



HAL
open science

METAN: a learning system for training of engineers to material observation and analysis

Jean-Luc Wybo, Evelyne Darque-Ceretti, Pierre Marcenac, Pierre Crescenzo

► **To cite this version:**

Jean-Luc Wybo, Evelyne Darque-Ceretti, Pierre Marcenac, Pierre Crescenzo. METAN: a learning system for training of engineers to material observation and analysis. CALISCE 1994 (International Conference on Computer Aided Learning and Instruction in Science and Engineering), Aug 1994, Paris, France. pp.401-407. hal-01290046

HAL Id: hal-01290046

<https://hal.science/hal-01290046>

Submitted on 17 Mar 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

METAN: a learning system for training of engineers to material observation and analysis

Jean Luc Wybo¹, Evelyne Darque-Ceretti¹, Pierre Marcenac², Pierre Crescenzo¹

¹CEMEF, Ecole des Mines de Paris, BP 207

06904 Sophia Antipolis, France. E-mail: wybo@cemef.cma.fr

²IREMIA - Université de la Réunion, 15, Av. R. Cassin, BP 7151

97715 St Denis Messag., France. E-mail: marcenac@univ-reunion.fr

Abstract

This paper proposes an implementation of a learning system for training of engineers to material observation and analysis. First, we present the objectives of this application. Then, we propose a decomposition of the system in modules and we describe the knowledge representation and the software architecture. Last, we illustrate the architecture by giving an example of the tutoring session.

keywords: tutoring system, material science, knowledge base, student model

1. Introduction

We present a learning system: METAN, using Artificial Intelligence techniques. The aim of this system is to propose an active teaching process in the field of analysis and observation methods, applied to material processing.

The engineer faced with choice or performance testing of a material, is often brought to sample observation and analysis. In order to succeed in that work, he must know those methods, their execution field, their feasibility and their limits.

Among the field of transformation processes, the materials are metals, polymers and ceramics. The choice of a material is conditioned by the whole of the required properties (e.g. wear, resistance, adhesion of paintings, possibility of recycling, esthetic outlook...). The definition and the evaluation of the properties need a set of observations and measurements, for instance roughness, surface elemental composition.

In order to bring up the students to a rational approach of the choice of methods adapted to a given problem, we have introduced in the METAN system an expert system and a knowledge base able to build an argument leading to an optimum solution. This argument is the driving path for evaluation of the student answering.

A classical off-chair teaching, usually short, does not always bring to the students a necessary practice and interactivity. On the other hand the students, whatever they are graduating or already working, have a more or less complete knowledge of methods and physical basis needed to solve a problem. A computer-aided teaching system must be able to adapt its pedagogy to the specific needs of each student, by providing the detail level that he requires and by evaluating each step of his reasoning. This is the goal that we want to reach in this study.

2. System Overview

The logical structure of the METAN system is composed of three parts:

- A *training module*, containing a series of chapters dedicated to general knowledge, materials under study, methods and solving scheme. This module uses an hypertext interface, allowing the adaptation of the information presented to the student: most advanced students follow the main

course screens, while others may call, using hypertext links, variable level explanations of topics presented in the main course. This flexibility corresponds to the use of a *differentiated pedagogy* [Astolfi 85].

- A *reasoning module*, using an expert system. Within training systems, there are two kinds of solvers: "blackboxes" [Anderson 88], like SOPHIE [Brown 75] for fault diagnosis and those using a "cognitive model", like SOCRATE [Moustafiades 89].

We have chosen this second approach to give to the system the ability to build the same reasoning that an "ideal student", as in NAIADÉ [Joab 90]. This module starts from the text of the problem given as exercise to realise a three steps solving method. The first step consists in deducing from the question all pertinent elements, i.e. those which may help to solve the problem. The second step is dedicated to choose the sample properties related to the problem and to determine the values which solve it. The last step of reasoning is the determination of needed transformations to apply in order to modify these properties, and then, to choose and schedule the best set of methods to realise these transformations.

- An *evaluation module*, in charge of collecting student answers for each step and to compare them with correct answers given by the reasoning module. Knowledge evaluation of student has motivated many studies, like GUIDON [Clancey 82], which assesses for each rule a belief level that the student knows it, or SIERRA [VanLehn 88], which uses a cognitive diagnosis trying to find a rational origin to student's errors. We have chosen a simpler first approach, in which the elements deduced by the reasoning module are presented to the student in combination with the related pieces of course (see section 4). For the first step, the student controls if the elements that he has deduced are true, if they are pertinent for the problem and if he has found all of them. For the second step, he verifies if he has chosen the good attributes and their correct values (is the problem well posed?). For the third step, did he choose the good transformations, actions and methods in the correct order? (i.e. is the problem well solved?). This module, considered as a tutor, proposes at each step a summary of correct answers, which give access to the related chapters of the course, using hypertext links as in the training module.

3. Knowledge representation

The METAN system uses two sets of knowledge: basic knowledge represented in the training module and reasoning method represented in the reasoning module. Basic knowledge is essentially related to observation and analysis methods, but also other topics that the student must know: properties of materials, needed to choose the appropriate methods and a set of general principles about physics.

We distinguish two main classes of reasoning methods: those relevant to sample observation and those relevant to their analysis. Each method (mostly analysis methods) produces data which need to be processed and it can be necessary to prepare the sample before observing or analysing it. To take into account these methods, we have introduced two more classes: preparation methods and data processing methods.

To represent knowledge, the METAN system uses an object oriented approach [Cox 86][Meyer 88] and offers four main classes: samples, methods, actions and transformations. These classes are linked together by relations: a method acts one or several actions, an action is composed of one or several transformations, a transformation modifies one attribute of the sample.

The knowledge itself can be found in many books which present basics about materials, for instance [Askeland 89]. For analysis methods, each kind corresponds to specific literature, for instance [Briggs 83] for Auger and X-ray spectroscopy.

The problem is asked to the student in this way: *do what on whom ? Do* is a verb, for instance: observe, analyse, ... *What* is the object of interest, for instance: stresses, roughness, ... *Whom* is the sample, for instance: brass (70/30), glass, ...

Starting from these elements, in the first step, the reasoning module applies knowledge to deduce all what is possible.

From these deductions, during step two, it identifies the conditions which solve the problem, as a set of sample properties to modify.

The last step of the reasoning consists in finding the corresponding transformations, then the actions including these transformations and finally the methods. We have introduced the concept of needs to represent use constraints. A method or an action is often associated to conditions to apply it, for instance observation methods using electron beams can be applied if the surface is conductive. The choice of a method may involve the search of a complementary method which function is to put the sample properties in a state permitting the use of the first method. In this example, if the sample is conductive (a metal for instance), then the electron microscope method can be chosen. If the sample is not conductive, there are methods which action is to deposit a thin layer of metal or carbon on samples, making them conductive enough. A possible solution will be to associate a preparation method (metallization) and an observation method (scanning electron microscope).

When there are several possible methods to solve the problem, we have to build an order to propose the best one. To define this order, we associate to methods several criteria and we use knowledge to assess the level of interest of each method.

4. General description of the METAN application

As in EDDI [Marcenac 90], the software structure of METAN will be based on three distinct units which will constitute the application (see Figure 1). These three units are called: the **expert**, the **tutor** and the **user interface**. By reference to the logical structure of the METAN system presented in section 2, the *training module* and the *evaluation module* will be managed by the **tutor**, while the **expert** will manage the *reasoning module*.

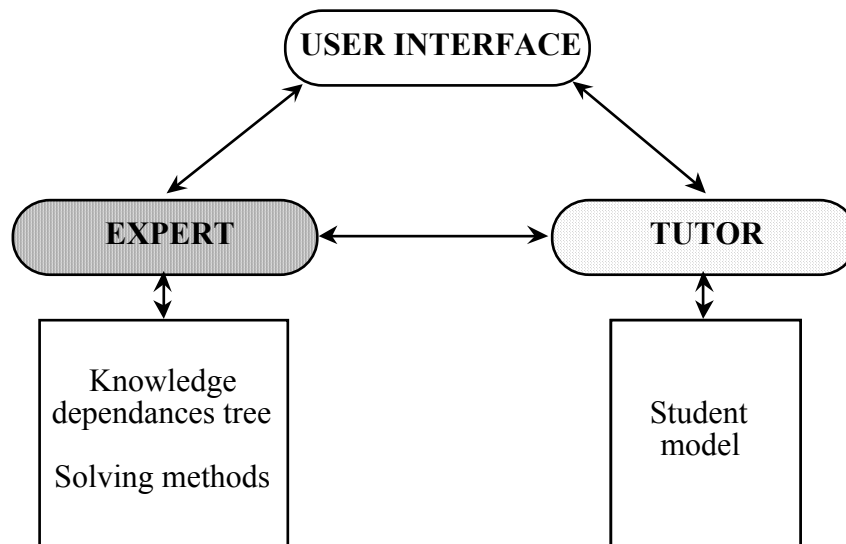


Figure 1: Software structure of the METAN application

These three units communicate to each other and have specific roles:

- The **expert**: It is the specialist of the domain and contains the knowledge base related to the training topics. This base is structured as a network of objects or tree of knowledge dependencies. Each object which type is knowledge is composed of several parts:

- references to hypertext entries in the course: the **tutor** may use these references to give to the student appropriate help ;
 - terms of the question: a list of terms or a call to a function which generate terms ;
 - knowledge base which contains all the useful data to solve the problem ;
 - training and correction methods, if specific methods are needed for this problem ;
 - etc..
- The **tutor**: It is the training expert of the application. It manages the hypertext course proposed to the student. It chooses the knowledge to teach and activates the corresponding training methods. To make the choice, it uses the tree of knowledge dependencies of the **expert**. The **tutor** manages a *student model* containing the knowledge that it has about the student. The *student model* is constituted by a list of concepts associated with booleans. Each boolean indicate if the student knows the associated concept. This allows the **tutor** to know the student deficiencies and to propose an appropriate help (for instance, a hypertext course or a description of the reasoning process).
 - The **user interface**: The course is proposed (by the **tutor**) as an hypertext document structured as a manual (parts, chapters, sections, etc.). It can be used sequentially like a book or using hypertext entries. This document is also used for correction when the student makes a mistake. The possible answers will be proposed to the student through a list in which he can select items. This list, which displays all available concepts, does not influence his decision.

5. The tutoring session

5.1. Organisation of a session

Figure 2 shows the organisation of a training session.

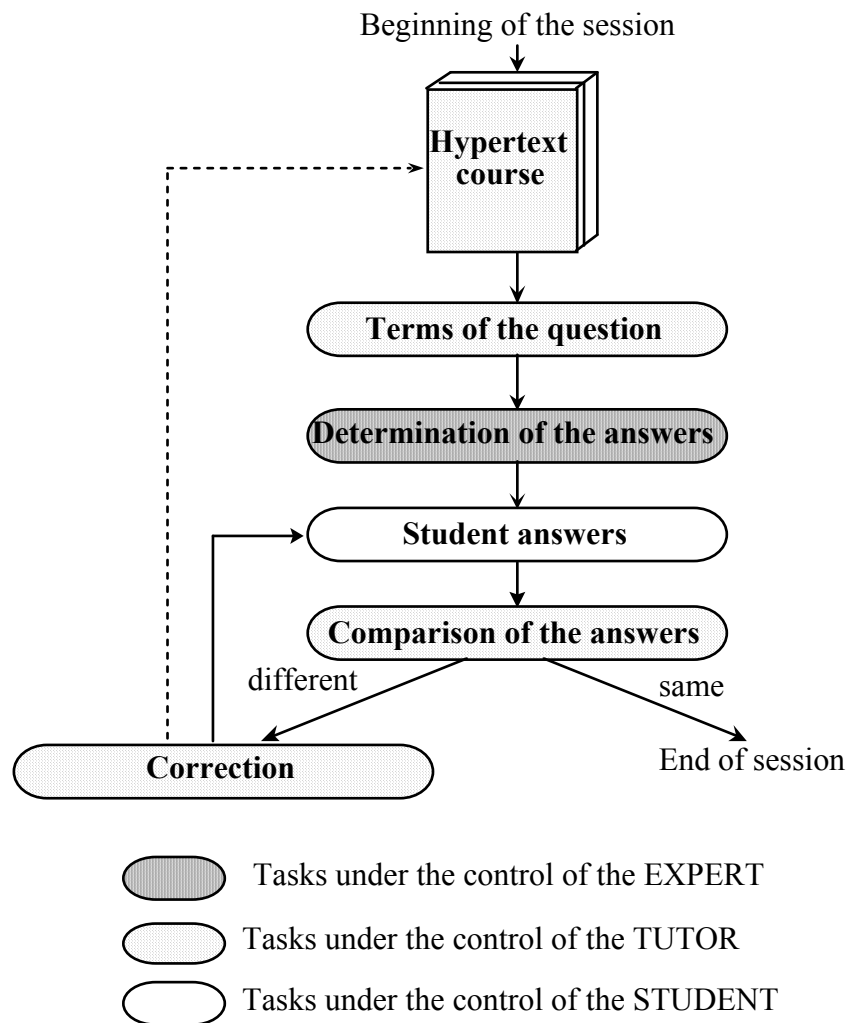


Figure 2: Organisation of a session

The different steps of a session are:

- hypertext course: the **tutor** proposes an hypertext course adapted to the training of necessary knowledge. The student can study this course as long as he wants and can short-cut some concepts or deepen others.
- terms of the question: the **tutor** establishes the terms of the problem and proposes them to the student.
- determination of the answers: the **expert** deduces the answers from the proposed terms. It generates a list of useful concepts which will constitute the reference list. It also creates a reasoning tree which can be used for explanation purposes.
- students answers: the student gives his answers as a set of concepts.
- comparison of the answers: the **tutor** compares the student answers with the **expert** answers from the reference list. If some concepts are missing, the student has three possibilities: he can access to the hypertext course to verify his knowledge, he can propose new concepts to add to his answers or he can abandon and ask for right answers. The student model is modified according to the given answers (booleans associated with the concepts are updated).
- correction: if the student fails in finding all the involved concepts, the **tutor** proposes, for each missing concept, an hypertext entry to the corresponding subset of the course.

5.2 An example of a session

Let us examine on an example how the argument is built (some of the hypertext links with the course are presented into brackets):

Question: Observe grains of 70/30 Brass

Step 1: inferences from the question

- To **observe** means to look with bare eye at one or several visible particulars (link: generalities)
- Visible means that an apparent positive contrast exists and that apparent size of the particulars is larger than eye's resolution (link: definitions)
- A **grain** is the basic element of a polymer or a metal sample (link: basics on materials)
- Size of grains is generally microscopic, i.e. invisible with eyes
- **70/30 brass** is a metallic alloy containing copper and zinc (link: metals)
- 70/30 means 70% copper and 30% zinc. This alloy is a single phase one (link: brass and copper alloys)
- In a single-phase alloy, the grains have an uniform composition, they are only distinguishable by their crystalline orientation (link: structure of materials)
- The crystalline orientation does not affect natural light reflection and transmission. Consequently, the contrast is nil (link: optical properties).

Conclusion of step 1: the sample is a single-phase alloy constituted of grains with uniform composition. This leads to consider two instances of the "Sample" class:

Class:	Sample	Class:	Sample
Sub-class:	Metal	Sub-class:	Metal
Name:	70/30 Brass	Name:	Grain
Material:	Brass	Material:	Brass
Structure:	Monophasic Alloy	Structure:	Monocrystal
Composed of:	Grains	Composed of:	Atoms
Attributes:	conductor: yes surface state: unknown contrast: nil ...	Attributes:	conductor: yes size: microscopic ...

Step 2: choice of attributes

- The object is the grain, itself a part of the "70/30 Brass" sample
- The grains are not visible because the contrast is nil and the size is too small.

Two properties are concerned:

- The **contrast**: this value is nil. It must be increased.
- The **size**: this value is "microscopic". It must be modified to a value visible with bare eye (around 30 microns).

Step 3: Choice of Actions and Methods

To improve the contrast, an appropriate instance of the "transformation" class exists, called "contrast increase", which is executed by the "crystalline orientation blow-up" instance of the "action" class.

Class:	Transformation
Name:	Contrast increase
Modified attribute:	contrast
Modification:	set to "strong"

Class: **Action**
 Sub-class: Chemical reactions
 Name: Crystalline orientation blow-up
 Attribute: chemical composition
 Needs: *polished sample*
 Transformations: Contrast increase
 Surface composition alteration

The "crystalline orientation blow-up" action is executed by the "chemical etching" method. To conclude, the contrast increase of the grains of the metal alloy is obtained by chemical etching (link: metallographic etching methods).

Class: **Method**
 Sub-class: Preparation
 Name: Chemical etching
 Applying criterion: *fitting with metal material*
 Needs: *polished sample*
 Action: Crystalline orientation blow-up

"Chemical etching" is a subclass of the sample preparation methods. It needs polishing of the sample and the choice criterion is the fitting to the metal or alloy, because each etching reagent is specific.

In order to reach a size magnification, there is one appropriate instance of the **transformation** class: "size magnification", executed by the instance of the **action** class: "magnification".

Class: **Transformation**
 Name: Size magnification
 Modified attribute: size
 Modification: multiply by the magnification factor

Class: **Action**
 Sub-class: Optic
 Name: Magnification
 Attribute: magnification factor
 Needs: *contrast different from nil*
 Transformations: Size magnification
 Observation field reduction

The "magnification" action is found in the "microscopes" methods. Thus, the size increase can be done by a microscopy method, subclass of the observation methods. The choice criteria are: magnification factor, use easiness and cost.

Two instances of the **Method** class are candidates:

Class:	Method	Class:	Method
Sub-class:	Microscopes	Sub-class:	Microscopes
Name:	Optical Microscope	Name:	Electron Microscope
Applying criterion:	<i>middle magnification</i> <i>easy use</i> <i>low cost</i>	Applying criterion:	<i>high magnification</i> <i>delicate use</i> <i>high cost</i>
Needs:	<i>polished sample</i>	Needs:	<i>conductive surface</i>
Action:	Magnification	Action:	Magnification

One needs a low magnification: around 100 or 1000, if possible at a minimum cost. This leads to the selection of one instance of the **Method** class: "optical microscopy" (link: microscopy methods).

From step 1, one did not learn anything about surface state. By default, this value is "any" and it should be modified to "polished" (needed by the "chemical etching" and "optical microscopy" methods). Applying the same argument, one finds the appropriate method: "polishing", instance of the preparation methods.

In order to apply the "chemical etching" method, the sample must be polished, so the succession of the two methods is fixed: polishing, then chemical etching.

The "optical microscopy" method is the observation method. It must be executed in last instance. One comes usually to that conclusion by noticing that the contrast is nil at the beginning: one must first increase the contrast and then observe it.

Conclusion of step 3: one must execute successively three methods: polishing, chemical etching and optical microscopy.

6. Conclusion

The structure of objects used in the METAN system has been determined, the content of teaching and the knowledge are collected from a master degree course of Ecole des Mines de Paris.

The present version of the application is composed of three units: the **tutor**, the **expert** and the **user interface**. The **expert** already contains several kinds of problems with the data to solve them. The **tutor** is a modest but evolutive module that can actually manage a *student model* as a list of concepts. It is able to evaluate answers and to propose an appropriate help when necessary.

The system is going to be tested with the students for whom it has been designed. On one hand, the prototype will be improved from these results, by increasing the explanation functionalities and by enriching the structure of the knowledge base. On the other hand, it will be continuously completed, following the evolution of the experts and the general progress of analytical methods. More generally, reasoning methods used in the METAN system can be applied to other engineering sciences.

References

- [Anderson 88] Anderson J.R. "The expert module", in Foundations of Intelligent Tutoring System, M.C. Polson, p 55-78, 1988
- [Askeland 89] Askeland D.R. "The science and engineering of Materials", 2nd edition, PWS-KENT publishing company, Boston, 1989
- [Astolfi 85] Astolfi J.P. "Pédagogie variée, diversifiée, différenciée", Cahiers Pédagogiques n°239, p19-20, december 1985
- [Briggs 83] Briggs D., Seah M.P. "Practical surface analysis by Auger and X-ray photoelectron Spectroscopy", John Wiley and Sons, Chicester, 1983
- [Brown 75] Brown J.S., Burton R.R., Bell A.G. "Sophie: A step towards a reactive learning environment", Int. Journal of Man Machine Studies, 7, p675-696, 1975
- [Clancey 82] Clancey W.J. "Tutoring rules for guiding a case method dialogue", Intelligent Tutoring Systems, Sleeman et Brown, p51-78, 1982
- [Cox 86] Cox B. "Object Oriented Programming: an Evolutionary Approach", Addison-Wesley, Reading, Mass. 1986
- [Joab 90] Joab M. "Modélisation d'un dialogue pédagogique en langage naturel", thèse d'Université de Paris VI, 1990
- [Marcenac 90] Marcenac P. "EDDI: Contributions aux Environnements de Développement de Didacticiels", thèse d'Université de Nice-Sophia Antipolis, 1990
- [Meyer 88] Meyer B. "Object-oriented Software Construction", Prentice Hall, New York 1988
- [Moustafiades 89] Moustafiades J. "Socrate: un système intelligent d'enseignement du diagnostic de pannes d'équipements asservis", thèse d'Université de Paris VI, 1989
- [VanLehn 88] VanLehn K. "Student Modeling", Foundations of Intelligent Tutoring System, M.C. Polson, p55-78, 1988