mulțimea lui de convingeri;
- (deny A) - emițătorul comunică receptorului că nu crede A.

(vezi și Comunicare între agenți)

Agent-Oriented Programming = Programare orientată agent

Programarea orientată agent (AOP = Agent-Oriented Programming) este o abordare în construcția sistemelor multi-agent în

care programarea agenților se face în termeni de noțiuni mentale, de exemplu convingeri, dorințe și intenții.

Noțiunea a fost introdusă de Yoav Shoham odată cu conceperea limbajului AGENT0. Programarea orientată pe agenți poate fi văzută ca o specializare a paradigmei programării orientate pe obiecte. Intuitiv, în timp ce programarea orientată pe obiecte propune considerarea unui sistem de prelucrare ca o colecție de module care comunică între ele prin transmitere de mesaje, AOP se specializează acest model prin fixarea stărilor (numite acum stări mentale) acestor module (numite acum agenți), stări care sunt compuse din componente mentale (convingeri, intenții, obligații, angajamente, decizii, etc.).

Vezi și Limbaj agent.

Agent Rationality = Raționalitatea unui agent

Raționalitatea unui agent este presupunerea conform căreia agentul va acționa astfel încât să-și realizeze scopurile și nu va acționa, în mod conștient, astfel încât să împiedice satisfacerea acestor scopuri. Vezi și Arhitectură BDI.

Litera B

BDI Architecture = Arhitectură BDI

O *arhitectură BDI* (BDI = Belief, Desire, Intention) este arhitectura unui sistem multi-agent care conține o reprezentare explicită a convingerilor, dorințelor și intențiilor agentului. Convingerile sunt, de obicei, informațiile pe care un agent le are despre mediul în care evoluează sau despre ceilalți agenți din sistem; convingerile pot fi adevărate sau false (agentul se înșeală). Dorințele sunt lucrurile pe care agentul ar dori să le vadă realizate. Dorințele nu trebuie neapărat să fie consistente iar agentul nu trebuie neapărat să acționeze pentru îndeplinirea tuturor dorințelor sale. Intențiile sunt acele lucruri pe care agentul s-a hotărât să le realizeze sau s-a angajat să le realizeze.

Beliefs = Convingeri

Vezi Arhitectură BDI

Blackboard Architecture = Arhitectură de tip tablă

Arhitectura de tip tablă este o arhitectură în care o colecție de surse de cunoștințe comunică prin scrierea într-o structură accesibilă global numită tablă.

Vezi și Comunicare între agenți.

Litera C

Cognitive agents = Agenți cognitivi

Vezi agenți inteligenți

Cognitive Multi-Agent Systems = Sisteme multi-agent cognitive

Vezi sisteme multi-agent

Cognitive State = Stare cognitivă

Starea cognitivă este starea internă a unui sistem intențional și este reprezentată, în mod tipic, prin convingerile, dorințele, intențiile, etc., care caracterizează agentul la un moment dat.

Litera D

Deliberative Architecture = Arhitectură deliberativă

O *arhitectură* a unui sistem multi-agent se numește *deliberativă* dacă se bazează pe reprezentarea explicită a unui model simbolic și pe manipularea de simboluri. De obicei, astfel de arhitecturi corespund sistemelor multi-agent cognitive.

Vezi și Arhitectură BDI.

Desires = Dorințe

Vezi Arhitectură BDI

Digital Personal Assistant = Asistent Digital Personal

Un asistent digital personal (PDA = Digital Personal Assistant) este un agent care colaborează cu utilizatorul pentru realizarea unei sarcini a utilizatorului. În loc de a răspunde la instrucțiunile utilizatorului, un PDA învață modul de lucru al utilizatorului și începe să ia inițiative selectând și executând în mod autonom acțiuni pentru realizarea sarcinilor.

Litera I

Information Agent = Agent informațional

Agentul informațional este un agent capabil să răspundă la întrebări puse de utilizator prin colectarea, filtrarea și manipularea informațiilor din diverse surse, de obicei surse disponibile electronic în Internet (Web).

Intelligent Agents = Agenți inteligenți

Agenții inteligenți, numiți și *agenți cognitivi* sau agenți tari au toate proprietățile agenților software (autonomie, abilități sociale, reactivitate, proactivitate) dar, în plus, sunt conceptualizați folosind noțiuni ce se aplică de obicei oamenilor, cum ar fi :

- *noțiuni mentale epistemice* (convingeri, cunoștințe);
- *noțiuni mentale de atitudine* (intenții, obligații, angajamente, decizii, etc.);
- *raționalitate* (în sensul capacității de acțiune efectivă pentru atingerea unui scop);
- *adaptabilitate* sau capacitate de *învățare*.

Din acest punct de vedere, agenții sunt văzuți ca sisteme intenționale, cu scopuri, intenții și atitudini proprii. De ce sunt priviți agenții inteligenți ca *sisteme intenționale*? Atunci când explicăm activitatea umană, este de multe ori util să facem afirmații de tipul: "Maria și-a luat umbrela deoarece a crezut că va ploua" sau "Mihai muncește pe rupte deoarece dorește să fie promovat." Așa cum spune Shoham, aceste afirmații fac apel la un fel de "psihologie folclorică" prin care comportarea umană poate fi explicată sau prezisă prin atribuirea de atitudini cum ar fi convingeri, dorințe, angajamente, speranțe, etc. Atitudinile utilizate în acest tip de psihologie folclorică sunt de fapt noțiuni intenționale.

Filozoful Daniel Dennett a definit sistemele intenționale ca entități a căror "comportare poate fi prezisă pe baza metodei prin care li se atribuie convingeri, dorințe și atitudini raționale". Marele cercetător în inteligență artificială John McCarthy spune că "Atribuirea de convingeri, voință, intenții, abilități și dorințe unor sisteme artificiale este legitimă atunci când acestea permit exprimarea acelorași informații despre sistem ca și în cazul oamenilor. Este utilă atunci când această atribuire ne ajută să înțelegem structura sistemului, comportarea trecută sau viitoare sau cum să depanăm sau să îmbunătățim acel sistem." Ce fel de sisteme sau obiecte pot și merita să fie descrise utilizând acest model intențional, cognitiv? Shoham, un deschizător de drumuri în domeniul SMA și creatorul conceptului de programare orientată pe agenți, spune: "Putem considera că un comutator al unei lămpi este un agent (foarte cooperativ) care este capabil să transmită curent după dorința noastră, care transmite curent atunci când crede că utilizatorul dorește lumină și nu mai transmite în caz contrar. Apăsarea comutatorului este modalitatea noastră de comunicare și de interacțiune socială cu acel agent. Dar cei mai mulți oameni ar considera o astfel de descriere ridicolă. De ce? Deoarece o astfel de abordare nu ne ajută cu nimic, mecanismul de funcționare al întrerupătorului fiind simplu și perfect cunoscut." Pe măsură ce sistemele de modelat devin din ce în ce mai complexe și mai greu de descris funcțional, așa cum este cazul sistemelor descrise cu ajutorul tehnologiei sistemelor multi-agent, abstractizarea componentelor acestor sisteme pe baza intenționalității lor se impune ca o paradigmă utilă în descrierea funcționării și comportării acestor sisteme.

Intentional System = Sistem intențional

Sistemul intențional este un sistem a cărui comportare poate fi prezisă sau explicată prin atribuirea de atitudini intenționale cum ar fi convingerile, dorințele, intențiile, etc., și a unui anumit grad de raționalitate. Vezi și Sisteme multi-agent, Programare orientată agent, Stare cognitivă.

Intentions = Intenții

Vezi Arhitectură BDI

Interface Agents = Agenți de interfață

Un *agent de interfață* este un agent care acționează ca un asistent al utilizatorului în utilizarea unei anumite aplicații. Dezvoltarea agenților de interfață este legată și de activitatea cooperativă susținută de calculator în care agenți umani și artificiali cooperează în mod inteligent pentru rezolvarea problemelor cu ajutorul calculatorului (vezi și activitatea cooperativă susținută de calculator).

Litera M

Mobility = Mobilitate

Mobilitatea unui agent este capacitatea unui agent de a călători într-o rețea de calculatoare. Exemplul tipic este cel al agenților din limbajul TELESRIPT.

Multi-agent systems = Sisteme multi-agent

Apariția sistemelor deschise și descentralizarea proiectării distribuite a impus o nouă abordare a rezolvării distribuite a problemelor, respectiv *sistemele multi-agent* (SMA). Un sistem multi-agent este un sistem distribuit format dintr-o colecție de agenți autonomi care interacționează într-un mediu comun, fiecare agent având cunoștințe, capacități de acțiune și scopuri proprii. De cele mai multe ori, sistemele multi-agent sunt sisteme inteligente.

Se pot identifica două categorii de sisteme multi-agent inteligente: SMA cu agenți cognitivi (vezi agenți inteligenți sau agenți cognitivi) numite și sisteme multi-agent cognitive, și SMA cu agenți reactivi, numite și sisteme multi-agent reactive.

Sistemele multi-agent cognitive încearcă să simuleze elemente ale modelului uman de comportare prin includerea noțiunilor de scop, cooperare, competiție, organizare în structuri sociale și stabilirea relațiilor de dependență în aceste structuri, capacitate de învățare și auto-perfecționare. O adevărată efervescență a cuprins comunitatea de inteligență artificială referitor la modele formale și informale, sisteme și componente program care sunt realizate pe baza acestor noțiuni și paradigme, domeniul SMA inteligente fiind un "hot-topic" al inteligenței artificiale.

Un sistem multi-agent cognitiv poate fi văzut ca un sistem particular bazat pe cunoștințe ce conține o reprezentare simbolică a cunoștințelor despre lumea în care evoluează și care este capabil să ia decizii (de exemplu ce acțiune să execute) prin intermediul unui proces inferențial (de raționament). Spre deosebire de un sistem bazat pe cunoștințe clasic, un agent într-un SMA reprezintă simbolic lumea atât prin convingeri, care sunt păreri despre lume, eventual incomplete sau eronate, cât și prin cunoștințe, ce reprezintă fapte adevărate despre acea lume. În plus, un agent trebuie să fie capabil să raționeze atât despre propriile lui convingeri și cunoștințe cât și despre cele ale celorlalți agenți cu care interacționează într-un anumit mediu.

Sistemele multi-agent reactive sunt sisteme în care agenții sunt unități simple de prelucrare, capabile să reacționeze la schimbările de mediu și să execute acțiuni simple în consecință. Modelul este în principal inspirat din structura comunităților biologice de insecte: o albină nu poate fi considerată inteligentă dar comportamentului stupului ca întreg și chiar modul de organizare a albinelor în acesta prezintă cert elemente de inteligență, executând acțiuni comune și coordonate în vederea realizării scopului. Conform adeptilor SMA reactive, inteligența reală este situată în lumea exterioară și nu la nivelul componentelor de prelucrare individuale, respectiv agenții. În plus, comportarea inteligentă este un rezultat al interacțiunii dintre agenți și mediu iar inteligența este o proprietate emergentă a sistemului ca întreg.

Adeptii SMA reactive aduc mai multe critici abordării cognitive a agenților, în special referitor la complexitatea de calcul și dificultatea algoritmilor necesari prelucrărilor simbolice. O întreagă direcție de cercetare se îndreaptă spre acest punct de vedere, concretizată până în prezent prin propunerea unor sisteme cu arhitecturi de subsumare sau care modelează îndeaproape structura și comportamentul colectivităților de insecte. Alte abordări subsimbolice din inteligență artificială, cum

ar fi rețelele neurale sau algoritmi genetici, vin să influențeze această nouă direcție. și, ca întotdeauna, unii cercetători susțin că soluția constă de fapt într-o abordare hibridă cognitiv-reactivă.

Vezi și agenți inteligenți.

Litera R

Reactive Architecture = Arhitectură reactivă

O arhitectură reactivă este arhitectura unui sistem multi-agent în care nu se folosește nici un fel de model simbolic centralizat al lumii, așa cum se face în mod tipic în sistemele bazate pe cunoștințe din inteligență artificială, și nu există raționament simbolic. Într-o astfel de arhitectură, agenții sunt reprezentați prin unități de prelucrare foarte simple iar inteligența sistemului este emergentă din interacțiune unităților de prelucrare cu mediul.

Vezi și sisteme multi-agent.

Reactive Multi-Agent Systems = Sisteme multi-agent reactive

Vezi sisteme multi-agent