

MADE – A Mobile Agent based system for Distance Evaluation

Vikram Jamwal and Sridhar Iyer

KR School of Information Technology, IIT Bombay, India

{vikram, sri}@it.iitb.ac.in

1. BACKGROUND AND MOTIVATION

The growth of Internet has led to new avenues for distance education. A crucial factor for the success of distance education is effective mechanisms for *distance evaluation* (DE). Existing Internet evaluation mechanisms, such as web based testing [1], rely principally on the client-server model. Such mechanisms usually do not scale well and also do not fully support features like: evaluation of subjective questions, delivery of dynamic content, and off-line examinations. These features are extremely desirable for distance evaluation and there is a need for alternate ways of designing such applications.

Mobile agents (MA) being autonomous and dynamic entities, offer many advantages over traditional design methodologies. We have designed and implemented, MADE, a mobile agent based system for distance evaluation¹. MADE aims to map closely to real world examination scenarios and addresses the full gamut of the examination process, viz., paper setting, distribution and testing, evaluation and result-Compilation. In this paper we describe MADE and show how MAs can be leveraged to effectively structure such large-scale applications.

2. MADE ARCHITECTURE

In MADE, we divide the examination process into three stages: (i) paper setting, (ii) distribution and testing, and (iii) evaluation and result compilation.

2.1 Paper Setting: This stage (Fig 2.1) involves setting of question papers with collaboration from several remotely located paper-setters. *Install Agents* are used to install the paper-setting application on each remote node (Steps 1, 2). Each paper-setter prepares a partial question paper (Step 3). Subsequently, the Paper Assembler node dispatches a *Fetch Agent* to collect these partial question papers (Step 4). Once all such question papers have been fetched, the Paper Assembler compiles them into one/more complete question papers (Step 5). One of these comprehensive question papers is sent to the examination centers, at the appropriate time (Step 6).

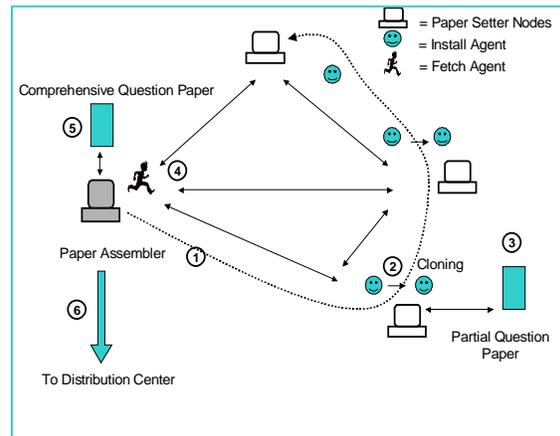


Fig 2.1 Paper Setting Scenario

2.2 Distribution and Testing: This stage (Fig 2.2) involves sending the question paper to different centers, distribution to students and collection of answer papers. *Courier Agents* are used to dispatch the question paper to the different examination centers (Steps 1a, 1b). The Distribution Server node at each center has a list of candidates enrolled for that center. The Distribution Server creates *Question Agents* (one per student), and dispatches them to each student node in that center (Steps 2, 3). Each *Question Agent* returns to the Distribution Server with the student's answers, after the designated duration or when the student finishes, whichever is

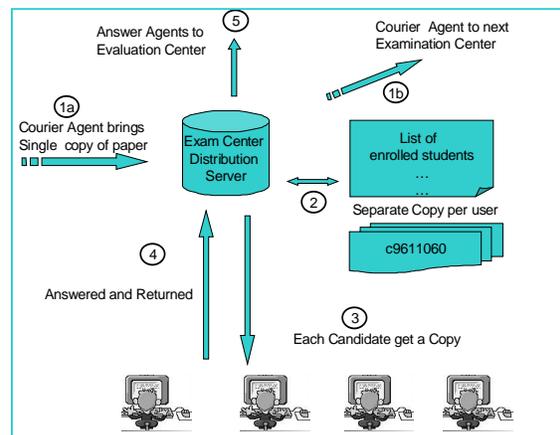


Fig 2.2 Distribution and Testing Scenario

¹ This work has been supported by a research grant from Ministry of Information Technology, Govt. of India.

earlier (Step 4). The Distribution Server creates *Answer Agents* (one per answer paper), and sends them to the evaluation center (Step 5).

2.3 Evaluation and Result Compilation: This stage (Fig 2.3) involves evaluation of the answer papers, compilation of the results and their publication. When an *Answer Agent* reaches the Evaluation Center (Step 1), it is supplied with an itinerary of the evaluators. The Answer Agent visits different evaluators until its evaluation is over (Steps 2, 3). It then moves to the Publishing Center and supplies the results (Step 4). Subsequently, the overall results are compiled and published (Step 5).

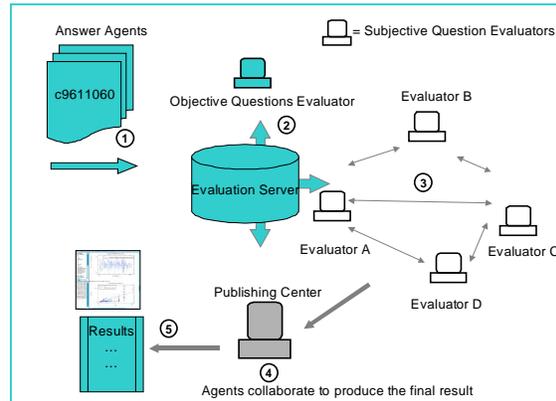


Fig 2.3 Evaluation and Result Compilation Scenario

We have implemented MADE using the Voyager ORB framework [2], and have successfully tested it in an experimental setup on a campus network [4]. Table 2.1 presents a summary of the different MAs used in MADE.

3. ADVANTAGES OF USING MOBILE AGENTS FOR DISTANCE EVALUATION

Many advantages of MAs have been cited in the literature [5]. The following are specific to our application:

- **Local Interactions:** MAs move to the node where the users (paper-setters, students, evaluators) are situated, thereby providing better response times.
- **Dynamic Content:** MAs encapsulate both logic and data, enabling them to carry dynamic content to the users. In MADE, a Question Agent can dynamically generate graphical displays of data, and carry specific tools, such as compilers, etc.
- **Support for push and pull:** MAs enable an application to exploit both push and pull models of content delivery. In MADE, a *need for push* exists in cases like distribution of question papers, workflow of answer papers among various evaluators, etc.
- **Force-Fetch:** *Partial computations at remote sites* may sometimes need to be force-fetched or returned. In MADE, a Question Agent at a student node can time-out after the designated examination duration and return to the server.
- **Hierarchical Management.** *Control by a central authority and remote management* are prime requirements in a distance evaluation application. In MADE, a remote node only needs to provide a *generic execution environment* and its managing node installs the relevant application components.

Other advantages include support for *subjective questions, content multicasting, and disconnected operations.*

Table 2.1 Mobile Agents used in MADE

	MA Type	No of Hops	Life-Duration	No of Instances	Itinerary
1	Install Agent	1 to # paper-setters	few months (paper setting duratn)	1 to # setters (10s)	Static: supplied by paper assembler
2	Fetch Agent	min. #paper-setters	few seconds – days	1 to # setters (10s)	Static initially, supplied by paper assembler; dynamic subsequently
3	Courier Agent	1 to # exam centers	few minutes	1 to #centers (100s)	Static: supplied by paper assembler
4	Question Agent	2 (to – fro student)	1 – 4 hrs (duration of exam)	# students (1000s)	Static: supplied by distribn server
5	Answer Agent	# evaluators	1 hr - few months (duration of evaln)	# students (1000s)	Static initially, supplied by evaln server; dynamic subsequently.

4. DISCUSSION AND CONCLUSIONS

A typical distance evaluation scenario is characterized by: large geographical distances, disconnected operations, dynamically interchangeable client-server roles and large number of interacting nodes. By using MAs as the principal design paradigm in MADE, we could overcome many of the limitations of existing systems. MADE caters to the full gamut of distance evaluation viz. paper-setting, distribution and testing, and evaluation by means of various types of MAs.

From our experience, we feel that MAs can be used effectively for structuring such large-scale distributed applications. The gains would be in terms of: scalability, flexibility, dynamic extendibility and independence from the network variations. Our ongoing and future work concerns: (i) finding means for effective handling of autonomy, (ii) improving the overall system reliability, and (iii) addressing additional security concerns that arise when we move from closed trusted systems to open systems.

5. REFERENCES

1. Chien Chou. Constructing a Computer-Assisted Testing and Evaluation System on the World Wide Web-The CATES Experience, in *IEEE Transactions on Education*, Vol. 43, No 3, Pages 266-272, August 2000
2. G. Glass, "ObjectSpace Voyager Core Package Technical Overview", *Mobility: process, computers and agents*, Addison-Wesley, Feb. 1999
3. Vikram Jamwal and Sridhar Iyer. Mobile Agents for Effective Structuring of Large Scale Distributed Applications, *Workshop on Software Engineering and Mobility, ICSE 2001*, Toronto, Canada, May 2001
4. Vikram Jamwal. MADE: Mobile Agent System for Distance Evaluation, *M.Tech. Thesis*, KR School of Information Technology, IIT Bombay, India, Jan. 2001
5. Danny B. Lange. Mobile Objects and Mobile Agents: The Future of Distributed Computing, In *Proceedings of The European Conference on Object-Oriented Programming '98*, 1998