ArchES: A Rule-based System for Hypothetical Social Scenarios in Archaeology

GEORGIOS K. GIANNIKIS[1], ATHOS ANTONIADES[2], DIONYSIOS POLITIS[1]
[1]Department of Informatics
Aristotle University of Thessaloniki
University Campus, Thessaloniki, GR-541 24
GREECE
[2]Department of Computer Science
University of Cyprus
University Campus, Nicosia
CYPRUS
http://www.csd.auth.gr
http://www.cs.ucy.ac.cy/

Abstract: - In this paper we present ArchES, a novel rule-based system implemented as a web tool for formulating and testing hypothetical archaeological social scenarios. Scenarios are organized by archaeologists and are based on rules and facts that derive from findings in excavation areas. ArchES analyses different variables that encode social factors affecting areas and findings, in the past as well as the present. The presented tool was implemented using Java web technologies and Jess as the rule engine. ArchES is part of the SeeArchWeb project that aims to develop a new instructional approach for the domain of Archaeology.

Key-Words: Rule-based System, Expert System, Excavations, Archaeology, Social Scenarios, Jess

1 Introduction

Nowadays, the evolvement of the Internet provides a novel platform for sciences and scientists to offer and request additional electronic services (e-services). Information technology applications in archaeology are rapidly expanding and new services are emerging, based on the development of new intelligent software. Among the most popular and valuable services in archaeology are the web-based expert systems [1, 2].

Archaeology is a problem-oriented discipline, which tries to solve WHY? questions [3]. Answers are unobservable social causes, such as actions and processes, related to observable elements, such as items found in excavation areas. Problem solving can be conceptualized as a form of learning, because it can be defined as the acquisition of knowledge (decision rules) that derives from existing data (facts) and is inserted in an intelligent information system. The prerequisite of building an information system is the existence of a formal and systematic knowledge relative to a very narrow subject that will be encapsulated in an expert system [4]. In this paper we present a rule-based system implemented as a service that offers to experts and non-experts the ability to formulate, organize, initialize and test hypothetical social scenarios, based on items that were gathered and facts that were concluded from excavation areas. The presented tool is called ArchES and is part of the Social Modeller Module of the SeeArchWeb project.

Expert systems are intelligent programs provided with high-quality specific knowledge about a specific area [5, 6]. The process of building an expert system typically involves a special form of interaction between the expert-system builder (Knowledge Engineer) and the expert in the specific area (Domain Expert). The Knowledge Engineer extracts from the Domain Expert strategies and rules for solving the problem. Extracting information (knowledge) is usually in the form of facts and rules. For example [7]:

Facts:
Site x has pottery.
Pottery is of type A.
Rules:
IF site x has pottery of type A THEN site x chronology is 5th century.

Facts and rules may not always be true/false with absolute certainty. A degree of certainty/uncertainty is commonly used, to express the validity of a fact or the accuracy of the rule. The collection of the domain knowledge is called knowledge base while the problem-solving tool that based on knowledge emulates human capabilities to arrive at a conclusion by reasoning is called inference engine.

The rest of the paper is organized as follows: in section 2 we describe the SeeArchWeb project and refer to the need for a designed tool such as ArchES. In section 3 we present an example scenario which will be analysed and represented in the form of the ArchES supported rules. In section 4 we present in details the ArchES web-based tool referring to its architecture, rule-model, and technologies needed for implementation and example execution by applying the above mentioned scenario. In section 5 we present related research efforts and applications. In section 6 we present a short discussion, conclusions and future work respectively.

2 The Social Modeller Tool

SeeArchWeb is a MINERVA project that aims to develop and present a new instructional approach for the subject domain of Archaeology based on networked technologies. The project emphasizes on a pilot study for the Prehistoric Archaeology of Southeastern Europe. In order to accomplish this aim, the SEEArchWeb infrastructure has 4 parts:

- The Web Course Module which is the basic learning resource for use by students, teachers and lecturers.
- The Social Modeller Module that provides the users (learners, social scientists and archaeologists) with a new instrument for analysis, comparison and testing of hypothetical social scenarios.
- The Excavation Cataloguer Module which is a standardized digital database used as a storage of archaeological excavation data.
- The Educational and Community Resources Module which presents to the general public, resources about the archaeology of Southeastern Europe through the development of a current fund of archaeological community related information.

ArchES, the Social Modeller Tool, is a novel instrument that formulates and tests hypothetical social scenarios provided by the expert Archaeologist. ArchES analyses different variables as social factors in Southeastern Europe in the past as well as the present. The social modeller use knowledge management techniques to analyse large amounts of information available through the Excavation Cataloguer Module database. The detailed defining of the social values provides a better understanding of past processes and offers benefits such as more inclusive understanding of the region today.

In terms of computer science, the Social Modeller Module includes an intelligent interface capable of accepting questions for social scenarios of the form: what would have happened if …? Based on an expert system that takes into account the knowledge database deposited at the project’s central server, ArchES deploys a rule-based decision taking mechanism for social scenarios to supply the answers to questions.

Modeling scenarios could be related to the:

- Representation of accumulation of wealth.
- Representation of social inequality and equality.
- Representation of exchange and trade mechanisms.
- Representation of settlement patterns (regional and local).
- Role of technology in society.

Social Modeller Module was designed for European graduate students in European archaeology and related disciplines, professional researchers and other social scientists.

3 An Example Scenario

In this section we present a general purpose example scenario to use in the ArchES system. Our scenario is motivated by the story of the lost continent, Atlantis. The story of Atlantis begins quite literally with two of Plato's dialogues, Timaeus and Critias [8]. These accounts are the only known written records which refer specifically to a lost civilization called Atlantis.

According to Plato, over 11,000 years ago there existed an island nation - larger than Libya and Asia combined - located in the middle of the Atlantic ocean – outside the 'Pillars of Heracles' - populated by a noble and powerful race. The people of this land possessed great wealth thanks to the natural resources found throughout their island. The island was a center for trade and commerce. The rulers of this land held sway over the people and land of their own island and well into Europe and Africa. This was the island of Atlantis.

To facilitate travel and trade, a water canal was cut through of the rings of land and water running south for 5.5 miles to the sea.

The city of Atlantis sat just outside the outer ring of water and spread across the plain covering a circle of 11 miles. This was a densely populated area where the majority of the population lived.

Beyond the city laid a fertile plain 330 miles long and 110 miles wide surrounded by another canal used to collect water from the rivers and streams of the mountains. The climate was such that two harvests were possible each year. One in the winter fed by the rains and one in the summer fed by irrigation from the canal.

Surrounding the plain to the north were mountains which soared to the skies. Villages, lakes, rivers, and meadows dotted the mountains.

Besides the harvests, the island provided all kinds of herbs, fruits, and nuts. An abundance of animals, including elephants, roamed the island.

For generations the Atlanteans lived simple, virtuous lives. But slowly they began to change. Greed and power began to corrupt them. When Zeus saw the immorality of the Atlanteans he gathered the other gods to determine a suitable punishment.

Soon, in one violent surge it was gone. The island of Atlantis, its people, and its memory were swallowed by the sea.

3.1 Derived Rules

Based on this section of Plato's scripts decision rules can be derived. In Table I we report a non exhausted list of possible rules.

Table I. *Derived rules for Atlantis Scenario*

ID	Rule
Rule1	If site's name is Atlantis then site is Atlantis.
Rule2	If site's age is 11,000 years then site is Atlantis.
Rule3	If site is on island and island is in Atlantic Ocean then site is Atlantis.
Rule4	If site is on plain and plain is enormous and plain is fertile then site is Atlantis.
Rule5	If site was destroyed and site was swallowed by the sea then site is Atlantis.
Rule6	If site was surrounded by water channels then site is Atlantis.
Rule7	If site had army or site was powerful then site is Atlantis.
Rule8	If site was trade centre or site was rich then site is Atlantis.

Furthermore, support factors should be invoked for the listed rules in order to express the weight of each one of the left hand side facts (premises) in the process of concluding the right hand side fact (consequent).

4 ArchES

In this section we present the ArchES system architecture and its possition and relation with the

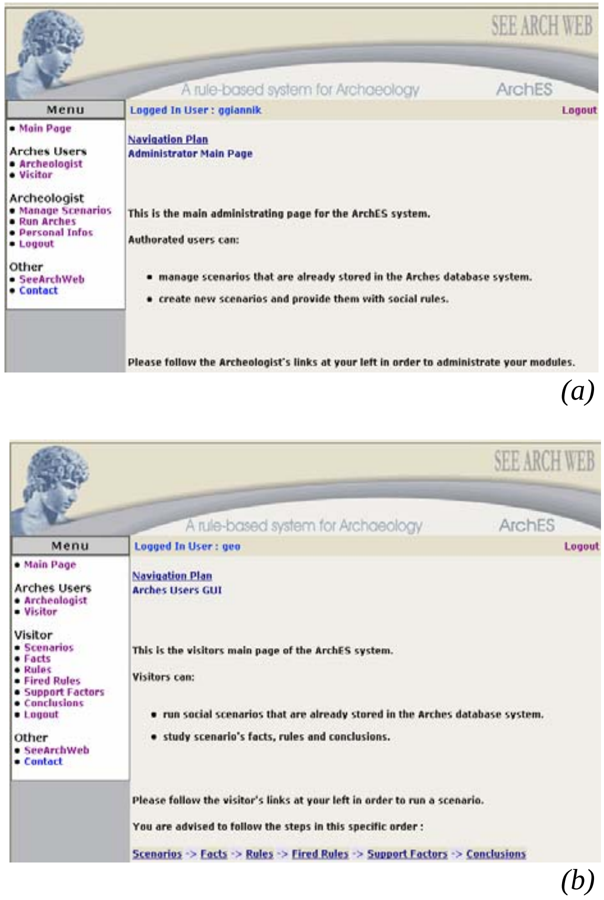


Fig. 1. ArchES users web tools,
(a) ArchaeologistUI, (b) VisitorUI

overall SeeArchWeb project. We explain the form of the rules that ArchES supports and finally we present in details the execution workflow for the Atlantis scenario.

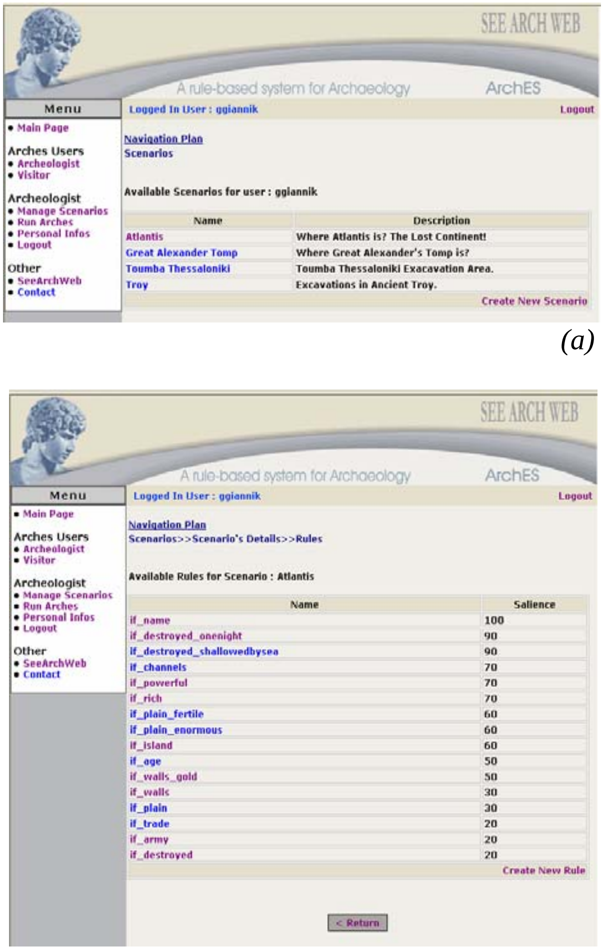
4.1 System Architecture

As already mentioned ArchES tool is part of the Social Modeller Module of the SeeArchWeb project. ArchES is a web tool that works in cooperation with the Excavation Cataloguer Module digital database. The tool consists of two independent intefaces (see Fig. 1), one for the domain expert – ArchaeologistUI – and one for the learner visitor – VisitorUI.

Domain expert that logs in the system has the ability to:

- create new social scenarios (see Fig. 2(a)), and
- manage formerly stored scenarios that he/she owns (see Fig. 2(b)).

Scenarios are organised by defining rules and the way they activate and act. As a result, a deduction system is fomulated. Rules are stored in the *rule base*, that is the system's relational database. On the other hand, system behavior is defined in the *control*



(a)

(b)

Fig. 2. ArchES Administrator tools,
(a) Available Scenarios, (b) Scenario's Rules

module, that can be expressed either as a tree-like structure or by providing a salience factor related to each rule.

ArchES visitor that enters the system, after a welcoming message, has the ability to:

- choose a social scenario to run,
- view all initial facts that are relative to the chosen scenario,
- view all rules that where initial entered in the system's inference engine,
- view just the fired rules that drive the system to a conclusion jointed with their computed support factor,
- view the final conclusion and its certainty factor.
- modify scenario's values (support factors) and reactivate the inference engine.

The intelligent component of the ArchES is implemented in Jess (Java Expert System Shell) [9]. Jess is a rule engine and scripting environment written entirely in Sun's Java language by Ernest Friedman-Hill at Sandia National Laboratories. It supports the development of rule-based systems which can be tightly coupled to code written in the powerful and portable Java language. Jess is a fast and powerful rule engine so it is an ideal tool for adding rules technologies to web-based software systems.

Jess was originally inspired by the CLIPS expert system shell, which has a successful history as a tool for expert system development. Later, Jess grows into a complete, distinct, dynamic environment of its own. By using Jess, we build a Java web tool that has the capacity to "reason", using knowledge that was supplied in the form of declarative rules by experts. Jess supports both forward and backward reasoning and has the full features of a programming language. The syntax of Jess rule language is similar to Lisp, that is a simple language, easy to learn and well-suited to both defining rules and procedural programming. Jess uses a special algorithm called *Rete* to match the rules to the facts. Rete makes Jess much faster than a simple set of cascading *if.. then* statements in a loop.

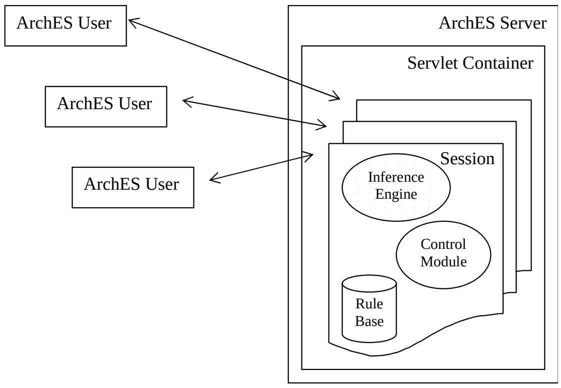


Fig. 3. Embedding Jess in ArchES

There are different ways to embed Jess in a web application [9]. Usually web-based software is divided into two broad categories: *client-side* and *server-side* applications. By choosing the first, Jess can be deployed as an applet. By choosing the second, Jess can be embedded using technologies as Servlets, Java Server Pages (JSP) and Web-Service (WS). Usually, a combination of them is preferable. In our application a combination of Java Servlets and JSPs were used in the way described in figure 3. Particularly, Java Servlets were used to deploy *Jess Engine* and to query the engine's state and JSPs are an intermediate factor in order to present the engine's state.

Rules are stored in a relational database; they are gathered by the expert who formulates the

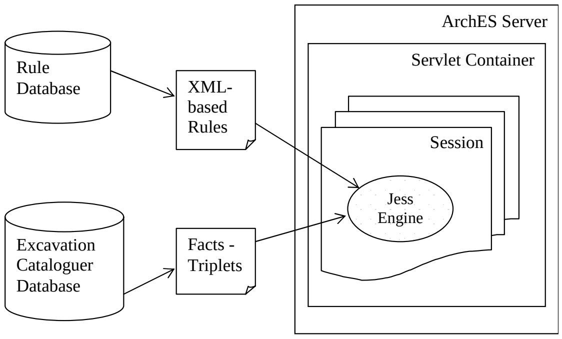


Fig. 4. ArchES Storage Architecture

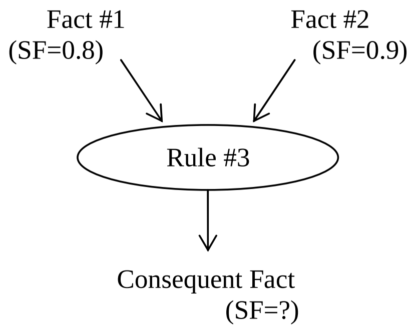


Fig. 5. ArchES Rule Model

applicable scenarios. During initialization of the Jess Engine, rules are extracted from database, transformed into an intermediate form (XML) and imported into the engine using the XML-based rule representation that Jess supports [9]. Besides rules, facts should also be imported in Jess Engine. Initial facts come from Excavation Cataloguer Module relational database.

A special way, inspired by RDF (Resource Description Framework) documents, is used to represent facts, both for initial facts and premises facts (lhs) in rules. A fact is represented by a triplet that contains a subject, a predicate and a value, they are given in a line-based plain text format.

(<*subject*>,<*predicate*>,<*object*>).

The way that data (rules and facts) are stored and transformed is shown in figure 4.

4.2 Rule Model

Rules provide a formal way of representing recommendations, directives, or strategies. They are appropriate when the domain knowledge results from empirical associations developed through years of experience solving problems in a specific area [5]. Rules are expressed as *if.. then* statements as shown in section 1. Rules of this manner are supported by ArchES.

Moreover, rules are enhanced with a support factor (SF). Support factor is a number that measures the certainty or confidence one has that a fact or rule is valid. For our tool, support factors (or else certainty factors) can take a value in the range of [0-1].

For instance, Rule 3 from table I could be enhanced with support factors and represented as shown in figure 5.

<u>Rule 3:</u>
IF site is on island with certainty 0.7
AND
island is in Atlantic Ocean with certainty 0.8
THEN site is Atlantis with certainty ?.

Besides rule's premises facts, initial facts are also enhanced with the support factor. For example, consider that among other initial facts are the below two:

<u>Initial Fact 1:</u>
Excavation site is on island with certainty 0.8.
<u>Initial Fact 2:</u>
Island is in Atlantic Ocean with certainty 0.9.

According to the presented model, a rule should only fire (activate) if:

- the *if* portion of the rule is satisfied by initial or intermediate (facts that were inserted in the inference during execution) facts, and
- facts' support factors are equal or greater than the support factor of the rule's premises facts.

When a rule fires, the action specified by the then portion performs and certainty factor is computed as the product of initial (or intermediate) fact's support factors.

In addition to this, each rule associates with a salience value that defines rule's priority in the control module. Variations in priority help us to construct a tree-like structure scenario (see Fig. 6)

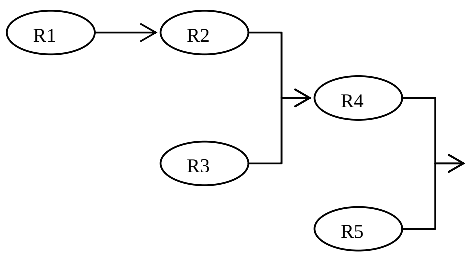


Fig. 6. Tree-like Structure Scenario

4.3 Scenario Execution

As soon as the insertion of social rules from the domain expert completes, the ArchES rule system is ready for execution.

Let us concern, that Atlantis Scenario rules shown in Table II are stored in the ArchES rule base. Rules are enhanced with support factors and a salience value (rules with greater salience have priority in firing). For example, rule 1, means that :

Rule 1:

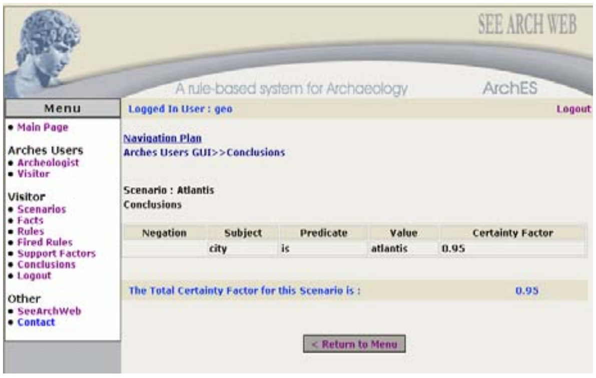
IF a fact supports, with absolute confidence, that site's name is Atlantis
THEN the site should be Atlantis.

Table II. *Inserted rules for Atlantis Scenario*

ID	Sal	Rule
Rule1	100	If site's name is Atlantis (*SF=1*) then site is Atlantis (*SF=?*).
Rule2	60	If site's age is 11,000 years (*SF=0.8*) then site is Atlantis (*SF=?*).
Rule3	60	If site is on island (*SF=0.7*) and island is in Atlantic Ocean (*SF=0.8*) then site is Atlantis (*SF=?*).
Rule4	70	If site is on plain (*SF=0.6*) and plain is enormous (*SF=0.8*) then site is Atlantis (*SF=?*).
Rule5	70	If site is on plain (*SF= 0.6*) and plain is fertile (*SF=0.8*) then site is Atlantis (*SF=?*).
Rule6	90	If site was destroyed (*SF=0.9*) and site was swallowed by the sea (*SF=0.9*) then site is Atlantis (*SF=?*).
Rule7	90	If site was surrounded by water channels (*SF=0.95*) then site is Atlantis (*SF=?*).
Rule8	50	If site had army (*SF=0.7*) or site was powerful (*SF=0.8*) then site is Atlantis (*SF=?*).
Rule9	50	If site was trade centre (*SF=0.7*) or site was rich (*SF=0.8*) then site is Atlantis (*SF=?*).

When the system executes, stored rules and facts are loaded in Jess engine, an inference rule chain is produced, rules, which their premises facts are satisfied, fires and a final consequent fact with its certainty factor conclude (see Fig 7). Apart from this, ArchES user has the ability to check initial facts and rules, fired rules, conclusion facts and their computed factors. Scenarios can be executed iteratively, while modifying their values will drive them to an alternative definition of hypothetical scenarios and different conclusions.

(a)



(b)

Fig.7. ArchES User tools,
(a) Fired Rules, (b) Consequent Fact

5 Related Work

During the last fifteen years a rapid growth of research activity related to rule-based systems in archaeology has occurred. Selected former research and implementation efforts that exceed classification and typology purposes and inquire functionality and social criteria are briefly presented in this section. Patel and Stutt in 1988 [10] developed the KIVA system, which by given a description of the spatial context of an area interprets the function of the areas and the activities that took place in an archaeological deposit. Frankfurt in 1992 [11] with the PALAMEDE system was able to analyse some effects of social processes on the archaeological record, notably, the concepts of wealth and hierarchy. Roger Grace in 1996 [12] produced a tool for classifying the shape and the technology of prehistoric tools, and to explain their functionality.

6 Conclusions and Future Work

From the literature review, it is clear that expert systems are popular in the archaeological domain. Computer can execute interpretation procedures that an archaeologist considers excellent. In other words, machine intelligence reproduces operations designed by a human. Combining technologies such as inteligent software (Expert Systems and Agents) with vision techniques (Machine Vision) and with Geographic Information Systems (GIS) and by having in mind the advantage that the Internet offers, we can implement powerful, operative and reliable software.

In this paper ArchES was presented, a novel rule-based system implemented as a web tool for formulating and testing hypothetical archaeological social scenarios. Scenarios are organized by archaeologists and are based on rules and facts that derive from findings in excavation areas. The advantages that ArchES offers, compared with other similar systems, are its multi-purpose – the system was design to provide to different users, learners, social scientists and archaeologists, with the ability to analyse, compare and test hypothetical social scenarios, its scalability – as the number of findings grows up, archaeologists can add more social scenarios or alter and extend already stored rules and its availability as a web-based tool.

Our aim for future work is the extension of this software to have the capability to learn, accumulate knowledge and combine observations with information derived from the project's database and by using adaptive pattern matching techniques to apply this knowledge to new situations and scenarios. Moreover a software agent module capable of accepting questions, in the form of human speaking, for social scenarios will be of utmost importance.

7 Acknowledgements

Jess, the Rule Engine for the Java Platform is Copyright (C) 2004 by the Sandia Corporation. Jess software, owned by Sandia National Laboratories, was made available upon request under the Academic R&D license.

References:

[1] Gardin, J.C., Artificial Intelligence and Expert Systems: Case Studies in the Knowledge Domain of Archaeology, Prentice Hall (Sd), 1989.

[2] Wilcock, J.D., A critique of expert systems, and their past and present use in archaeology, *In Ennals, R. & J.-C. Gardin (eds), Interpretation in the humanities: perspectives from artificial intelligence*, LIR Rep. 71, British Library, 1990, pp. 130-142.

[3] Barcelo, J.A, Notes about the use of Neural Networks for Archaeological Modelling, 2004, http://seneca.uab.es/prehistoria/Barcelo/

[4] Huggett, J.,Expert systems in archaeology, *in J. Richards and M. Cooper (eds.) Current Issues in Archaeological Computing*, British Archaeological Reports, Oxford, 1985, pp. 123-142.

[5] Waterman, D.A., A Guide to Expert Systems, Addison-Wesley Publishing Company, 1986.

[6] Liebowitz, J., Handbook of Applied Expert Systems. Boca Raton, FL: CRC Press, 1997.

[7] Barcelo, J.A., Expert systems as Cognitive Emulation. An Archaeological viewpoint, 2001, http://seneca.uab.es/prehistoria/Barcelo/

[8] Atlantis - Fact, Fiction or Exaggeration?, 1999, http://www.activemind.com

[9] Friedman-Hill, E., Jess in Action, Rule-Based Systems in Java, Manning Publishing Company, 2003.

[10] Patel, J., Stutt, A., KIVA: an archaeological interpreter Human Cognition, *Research Laboratory Technical Report, No. 35*. Milton Keynes: The Open University, 1988.

[11] Frankfurt, H.P., Palamede - application of expert systems to the archaeology of prehistoric urban civilizations, *Computer Applications and Quantitative Methods in Archaeology*, 1992.

[12] Grace, R., Expert Systems for Lithic Analysis, 1996. http://www.hf.uio.no/iakk/roger/lithic/expsys.html