

Interactive Learning Systems through Virtual Reality

Prof.dr. Bogdan GHILIC-MICU
Catedra de Informatică Economică, ASE București

Complexity of today's society systems impose the need for learning systems that would ensure both performance of the operators and a means to improve learning at the place of work. Creation of an interactive learning system that is getting closer to the real environment aims at reaching a learning system for the perfect training of the worker. Below there are examined preliminary data associated to this new learning system, based on the theory of learning by experience, putting on display the way in which the educational reform will be made at the future place of work.

Keywords: *e-learning, interactive systems, network, virtual space, collaborative learning.*

Sistemul de învățare interactivă este un sistem educațional colaborativ care permite mai multor cursanți aflați în locații geografice diferite să interacționeze într-un spațiu virtual tridimensional. Pentru a efectua experimente virtuale pot fi folosite și sarcini creative. Cursanții pot să creeze dinamic obiecte noi și să le modifice proprietățile. Sistemul menține starea spațiului, așa cum este văzut de fiecare cursant. Pentru a menține consistența spațiului este nevoie de o metodă eficientă de comunicare. În cadrul spațiului virtual materialele tridimensionale sunt bazate pe OMG-CORBA, un model de arhitectură distribuită. Este introdusă și o metodă de creare a noilor obiecte și de modificare a proprietăților lor. În cele din urmă, după dezvoltarea și implementarea unui prototip de sistem se pot evalua diferitele aspecte care privesc metodele propuse.

Studiile recente asupra instruirii asistate de calculator sugerează că *învățarea interactivă*, în care cursanții interacționează activ cu materialul educațional și *învățarea colaborativă*, în care cursanții vorbesc cu ceilalți din grup, joacă un rol semnificativ în dobândirea de noi cunoștințe. În plus, oamenii sunt mai interesați de educația asistată de calculatoare, în special de *învățarea cooperativă asistată de calculator*. În cadrul acestui nou tip de învățare sunt folosite diferite tehnologii precum poșta electronică, știrile, conversațiile, conferințele video/audio și tablele albe. Odată cu recente salturi în tehnologia informației și comunicațiilor, mediile grafice tridimensionale

(3D-CG) pot fi aplicate în diferite domenii, inclusiv educație. Prin utilizarea materialelor educaționale prezentate folosind 3D-CG, cursanții pot înțelege mai bine construcțiile tridimensionale, pot studia dinamica (de exemplu mișcarea în spațiu) și pot observa obiectele din diferite unghiuri de vedere. Tehnologia 3D-CG permite atât construirea unei lumi virtuale (ce nu poate fi realizată în realitate) cât și descrierea de structuri complicate. Cercetările recente au indicat codificarea materialul educațional prin utilizarea limbajului de modelare a realității virtuale, prin care datele de modelat pot fi descărcate local de pe un server, permițând mai multor utilizatori accesul la aceeași lume virtuală. Totuși, utilizatorii nu se pot recunoaște între ei și nu pot colabora în manipularea obiectelor virtuale. Fiecare evoluează într-o copie proprie a lumii virtuale. S-a dezvoltat un mediu virtual în care utilizatorii pot percepe și reacționa la acțiunile altor utilizatori simultani și pot manipula obiectele cooperativ.

Mediul colaborativ de învățare folosind un spațiu virtual are următoarele facilități:

(1) *Spațiu virtual comun* (cameră). O școală obișnuită conține mai multe unități de spațiu, fiecare fiind desemnată ca o sală de clasă. Cursanții din interiorul unei săli lucrează cu același material educațional. Un cursant poate fi prezent într-o singură cameră la un moment dat, dar se poate mișca liber dintr-o cameră în alta.

(2) *Viziune și parcurgere tridimensională a materialelor educaționale*. Cursanții din ace-

eași cameră văd aceleași obiecte tridimensionale. Dacă o proprietate a unui obiect se schimbă, atunci toți văd schimbarea în același timp. Fiecare cursant se poate deplasa prin cameră și vedea obiectele din diferite unghiuri de vedere.

(3) *Ferestre de identificare*. Pentru a permite recunoașterea reciprocă a cursanților care lucrează în aceeași cameră, pe ecranele calculatoarelor apar „ferestre” care identifică locul fiecărui cursant în clasă și direcția în care privește.

(4) *Conversații în timp real*. Pentru comunicare cursanții pot utiliza dialogul text sau conferințele audio/video, ceea ce conduce la o învățare colaborativă eficientă.

(5) *Operarea colaborativă a materialelor educaționale*. Cursanții pot interacționa activ cu materialele virtuale. De exemplu pot muta un obiect, îl pot roti, îi pot schimba forma, culoarea sau îi pot adăuga o nouă parte. De asemenea, pot activa unul din comportamentele predefinite ale obiectelor (de exemplu pot forța un obiect să cadă, ca și cum ar fi sub influența gravitației în lumea reală). Fiecare cursant observă schimbările produse de numeroasele tipuri de comportamente în același timp.

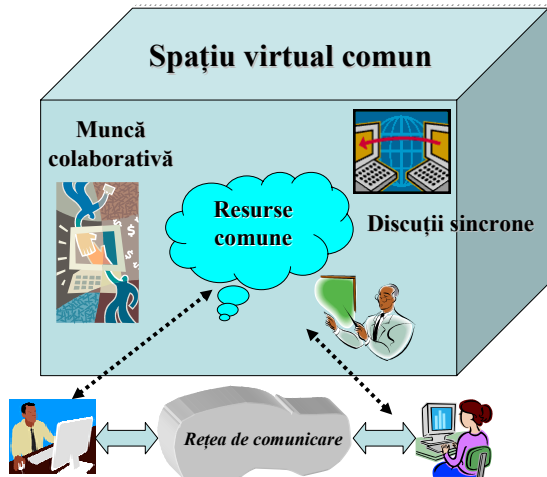


Fig. 1. Conceptul de sistem cooperativ interactiv

În scop educativ, unele sistemele pot fi folosite la studiul limbilor străine. Totuși, pentru situațiile în care interacțiunea strânsă cu obiectele este necesară – cazul experimentelor fizice și muncii creative – mecanismele oferite nu sunt suficiente. În aceste situații este

foarte important pentru sistem să gestioneze strict consistența stării dinamice a mediului virtual comun. Pentru ca schimbările în starea obiectelor să fie observate de toți cursanții, trebuie rezolvate următoarele aspecte:

- ✱ memorarea stării curente a spațiului comun și gestionarea schimbărilor dinamice;
- ✱ stabilirea celui mai bun model de obiecte pentru materialul educațional;
- ✱ alegerea celei mai bune interfețe pentru crearea/distrugerea obiectelor și modificarea proprietăților lor;
- ✱ realizarea comunicației între obiectele distribuite.

În sistemele educaționale, pentru cursanți este important să aibă posibilitatea de a crea noi obiecte și de a le schimba proprietățile. De aceea, sistemul trebuie să memoreze starea statică a materialelor educaționale și să gestioneze starea lor dinamică. Pentru a accelera procesul de desenare grafică, fiecare cursant trebuie să aibă propria copie a obiectelor dinamice grafice. Există două alternative pentru ca un cursant nou să obțină starea curentă a spațiului comun: de la server sau de la alt cursant. În cel de al doilea caz, dacă toți cursanții părăsesc camera, starea curentă dispare. Pentru a păstra totuși starea curentă, se adaugă sistemului un cursant special, numit *pseudo-cursant* (figura 2). Materialul educațional este creat și memorat în baza de date de materiale. Atunci când pseudo-cursantul este lansat, el încarcă materialul din baza de date și creează o copie locală. Atunci când cursanții A și B intră în cameră, fiecare primește o copie a copiei pseudo-cursantului și toate copiile sunt actualizate.

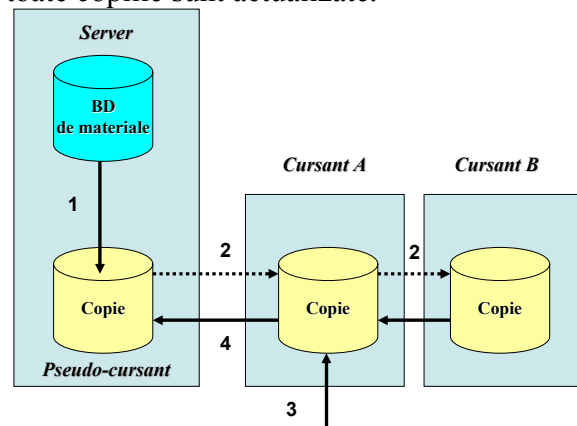


Fig. 2. Managementul datelor despre materiale

Un obiect distribuit poate fi definit prin arhitectura CORBA. Baza CORBA este obiectul broker de cereri (ORB) și limbajul de definire a interfeței (LDI). Fiecare interfață specifică este definită prin LDI și obiectele distribuite comunică între ele prin ORB. Astfel, fiecare obiect poate fi accesat prin ORB fără a ști locația sa geografică, folosind interfața

simplă definită prin LDI. Figura 3 prezintă structura fiecărui obiect educațional în spațiul virtual. Resursele din fiecare obiect educațional sunt gestionate de un singur obiect CORBA - „echipament”. El gestionează obiectele scenei, comportamentele și alte resurse. Un obiect de scenă are noduri grafice, un comportament și poate avea obiecte sunet.

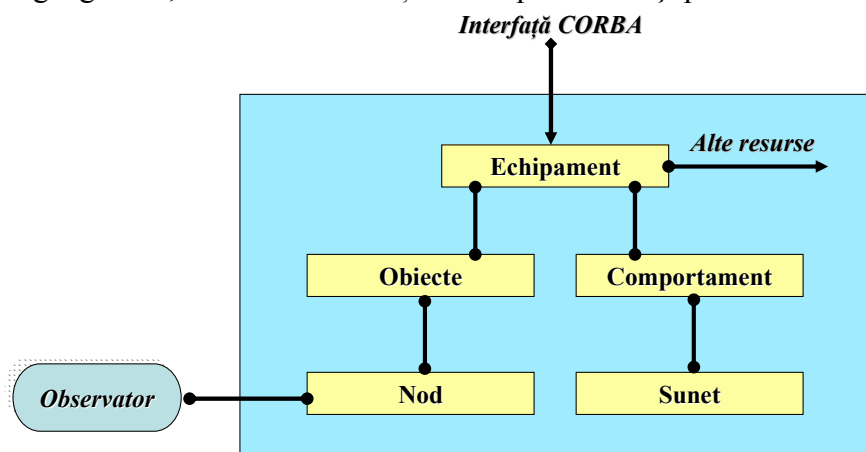


Fig. 3. Structura unui obiect educațional

Pentru ca mai mulți cursanți să împartă starea curentă a spațiului virtual, trebuie să schimbe mesaje între ei, ori de câte ori starea camerei se modifică. Mesajele sunt foarte dese, pentru a asigura sincronizarea corectă a stărilor memorate de fiecare cursant. Aceasta necesi-

tă rețele de viteză foarte mare, cu întâzieri reduse. Pentru o utilizare practică, mesajele trebuie clasificate și trebuie identificată cea mai bună metodă de comunicație pentru fiecare categorie (figura 4).

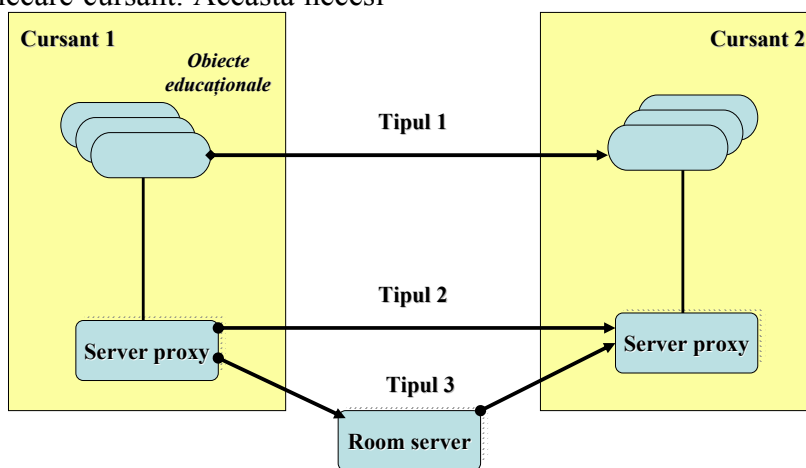


Fig. 4. Mecanismul de selectare

Tipul 1. Operații care nu afectează alți cursanți (de exemplu citirea stării unui obiect). Pentru aceste operații, cursantul trimite cereri direct obiectelor respective.

Tipul 2. Operații care cer o sincronizare strictă între toți cursanții (de exemplu aprinderea luminii). Pentru aceste operații, cursan-

mul trimite cererea întâi la serverul camerei, care îi verifică acceptabilitatea și starea în cameră, apoi o retrimite către destinație. Serverul camerei trimite și un mesaj către toți cursanții, anunțându-i să își actualizeze starea obiectelor.

Tipul 3. Operații care necesită un răspuns

rapid (de exemplu mișcarea unui obiect sau schimbarea unghiului de vedere propriu). Pentru astfel de operații, cursantul trimite o cerere către toți ceilalți cursanți prin *multicasting IP*, nu prin ORB. Cererile de acest tip sunt de obicei urmate de cereri de tipul 2. De exemplu, atunci când un obiect este mișcat, poziția sa este emisă în rețea. Atunci când mișcarea încetează, fiecare cursant actualizează starea obiectului său *proxy* prin cereri de tipul 2.

În sistemul prezentat, cele trei mecanisme de sincronizare sunt selectate automat de un obiect proxy al fiecărui cursant pentru a ascunde procesul de selecție și pentru a face interfața mai simplă. În mesajele de tipul 2, serverul camerei verifică acceptabilitatea mesajului trimis de fiecare cursant pentru a menține consistența stării camerei. Pentru a menține o consistență strictă, cea mai simplă cale ar fi folosirea unui mecanism de blocare care permite unui singur utilizator să opereze asupra unui obiect la un moment dat. Acest mecanism însă ar face dificilă pentru utilizatori percepția senzației de operare colaborativă asupra unui obiect, ceea ce contrazice chiar obiectivul de a oferi un mediu colaborativ.

Pentru ca mai mulți cursanți să manipuleze același obiect, trebuie trimise simultan mai multe mesaje care afectează acel obiect. Pentru mesajele care schimbă starea obiectului trebuie păstrată ordinea corectă. De exemplu, să presupunem că mai mulți utilizatori vor să schimbe culoarea unui semafor care are trei culori (roșu, galben, verde). Dacă ordinea mesajelor care ajung la fiecare cursant diferă, culoarea finală va fi alta la fiecare cursant. Pe de altă parte, mesajele care afectează obiecte separate sau proprietăți independente nu trebuie neapărat menținute într-o anumită ordine. De exemplu, dacă un mesaj care schimbă culoarea și unul care schimbă forma ajung în altă ordine, rezultatul final va fi același la toți cursanții. În plus, pentru unele mesaje, validitatea lor depinde de starea curentă a obiectului. Serverul trebuie să aibă un mecanism de respingere a cererilor care nu mai sunt valide. De exemplu, pentru un întrerupător electric, dacă lumina este stinsă, operația de aprindere

este validă, dar operația de stingere nu este. Pentru a verifica aceste restricții, o funcție locală de verificare la fiecare cursant nu este suficientă, ci este nevoie de un mecanism global de gestiune a stării întregii camere.

Serverul camerei trebuie să controleze secvența mesajelor astfel încât mesajele care determină schimbări ale stării să ajungă la fiecare cursant în aceeași ordine. În sistem, serverul serializează mesajele care ajung la el și apoi le trimite cursanților. Ordinea de sosire a mesajelor se poate schimba datorită traficului în rețea, dar numărul de secvență atașat permite cursantului să refacă ordinea corectă. Mecanismul de management al mesajelor din sistem a fost proiectat pentru a utiliza numere de secvență (figura 5). Serverul și fiecare cursant au câte o tabelă în care memorează ultimul număr de secvență S_n corespunzător fiecărui obiect. Într-o situație staționară serverul și cursanții au același număr de secvență corespunzător unui anumit obiect. Atunci când are loc un eveniment la un cursant, acesta trimite un mesaj cu numărul de secvență S_{n+1} către server, iar acesta îl retrimite cursanților. Serverul și cursanții actualizează numărul de secvență la S_{n+1} după procesarea mesajului. Serverul acceptă mesaje cu număr de secvență mai mare decât cel memorat în tabela sa. Atunci când doi cursanți trimit un mesaj referitor la același obiect aproape în același timp, cel care ajunge primul este acceptat, iar celălalt respins deoarece nu are un număr de secvență acceptabil. În acest fel starea camerei și a obiectelor este consistentă, chiar dacă mai mulți cursanți trimit cereri aleator.

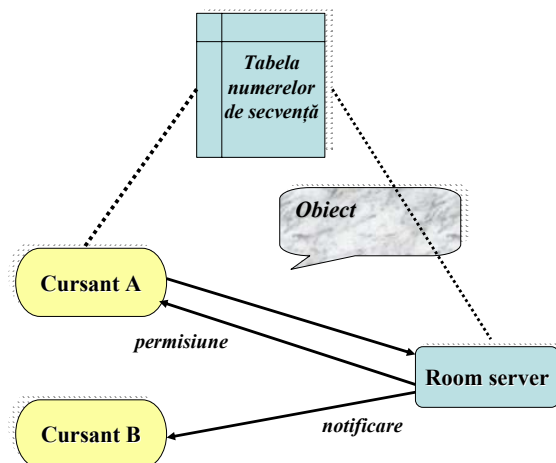


Fig. 5. Numărul de secvență al mesajului

S-a dezvoltat un sistem experimental care oferă un mediu colaborativ de învățare bazat pe considerațiile de mai sus. Sistemul a fost proiectat astfel încât mai mulți cursanți să poată lucra împreună într-un mediu de învățare tridimensional, să învețe într-o manieră interactivă și colaborativă. Principala facilitate este posibilitatea cursanților de a se deplasa în spațiul de studiu și de a-și schimba unghiul de vedere. În plus ei pot vizualiza ferestrele de identificare ale celorlalți cursanți și să discute cu ei în timp real. Pentru a asigura învățarea colaborativă, sistemul permite cursanților:

- ▶ să activeze comportamente atașate obiectelor;
- ▶ să creeze noi obiecte și să le modifice proprietățile;
- ▶ să atașeze noi comportamente obiectelor;
- ▶ să definească un obiect grup care include mai multe obiecte fii.

Așa cum se vede în figura 6, sistemul constă într-un server care gestionează materialul educațional din cameră, mai mulți cursanți și rețele care conectează serverul și cursanții. Serverul și cursanții rulează pe calculatoare personale cu SO Windows NT sau Windows 95. Software-ul de desenare folosit pentru grafica tridimensională este Open Inventor (Template Graphics Systems Inc.). Este folosită o placă de accelerare OpenGL pentru a asigura performanțe de desenare suficiente.

Pentru software-ul CORBA s-a folosit Orbix (IONA Inc.). Sistemul conține mai multe blocuri:

(1) **Blocul pentru server** care include:

✱ *Baza de date de materiale*: memorează materialele educaționale și datele geografice și poate fi accesată prin protocolul HTTP; fiecare cursant își descarcă materialul necesar de aici.

✱ *Server-ul camerei*: primește un mesaj de la cursant, verifică acceptabilitatea sa și îl trimite cursanților atunci când operația corespunzătoare se produce la aceștia.

✱ *Pseudo-cursantul*: memorează starea statică a camerei; un cursant care intră în cameră preia starea camerei de la pseudo-cursant, celelalte funcții fiind aproape la fel cu cele ale unui cursant obișnuit.

(2) **Blocul pentru cursant** care include:

✱ *Blocul de control al comunicației*: controlează trimiterea și recepționarea mesajelor și descărcarea materialelor din baza de date.

✱ *Blocul de control al interfeței utilizatorului*: controlează interfața pentru a schimba unghiurile de vedere și pentru a manipula obiectele.

✱ *Blocul de control grafic*: desenează spațiul tridimensional.

✱ *Blocul de control audio*: controlează efectele sonore și conversația în timp real între cursanți.

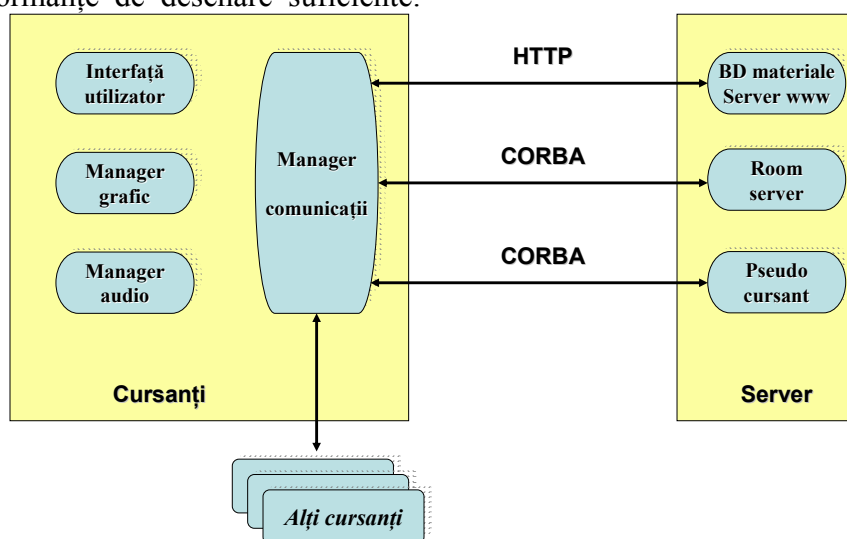


Fig. 6. Arhitectura sistemului

Învățământul deschis la distanță implică diversificarea rolurilor convenționale în peda-

re ale practicienilor, prin introducerea unor roluri noi. În sistemele tradiționale există un

personaj cheie: profesorul. Educația la distanță nu diminuează neapărat rolul acestui personaj, dar crește importanța altor persoane. Din perspectiva cursantului este importantă alegerea unui curs bun, iar consultanța în vederea alegerii o va primi de la un consilier, de la un administrator, dar sigur nu de la un profesor autor de curs sau de la un tutor. Similar, eficiența muncii unui tutor va fi direct proporțională cu calitatea muncii autorului de curs.

În concluzie, un program de educație la distanță eficace necesită un efort de echipă susținut și fiecare îmbogățire a experienței de învățare a cursantului are importanță. Cu această premisă sunt de evaluat mai multe aspecte datorită unui avantaj mai larg de semnale. Pe de altă parte, cursantul are libertatea să-și definească propriul parcurs de învățare și să folosească liber resursele oferite sau disponibile, nelimitându-se la materialul de curs, care poate fi folosit în egală măsură și importanță cu alte materiale.

Bibliografie

- ⌘ Arthur K., Booth K., Ware C., *Evaluating 3D task performance for fish tank virtual worlds*, ACM 11 (3), pp. 239-265, 1993
- ⌘ Hagsand O., *Interactive multiuser VEs in the DIVE system*, ACM MultiMedia, Vol.3, No.1, 1996
- ⌘ <http://www.omg.org/corba/corbaiiop.htm>
- ⌘ <http://www.worlds.net/>
- ⌘ Kato Y., Kawanobe A., Kakuta S., Hosoia K., Fukuhara Y., *Advanced collaborative educational environment using virtual shared space*, ED-MEDIA'96, pp.348-353, Boston, July, 1996
- ⌘ Lamport L., *Clocks and the ordering of events in a Distributed System*, ACM 21 (7), pp.558-565, 2000
- ⌘ Lind K., Hudak P., *Memory coherence in shared virtual memory systems*, ACM transaction on Computer Systems, pp.321-359, 2001.
- ⌘ Pesce M., *VRML: Browsing & Building Cyberspace*, New Riders Publishing, 1995
- ⌘ Rickel J., Johnson W., *Intelligent Tutoring in Virtual Reality: A preliminary Report*, AIED97, pp.294-301, 1997.
- ⌘ Snowdon D., West A., *AVIARY: Design issues for future large scale virtual environments*, Presence, Vol. 3, No. 4, 1994, pp. 288-308, MIT press.
- ⌘ Stice, J. E., *Using Kolb's learning cycle to improve student learning*, Engineering Education 77(5), 291-296.