

## Tehnologia Microsoft Agent pentru învățarea regulilor de functionare a conturilor

Lect. Bogdan PATRUT  
Universitatea din Bacău

*This paper briefly describes the Microsoft Agent technology for constructing interactive user interfaces. Then there is presented a program in Visual Basic that allows user to learn the rules about the accounts.*

**Keywords:** *intelligent software agents, accounting, user interface, Visual Basic.*

### **A**gentii inteligenti

Un agent este un program autonom, care aflat într-un mediu oarecare încearcă să-și atingă obiectivele pentru care a fost proiectat. Agentul este autonom în sensul că este capabil să acționeze fără intervenția oamenilor sau a altor sisteme, el deținând controlul atât asupra stării sale interne, cât și asupra comportamentului său.

Agentul preia informații din mediu prin senzori și poate acționa prin efectori asupra mediului, modificându-l. Un agent inteligent ar fi un agent software capabil de acțiuni flexibile, realizate în vederea îndeplinirii obiectivelor sale. Prin flexibilitatea unui agent software se înțelege că acesta este:

- *reactiv*, adică percepe mediul și răspunde rapid la schimbările din mediu, pentru a-și atinge obiectivele;
- *pro-activ*, adică are un comportament orientat pe țel, având inițiativa în vederea îndeplinirii obiectivelor;
- *competent social*, adică agentul inteligent poate să interacționeze cu alți agenți sau cu utilizatorul pentru a-și atinge obiectivele.

Este greu de imaginat că se pot crea asemenea agenți, mai ales care să aibă inițiativa și, deși există cercetători care desfasoară o muncă susținută în domeniu, există și voci critice, care spun că agenții sunt niste programe ca toate celelalte. Când privește reactivitatea, aceasta nu este greu de proiectat, dacă se folosesc modele teoretice și tehnici de implementare corespunzătoare și se au în vedere cât mai multe intrări posibile pentru senzorii agenților. O parte dintre problemele de abilitate socială sunt deja rezolvate la mulți agenți. Este mai ales cazul realizării unei in-

teracțiuni agent software - utilizator uman cât mai asemănătoare cu o interacțiune interumană naturală.

### **Tehnologia Microsoft Agent**

O tehnologie care se bucură de un succes mare în rândul dezvoltatorilor de aplicații este Microsoft Agent, care pune la dispoziția programatorului mici caractere animate, care se manifestă uman și social și se pot comporta ca niște ghizi, instructori, prezentatori.

Microsoft Agent oferă o tehnologie modernă pentru crearea de interfețe conversaționale inovative pentru aplicații și pagini Web, fiind un punct de plecare pentru modalități de comunicare mult mai naturale între oameni și calculatoarele lor. MS Agent este o multime de servicii software care permit programatorilor să încorporeze personaje sau caractere animate interactive în software-ul lor, mai ales în cel cu specific educațional. Aceste caractere pot vorbi, se pot anima și pot gestiona în diferite feluri, se pot deplasa pe ecran dintr-un loc într-altul sau chiar pot înregistra secvențe audio și accepta comenzi vocale.

Pentru a folosi Microsoft Agent în Visual Basic, de exemplu, trebuie ca mai întâi să descărcăm de pe site-ul <http://www.microsoft.com/products/msagent/downloads.htm>, fișierul MsAgent.exe. Apoi va trebui să descărcăm unul sau mai multe caractere (personaje). Putem alege personajele de la Microsoft (Genie, Merlin, Robby și Peedy) sau alte câteva zeci pe care le putem descărca de la terți (de exemplu noi l-am ales pe James de la <http://www.msagentring.org/>). Împreună cu personajele trebuie descărcată și lista de comenzi asociate fiecărui personaj, fără de care

ne va fi dificil sa-l coordonam.

Daca dorim sa folosim facilitatea "text-to-speech" vom avea nevoie de un "motor" capabil sa reproduca textul în limba dorita. Din nefericire cei de la Microsoft nu ofera un astfel de motor si pentru limba româna, însa se poate folosi cu succes limba italiana, în multe cazuri; eventual se pot face mici artificii, asupra carora vom reveni. Pentru ca mesajele pe care le afiseaza un personaj sa fie afisate corect în limba româna se va descarca de la Microsoft fisierul *AgtX0418.exe*. Apoi se vor descarca fisierele pentru vorbire în limbile dorite (de exemplu, *lhttsiti.exe* pentru limba

italiana). Dupa descarcarea tuturor executabililor respective, se vor lansa în executie pe rând, pentru ca sa se instaleze corect pe calculatorul dumneavoastra.

### Folosirea interfeței caracter-personaj

O data instalata interfața Microsoft Agent, o putem folosi într-un program Visual Basic adaugând proiectului nostru un control de tip MS Agent 2.0, asa cum este indicat în figura 1, care reprezinta interfața unui program de învățare a regulilor de functionare a conturilor.

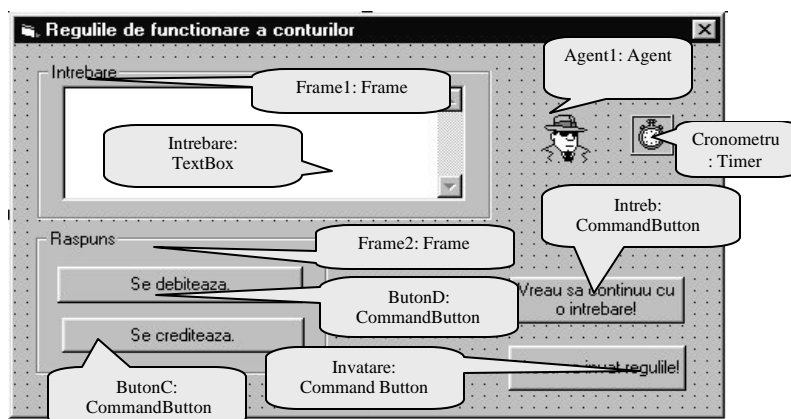


Fig. 1. Interfața programului

Putem apoi folosi metodele *Speak* - pentru a vorbi sau *MoveTo* pentru a deplasa personajul într-o locație data. Cu *GestureAt* îl putem face sa indice cu mâinile un anumit loc, iar cu *Play* avem la dispozitie o sumedenie de animatii, încorporate în fiecare personaj si care difera de la unul la altul.

Declaratiile de variabile nu sunt chiar asa de complicate, pot fi învățate pe baza unui exemplu, cert este ca un agent poate fi reprezentat sau poate controla si mai multe caractere (personaje). Astfel, daca ati instalat controlul *Agent1* de tip *MS Agent 2.0*, puteti încarca caracterul *James* din fisierul *james.acs*, numindu-l (de exemplu) *Profesor* astfel:

```
Dim Profesor As IAgentCtlCharacterEx
Agent1.Characters.Load "Profesor",
"james.acs"
Set Profesor = Agent1.Characters ("Profesor")
```

Apoi puteti stabili interfața cu mesaje în limba româna, iar vocea sa fie a unui italian, pe nume Stefano (consultati fisierul "help" pe n-

tru detalii privind proprietatea *TTSMModeID*). Cu metoda *Show* îl aduceti pe ecran pe "Profesor".

```
Profesor.LanguageID = &H409 ' mesajele de eorare si informatii in limba romana
Profesor.TTSMModeID = "{7EF71701-A92D-11d1-B17B-0020AFED142E}"
' va vorbi in limba italiana, cea mai apropiata de romana,
' codul este pentru vocea masculina cu numele "Stefano"
Profesor.Show
```

Este simplu sa-l faci pe profesorul James sa vorbeasca, prin metoda *Speak*. Astfel, daca vrei ca James sa spuna "Gresit!", vei scrie *Profesor.Speak "Gresit!"*. Dar va pronunța în italiana, asa ca poti sa-l pacalesti puțin, folosind atributul *Map*, astfel:

```
Profesor.Speak
"\Map=""Greshit!""=""Gresit!""", în care primul text este cel vorbit în realitate (si va suna bine!), iar cel de al doilea este textul care va aparea într-un balon explicativ deasupra capului lui James.
```

## Realizarea unui program de învățare a regulilor de funcționare a conturilor

În continuare vom prezenta un program care va avea două funcțiuni: de învățare a regulilor de funcționare a conturilor și de interogare a utilizatorului cu privire la operația care are loc asupra unui cont. În prima două variante se va afișa un text simplu, preluat din fișierul "reguli.txt", cu următorul conținut:

Regulile de funcționare a conturilor sunt:

1. Dacă X este cont de activ și X crește, atunci X se debitează.
2. Dacă X este cont de activ și X scade, atunci X se creditează.

3. Dacă X este cont de pasiv și X scade, atunci X se debitează.

4. Dacă X este cont de pasiv și X crește, atunci X se creditează.

Conturile "bifuncționale" se comportă după regula corespunzătoare în funcție de funcționalitatea lor curentă.

În cea de-a doua, utilizatorul va trebui să răspundă, prin click pe butonul corespunzător, la o întrebare afișată în caseta de text Întrebare. Această întrebare este generată automat, aleatoriu, și este de forma celei din figura 2.

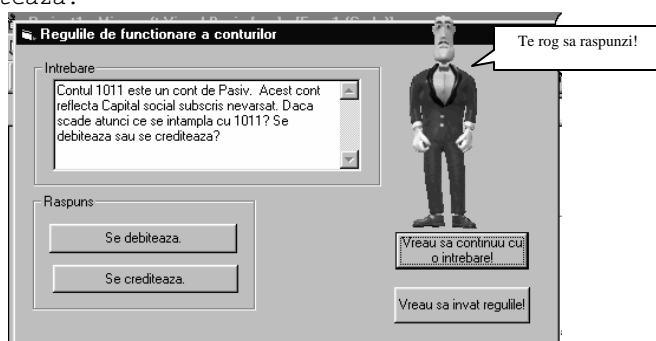


Fig. 2. Funcționarea programului în varianta verificare

Contul CONT este un cont de TIPUL\_CONTULUI. Acest cont reflectă DENUMIREA\_CONTULUI. Dacă TIP\_MODIFICARE atunci ce se întâmplă cu CONT? Se debitează sau se creditează?

Pentru a genera textul vorbit de personajul James se va folosi un cronometru special și variabila *secunde* care descrește o dată cu trecerea timpului. James va ține cont și de intrările senzoriale ale agentului (reprezentate de acțiunile utilizatorului, răspunsurile sau intențiile sale). Astfel personajul va comunica utilizatorului un mesaj, în conformitate cu contextul dat. Dar chiar în cazul alegerii unui mesaj, aceasta va fi comunicat în una sau alta dintre variantele disponibile, folosind opera-

torul "|" în metoda *Speak*. De exemplu

```
Professor.Speak
"\Map="Greshit!"=="Gresit!" "\Nu e bine. |Eronat."
```

va alege una dintre cele trei variante pentru a le vorbi. De remarcat că profesorul James va pronunța chiar și cuvântul "Gresit" de stul de corect, folosind artifiциul adaptării regulilor de pronunție din limba italiană la limba română. În cazul prezentării regulilor de funcționare a conturilor, profesorul se va deplasa într-un alt loc de pe fereastra programului și va ridica mâna pentru a indica explicațiile respective (figura 3).

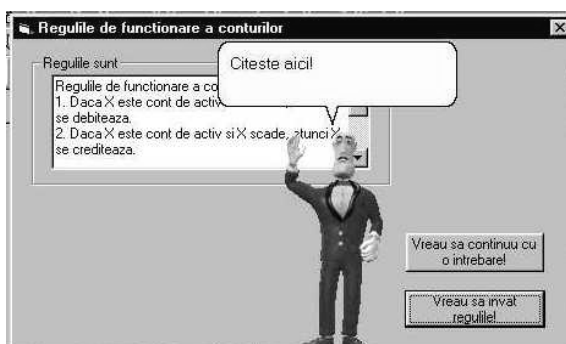


Fig. 3. Explicarea regulilor de funcționare a conturilor

## Programul JAMES-CONT

Aplicatia Visual Basic descrisa poate fi descarcata de pe Internet de la adresa [www.edusoft.ro](http://www.edusoft.ro), dar mai jos este prezentat si explicat listing-ul programului:

```
Dim text_din_nou As String ' variante de
text pentru reluarea raspunsului
Dim text_raspunde As String ' variante
de text pentru solicitarea raspunsului
Dim text_felicitari As String ' variante
de text pentru continuare
Dim text_alta_intrebare As String ' va-
riante de text pentru alta intrebare
Dim secunde As Integer ' numarul de se-
cunde ramase
Dim Profesor As IAgentCtlCharacterEx '
caracterul/personajul James
Dim raspuns As Integer '1 = se
debiteaza, 2 = se crediteaza
Dim raspuns_dat As Integer '0 = nu a
raspuns, 1 = a raspuns corect, 2 = a
raspuns gresit
Dim ptop As Integer 'ordonata initiala a
personajului
Private Sub ButonC_Click()
' alegerea butonului cu raspunsul
credit, 2 - corect
If raspuns = 2 Then
Profesor.Speak "Este co-
rect!|Corect!|OK!"
raspuns_dat = 1
Else
Profesor.Speak
"\Map="Greshit!"="Gresit!"\Nu e bi-
ne.|Eronat."
raspuns_dat = 2
End If
End Sub
Private Sub ButonD_Click()
' alegerea butonului cu raspunsul
debit, 2 - corect
If raspuns = 1 Then
Profesor.Speak "Este co-
rect!|Corect!|OK!"
raspuns_dat = 1
Else
Profesor.Speak
"\Map="Greshit!"="Gresit!"\Nu e bi-
ne.|Eronat."
raspuns_dat = 2
End If
End Sub
Private Sub Cronometru_Timer()
secunde = secunde - 1
If (secunde = 15) And (raspuns_dat =
0) Then
' daca in primele 15 secunde nu
se raspunde, se avertizeaza
Profesor.Speak "Apsa pe un bu-
ton!|Raspunde!"
Profesor.MoveTo 200, 100 'mutare
personaj
Profesor.GestureAt 150, 300 'in-
dicarea butoanelor
End If
```

```
If secunde = 0 Then 'timul s-a scurs
If raspuns_dat = 2 Then '
raspunsul a fost gresit
Profesor.Speak text_din_nou
raspuns_dat = 0
secunde = 20
ElseIf raspuns_dat = 1 Then '
raspunsul dat a fost corect
Profesor.Speak
text_felicitari
Profesor.MoveTo 300, ptop
Intreb_Click
Else ' nu a raspuns nimic
Profesor.Speak
text_alta_intrebare
Profesor.MoveTo 300, ptop
Intreb_Click
End If
End If
End Sub
Private Sub Form_Load()
Randomize
Framel.Caption = "Intrebare"
Agent1.Characters.Load "Profesor",
"james.acs"
Set Profesor =
Agent1.Characters("Profesor")
Profesor.LanguageID = &H409 ' mesa-
jele de eorare si informatii in limba
romana
Profesor.TTSMODEID = "{7EF71701-
A92D-11d1-B17B-0020AFED142E}" ' va vorbi
in limba italiana, cea mai apropiata de
romana, voce masculina Stefano
Profesor.Show
Form1.Left = 100
Form1.Top = 100
Profesor.MoveTo 300, Profesor.Top
ptop = Profesor.Top
t1 = "\Map="Respunde la
intrebare!"="Raspunde la
intrebare!"\
t2 = "\Map="Hai sa te vad!"="Hai
sa te vad!\
t3 = "\Map="Shtii?"="Stii?"\
t4 = "\Map="Pots samhi
spui?"="Poti sa-mi spui?"\
t5 = "\Map="Te rog sa
raspunz!"="Te rog sa raspunzi!"\
text_raspunde = t1 + "|" + t2 + "|"
+ t3 + "|" + t4 + "|" + t5
text_din_nou = "Incearca din nou!|Ia
mai gandeste-te!|Acum vei sti!"
text_felicitari =
"Felicitari!|Trecem mai departe.|O noua
intrebare.|Sa incercam altceva.|Bravo!"
text_alta_intrebare = "Sa incercam
alta intrebare!|Incearca altceva!|Poate
acum vei reusi"
End Sub
Private Sub Intreb_Click()
If Profesor.Top <> ptop And Profe-
sor.Left <> 300 Then
Profesor.MoveTo 300, ptop
End If
Cronometru.Interval = 1000
Framel.Caption = "Intrebare"
Frame2.Visible = True
```

```

    secunde = 30 ' se alege un cont ca-
recare din fisierul contab.txt
    r = 1 + Int(Rnd * 4) ' nr. de
randuri din fisierul contab.txt
    Open App.Path + "\contab.txt" For
Input As #1
    For i = 1 To r
        Line Input #1, s
    Next i
    Close #1
    ' se determina simbolul, denumirea
contului si tipul acestuia
    p = InStr(s, "=")
    contul = Left(s, p - 1)
    s = Right(s, Len(s) - p)
    p = InStr(s, "=")
    denumirea_contului = Left(s, p - 1)
    tc = Right(s, 1)
    If tc = "a" Then
        tipul_contului = "Activ"
    Else
        tipul_contului = "Pasiv"
    End If
    ' se alege o modificare din cele do-
ua variante
    r = Int(Rnd * 2)
    If r = 0 Then
        modificare = "creste"
    Else
        modificare = "scade"
    End If
    ' se genereaza in s textul
intrebarii
    s = "Contul " + contul
    s = s + " este un cont de " + ti-
pul_contului + ". "
    s = s + " Acest cont reflecta " +
denumirea_contului + ". "
    s = s + "Daca " + modificare
    s = s + " atunci ce se intampla cu "
+ contul + "? "
    s = s + "Se debiteaza sau se
crediteaza?"
    Intrebare.Text = s
    rezultat = tc + "=" + modificare
    ' conditiile de raspuns corect si de
raspuns gresit
    If (rezultat = "a=creste") Or (re-
zultat = "p=scade") Then
        raspuns = 1 ' se debiteaza
    Else
        raspuns = 2 ' se crediteaza
    End If
    raspuns_dat = 0
    Profesor.Speak text_raspunde
    Profesor.GestureAt 300, Profesor.Top
+ 100

```

```

End Sub
Private Sub Invatare_Click()
    ' butonul de Invatare
    Cronometru.Interval = 0
    Frame2.Visible = False
    Profesor.MoveTo 200, 100
    Open App.Path + "\regula.txt" For
Input As #1
    t = ""
    While Not EOF(1)
        Line Input #1, s
        t = t + s + Chr$(13) + Chr$(10)
    Wend
    Close #1
    Frame1.Caption = "Regulile sunt"
    Intrebare.Text = t
    Profesor.Speak "Acestea sunt reguli-
le.|\Map="Citeshte aici!"="Citeste
aici!"\|Iata regulile!"
    Profesor.GestureAt 100, -50
End Sub

```

Se observa, din program, ca acesta are nevoie de cele doua fisiere REGULI.TXT, prezentat deja, si CONTAB.TXT. Acesta contine mai multe rânduri, pe fiecare fiind descris un cont, ca în exemplul nostru:

```

456=Decontari cu asociatii privind
capitalul=a
1011=Capital social subscris
nevarsat=p
321=Obiecte de inventar=a
401=Furnizori=p

```

E momentul ca James sa va ureze succes la exercitiile de contabilitate!

### Bibliografie

- [1] [www.microsoft.com](http://www.microsoft.com) - The Microsoft Company
- [2] The Microsoft Platform SDK - Microsoft Agent 2.0 Help File, Microsoft 1999
- [2] Bogdan Patrut - "Aplicatii în Visual Basic", Editura Teora, 1998-2003
- [3] Daniel Popa - "Microsoft Agent", [www.e-press.ro](http://www.e-press.ro)
- [4] Vasile Patrut, Aristitia Rotila - "Contabilitatea întreprinderii", Editura Alma Mater, Bacau, 2002
- [5] [www.edusoft.ro](http://www.edusoft.ro) - SC EduSoft SRL