



PERGAMON

Expert Systems with Applications 26 (2004) 9–18

Expert Systems
with Applications

www.elsevier.com/locate/eswa

PetroGrapher: managing petrographic data and knowledge using an intelligent database application

Mara Abel^{a,*}, Luís A.L. Silva^a, Luiz F. De Ros^b, Laura S. Mastella^a, John A. Campbell^c, Taisa Novello^a

^a*Instituto de Informática, UFRGS, Avenida Bento Gonçalves, 9500, Bloco IV, Campus do Vale, Bairro Agronomia, Caixa Postal 15064, 91501-970 Porto Alegre, RS, Brazil*

^b*Instituto de Geociências, UFRGS, Avenida Bento Gonçalves, 9500, Campus do Vale; Bairro Agronomia, Caixa Postal 15001, 91501-970 Porto Alegre, RS, Brazil*

^c*Department of Computer Science, University College London, Gower Street, London WC1E 6BT, UK*

Abstract

This paper describes the *PetroGrapher* system, an intelligent database application to support petrographic analysis, interpretation of oil reservoir rocks, and management of relevant data using resources from both knowledge-based system technology and database technology. In this project, the visual tacit knowledge applied in petrographic analysis was rendered explicit through the collection of cases (rock descriptions), which were then used in the development of a domain ontology organized in a partonomy. Expert-level basic features, which we call ‘visual chunks’, were identified. The cases were further compared against the ontology to elucidate the relations between features in descriptions of rocks, visual chunks and expert interpretations. The domain knowledge was represented through a set of frames and knowledge graphs. The knowledge graphs are applied to recognize the visual chunks in the user data and retrieved the related interpretation. The system was developed as a structure tightly coupled with a relational database system, which acts as a repository for the knowledge base and the user data, and an object-oriented component, which preserves the semantics of data and develops inferences. The system was validated by three groups of users with different levels of expertise.

© 2003 Elsevier Ltd. All rights reserved.

Keywords: Knowledge engineering; Intelligent databases; Sedimentary rock interpretation

1. Introduction

Along with war, oil exploration is probably the largest-scale human activity for which the acquisition, circulation and use of knowledge are critical for decision-making. The extremely large costs of offshore exploration impart an extraordinary value to the knowledge that supports decisions, such as choosing an area in which drill, the reservoir-rock unit to target, what procedures to use during drilling, what are the criteria and parameters for the evaluation of the rock and the fluid data generated, and what methods to use during development and production of the oilfields. The very large amount of data and the multiple areas of expertise that are essential for oil exploration

generate a huge volume of knowledge, which oil companies struggle to secure, administer and use in the most effective ways.

By contrast, the knowledge management of critical areas within oil exploration is still inefficient and in no more than an early stage of development. One of these areas, the special focus for our project, is the systematic description and interpretation of reservoir rocks, or Sedimentary Petrography. This area can be understood as the base for criteria for the evaluation of oil reservoirs as future productive oilfields, as well as for increasing the output of oilfields that are under development. Petrography is a specialized and labor-intensive activity in which large amounts of data are generated, both in qualitative (textual), and in quantitative (numerical) formats. Despite the key importance of petrographic information in oil exploration, the aggregated knowledge is usually poorly managed, for many different reasons.

The most common pitfall in management of petrographic information about reservoirs is the lack of standardization of

* Corresponding author. Tel.: +55-51-33166825; fax: +55-51-33191576.

E-mail addresses: marabel@inf.ufrgs.br (M. Abel), llima@inf.ufrgs.br (L.A.L. Silva), lfderos@inf.ufrgs.br (L.F. De Ros), mastella@inf.ufrgs.br (L.S. Mastella), j.campbell@cs.ucl.ac.uk (J.A. Campbell), novello@inf.ufrgs.br (T. Novello).

the nomenclature and of the data-acquisition procedures. As a consequence, the domain knowledge is poorly understood and transferred and the learning process is confined to being an individual experience. Furthermore, commonly each petrographer, group of petrographers or project team develops an individual ontology and set of data, and only general information finds its way into the corporate database. Consequently, most of the petrographic information is not efficiently retrievable when one wants to combine, refine and reuse to produce new knowledge. Because of these human and technical limitations, the use of petrographic information and knowledge within oil companies is limited to a fraction of the potential of petrography as an exploration and production tool—or, worse, wrong decisions are made based on limited, biased and/or misunderstood knowledge.

Knowledge-based systems (KBS) can apply powerful abstract constructs to model the petrographic domain and operate robustly even with few instances of these constructs. The user is not required to know fully the model to interact with the system. However, when an application requires an extensive search over atomic data items of the same kind, such as tables or numerical data, such systems degrade in performance. The traditional KBS was not intended to support bulky processing of persistent knowledge. On the other hand, database (DB) applications apply simplified constructs to represent data and store a huge quantity of instances of the data. The retrieval of a large amount of the same type of data is very efficient, even though the user should know completely the DB schema to formulate the queries. Also, multiple consultations over particular values of specific instances, which are typical for KBSs, become a very complicated task to accomplish using traditional DB query languages and therefore reduce the performance of the consequent search (Bertino et al., 2001).

Relational database management systems (RDBMS) were designed to solve a particular class of problem—store, manage and consult a large amount of the same type of data—in opposition to KBSs, which are intended to extract new useful knowledge from knowledge that is already represented and stored. They achieve their goals by applying distinct views over the reality with different levels of granularity of the captured objects (Bertino et al., 2001; Mattos, 1991). Taking advantage of both resources in developing an information system for geology faces problems that go beyond the simple integration and mapping of data and knowledge with distinct semantic levels. Intelligent database (IDB) systems integrate resources from both RDBMSs and KBSs to provide a natural way to manage information, making it easy to store, access and apply (Bertino et al., 2001; Parsaye et al., 1989).

Our aim with this work is to present the main features of an advanced, IDB application called *PetroGrapher* which is been developed to orient, standardize, manage, retrieve, and process petrographic information, as well as to produce

interpretations of the genetic and quality aspects of reservoir rocks of sandstone.

Basically, petrographic interpretation refers to the formal description of visual aspects of a rock sample, as they appear in naked-eye analysis and under an optical microscopic. Starting from the petrographic features that are discerned, the petrographer infers the possible geological interpretation(s) of the rock, which will strongly influence the method of evaluation of the potential of the geological unit as an oil reservoir. The geologist analyses the diagenetic environment in which the rock was possibly produced, according to the features that are imprinted in the rock by the conditions of this environment.

The *PetroGrapher* IDB application was conceived in order to allow a user with a medium level of expertise to describe petrographic features in his/her own level of technical language. The system has the role of applying knowledge to recognize within those features the items that can serve as diagnostic cues for higher levels of expertise in interpretation, in some imitation of a process of visual interpretation (but with even images being described symbolically). The knowledge was understood and elicited with the help of an expert using knowledge engineering techniques, and modeled through data and knowledge models, which were implemented in an object-oriented system.

The description of rock features through symbolic models has been attempted previously, in the Rock III phase of the Sisyphus knowledge acquisition project, (Gaines, 1998; Gappa & Puppe, 1998; Jansen et al., 1998), whose goal was to develop a knowledge base to act as a tutorial source for igneous rock identification. The original problem of rock classification, which requires a medium level of expertise in Petrology, was transformed into a problem of identifying the minerals of the igneous rock, clearly involving a lower level of expertise. In our interpretation, the project failed to achieve the goal (as said in) (Gappa & Puppe, 1998) because the knowledge base provided primitives to describe individual minerals, but no constructs to represent the relevant relationships among them that had been recognized by the user. An experienced petrologist would not classify a rock by first recognizing its minerals (a false premise in the Rock III experiment) but he/she would see first the many relationships among many minerals at the same time. A rock feature, in the petrologist's mind, acts as a pattern which is recognized almost without any intervening abstract level of inference.

The Petrog system, from Conwy Valley Systems Ltd, (Conwy, 2002) has better constructors to describe sedimentary rock features, including textures and paragenetic relations. These could be improved and/or extended by the use of knowledge associating them with diagenetic features of oil reservoirs, if some interpretation functionality were added to the system.

Without such knowledge, non-symbolic approaches applied to rock interpretation have also achieved poor

results when dealing with sedimentary rocks. The complexity of the environment introduces more variables and dynamic behavior that could be treated by a purely numerical approach. The ADDGEO software (Garcia et al., 2000) applies neural networks to identify constituents in carbonate rocks (which are mainly bioclasts). The software allows capturing an image from a thin section and selecting a part of the image to be tested. The neural network attempts to identify the class of bioclast based on the shape of the boundary of the constituent selected. Since the identity of carbonate constituents are defined by many other visual attributes (including paragenetic relations with other elements in the rock and knowledge about the sedimentary unit where the sample has been collected), the system only succeeds when the identification is simple enough to be delivered by a novice.

Geological knowledge models, here exemplified through the Petrographic domain, need to deal with large numbers of visual attributes, which are chosen, combined and applied by the expert in some abstract (thus, in practice, symbolic) way dynamically during the inference process. These characteristics have not been fully analyzed in previous projects. In the *PetroGrapher* project we have acknowledged these limitations and proposed some tactics for capturing part of the expertise in rock interpretation within a software application. We believe that the development approach and the architecture of this project can be reused in other knowledge-intensive domains that demand visual knowledge in the problem-solving process.

2. Knowledge acquisition and modeling

Expertise in sedimentary petrography is mainly built from *visual knowledge*, which means that the information that supports the reasoning process is firstly collected by visual pattern-matching—configuring what was described by Nonaka and Takeuchi (Nonaka, 1994) as tacit knowledge—and only later on will be (partially) named and organized for incorporation into an individual's explicit body of knowledge. The visual knowledge includes the recognition of constituents (a very hard task, as pointed out by), (Gaines, 1998; Jansen et al., 1998) textural and structural features and paragenetic relations, all of them fundamental tasks in reservoir evaluation. The challenge in this project was in identifying and naming the visual knowledge applied by expert in support of inference, in order to add this knowledge to the domain ontology, which had the role of being the framework for the schema and knowledge base of the IDB application.

Initially, the knowledge about description and interpretation tasks for reservoir rocks was acquired as cases, or complete descriptions of reservoir-rock samples with their corresponding interpretations. The main contribution of this case-based modeling approach was in making the knowledge acquisition process easier, and providing the general

structure to represent the declarative part of the domain knowledge. The structure of the collected cases is mainly an aggregate of parts, each one describing some particular aspect of the rock sample (mineralogy, textures, etc.) and related in some way to the interpretation, also included in the description (the solution part of a case). As a consequence, the descriptive model developed from the cases was not organized in an hierarchical structure of concepts based on *is-a* relationships, but built as a *partonomy* in which the whole set of concepts correspond to an unique case, represented by the concept *Sample* (Fig. 1a). The *part-of* relationship is the main structural relation in many ontological models, especially when they are intended to represent documents or semi-structural data, such material that is shared via the Internet (Studer et al., 1998), or the rock descriptions studied in this project.

The set of cases provided by the expert was abstracted to compose the aggregated concept *Sample* which represents the root of the knowledge schema. *Sample* is composed of a set of concept components, each one representing a kind of reservoir rock analysis, such as *Macroscopic Description* concept, *Microscopic Description* concept, etc. The *Diagenetic Interpretation* of the rock, whose instance is generated by the inferential component of the IDB application, is also a component of the concept *Sample*.

The knowledge structure was represented in a frame-based formalism, whose relationships (such as *part-of* in Fig. 1b) and properties (e.g. *Modifier* and *Location*, in Fig. 1b) are both represented as slots of the concepts. The facets of concepts are used to represent the integrity constraints over the attribute values, such as the domain or range of values, or an instance of another concept. Fig. 1b shows facets such as *one-of*, but the model includes *list-of*, *value-type*, *maximum-cardinality* and *range* as valid restrictions over the attributes.

Having defined the ontology and representation of the domain, the expert was invited to show the relations between the described features of the model and pieces of the interpretation from each collected case. This approach drew special attention to the rock aspects that were not being described, because they were not part of the known and accepted domain ontology yet but, even so, had been used for interpretation (Abel, 2001). Each of these visual aspects applied to the defined diagenetic interpretation was further associated with a group of features that could approximately describe it, based on the common nomenclature of sedimentary petrography. These groupings were first called *visual chunks* by Abel (2001), by analogy with the term *chunking* applied by De Groot in his earlier work about reasoning in chess (De Groot, 1965; VanLehn, 1989). Each of the recognized visual chunks was given a geological name by the expert, which then became part of the knowledge nomenclature.

Although the visual chunks have received names allowing unique identification in the model, they are not yet part of the domain ontology, that is, the name has no

Sample Concept	
Identification Concept	
Qualitative Description Concept	
Macroscopic Description Concept	
Microscopic Description C	
Diagenetic Composition C	
Detrital Composition Conc	
Diagenetic Sequence Con	
Genetic Interpretation Conce	
Provenance Implications C	
Depositional Environment/	
Diagenetic Environment/P	
Classification Concept	
Porosity Concept	
Macroporosity Concept	
Microporosity Concept	
Quantitative Description Conce	
Compositional Table Concept	
Total Table Concept	
Other Analysis Concept	
Important Observations Conc	
Pictures Concept	
Additional Analysis Concept	
	Diagenetic Composition Concept
	Part-of Sample Concept
	Set-of Diagenetic Constituents Concept
	SampleID one-of <Identification Concept.SampleID>
	ConstituentName one-of <Diagenetic Nomenclature Concept.ConstituentName>
	ConstituentSet one-of [Silica, Feldspar, Infiltrated clays, Pseudomatrix clays, Authigenic clays, Zeolites, Carbonates, Sulphates, ...]
	Habit one-of [<Habit Nomenclature Concept.HabitName>]
	Amount Real
	NominalAmount one-of [Abundant, Common, Rare, Trace]
	Location one-of [Intergranular continuous pore-lining, Intergranular discontinuous pore-lining, Intergranular pore-filling, ...]
	Modifier one-of [Dissolved, Zoned, Fractured, Recrystallized]
	ParageneticRelation one-of [Covering <one-of [Diagenetic Nomenclature Concept.ConstituentName]>, Covering <one-of [Detrital Nomenclature Concept.ConstituentName]>, Covered by <one-of [Diagenetic Nomenclature Concept.ConstituentName]>, Replacing grain of <one-of [Detrital Nomenclature Concept.ConstituentName]>, Replacing matrix of <one-of [Detrital Nomenclature Concept.ConstituentName]>, ...]
	ParageneticRelation ConstituentSet one-of [Silica, Feldspar, Infiltrated clays, Pseudomatrix clays, Authigenic clays, ..., Detrital quartz, Detrital feldspar, Plutonic rock fragments, Volcanic rock fragments, ...]
	Description String

a) The domain concepts that define the case model in the knowledge acquisition phase

b) The slots and facets with associated domain values of the Diagenetic Composition Concept

Fig. 1. Some concepts of the domain model.

common shared meaning among petrographers. Ontology, a philosophical concept discussed first by Aristotle at some time before 322 B.C., studies all the issues involved with the things (physical items, concepts, etc.) that constitute the special 'world' for some community; in particular, the names that people agree to use whenever they refer to those things. More recently, (Benjamins, 2000; Fernández-Breis & Martínez-Béjar, 2002; Gómez-Pérez & Benjamins, 1999; Gruber, 1992; Guarino, 1995) the word 'ontology' refers to the descriptive part of a knowledge model. As the terms extracted during our knowledge elicitation are mainly unknown to geologists (who have had no prior experience of or training to use those terms), any rule or relation built around them by the IDB application would not be particularly helpful as a support for human reasoning because they are not present in the geologists' ontology.

The relationships between tacit features and interpretations were explored first in the work of Leão (1988) which has defined *knowledge graphs* (or *K-graphs*) as a knowledge-acquisition tool to extract relations between symptoms and cardiopathic dysfunctions in medical diagnosis. The definition of *K-graphs* was extended in this work as a formalism to represent visual chunks and the component relationship connecting each chunk with the features that can describe it. They also describe the diagenetic interpretations and the implicit relationship that links visual chunks with the interpretation of a sample. The two relationships support the inference process in the IDB application.

K-graphs have greater expressivity and granularity when compared to other formalisms which associate evidences to hypotheses, such as production rules or Bayesian nets. For example, what can be expressed in one *K-graph* here can easily require more than 20 production rules in that alternative formalism. The knowledge modeling using *K-graphs* is able to represent the expert knowledge about the petrographic problem-solving, minimizing the semantic loss that would occur if production rules were used instead. Besides, the use of production rules demands several forward or backward chaining steps in order to find which pieces of evidence are associated with which hypotheses, or vice versa, whereas *K-graphs* make available, after just one query to the DB, all the information that must be sought by pattern-matching in the DB of rock sample descriptions.

K-graphs can be considered a special kind of Bayesian net. However, we can observe that the main difference between the two is in their node structure. All the nodes represented in a Bayesian net have the same semantics (one node represents one domain variable, which owns a set of states with their associated probability values), whereas nodes of *K-graphs* represent three different structures: interpretation, chunks and features, which have different semantics. Moreover, as Bayesian nets are probabilistic nets, every single statistic about the uncertainty in the domain variables must be available to the knowledge engineer, in order to model the domain.

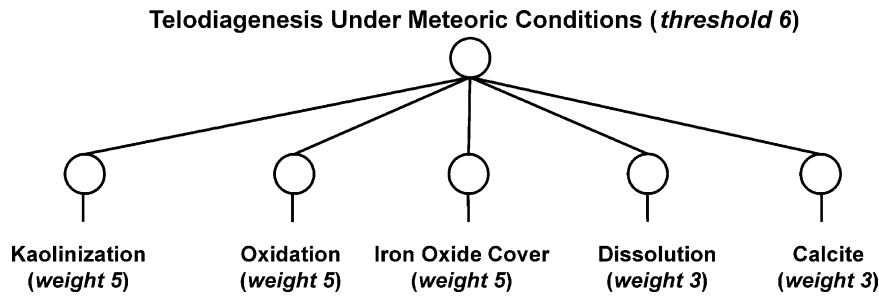


Fig. 2. A K-graph and associated visual chunks as acquired from the expert.

We define an adapted representation of K-graphs as an AND/OR tree in which:

- The root node represents the diagenetic interpretation to be suggested for the particular rock sample;
- The leaf nodes represent the visual chunks identified by the experts in the rock sample image, as evidence to support the interpretation. These nodes are ordered by their influence in indicating inference and can be combined (i.e. considered together) to increase these influences and the degree of certainty of the interpretation.

In our IDB application, K-graphs are used in arriving at suggestions of the possible origin of a rock sample (diagenetic environment/process interpretation task), based on the evidence described by the user according to the domain’s shared ontology. The diagenetic environment/process interpretation task is a typical and central task in visual interpretation in the sedimentary petrography.

Influence indices can be assigned to each chunk using a scale of 1–6, meaning the influence level of the presence of chunks in indicating some particular interpretation. For instance, *Kaolinization* (Fig. 2) has weight 5 to indicate *Telodiagenesis under Meteoric Conditions interpretation*. The depicted K-graph has four other chunks: *Oxidation*, *Iron Oxide Cover*, *Dissolution*

and *Calcite*, with, respectively, weights 5, 5, 3 and 3. No one chunk by itself in this K-graph can indicate the interpretation; what is necessary to achieve any interpretation here is a combination of these chunks.

The K-graphs in the reservoir rock interpretation domain are associated with threshold values, indicating a sufficient number of diagenetic environmental aspects that may be found in the image description of a rock sample. In Fig. 2, the threshold to achieve the interpretation is 6. The diagenetic environments imprint their characteristics on the rock in more than one aspect; the more aspects are found in the image of the rock sample, the more probably is the interpretation correct.

The visual chunks are described in the K-graphs as logical combinations of CAV (concept–attribute–value) triples, which describe geometric aspects (color, size, texture) represented in the domain ontology. Such aspects exist routinely in the knowledge of a petrographer who is not an expert but is better informed, through practice, than a good student. But the particular combination of these common features, which is only recognized in the high levels of expertise, can lead to a specific interpretation which is different from any one that could be obtained from the same features viewed separately in isolation.

Fig. 3 shows an example of this needed logical combination of CAV represented as a visual chunk named

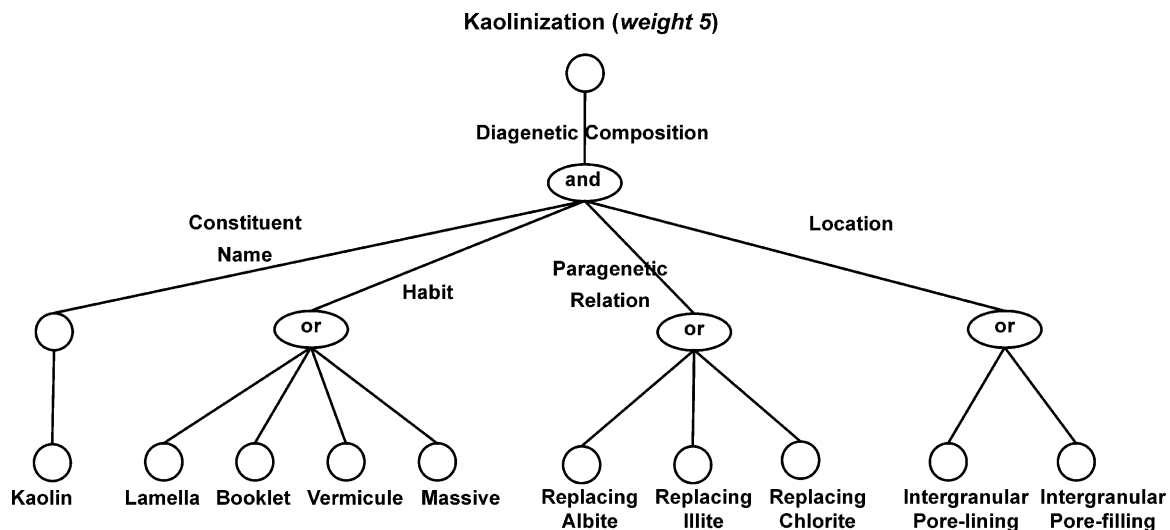


Fig. 3. The AND/OR structure of the Kaolinization chunk.

Kaolinization. This chunk is represented by the logical sentence:

Diagenetic Composition with constituent Name = Kaolin AND (Diagenetic Composition with habit = Lamella OR Diagenetic Composition with habit = Booklet OR Diagenetic Composition with habit = Vermicule OR Diagenetic Composition with habit = Massive) AND so on.

In our project, we have modeled 33 different visual chunks and their respective weights on the way to constructing six different *K*-graphs of reservoir rock interpretations, which can then be analyzed by the inference component of the IDB application and matched against an image description of a rock sample. The ‘Sample’ concept is defined through 130 different attributes (e.g. *GrainSize*, *Shape*, *Orientation*) distributed over the 13 subcomponents of *Sample* (which can be *Detrital-Composition*, *Microporosity* or *Rock-Classification*, among others), along with eight instantiated concepts which compound the knowledge base (and which also describe the porosity, mineral and habits catalogue). The main assumptions and requirements of this problem-solving method can be found in [Abel \(2001\)](#) and [Silva \(2001\)](#).

3. The mapping between the knowledge model and the database model

IDB architectures rely on the idea of having an integrated data and knowledge management tool ([Schreiber et al., 1999](#)). The proposed architecture ensures that the RDBMS and the KBS can remain independent even though there is a tight effective coupling during the inference process. In our project, the domain knowledge model was mapped to a DB model, producing a DB schema for the IDB application. The problem-solving model also was mapped to a DB model, enabling the storage and management of large amount of structural data about problem-solving concepts. Therefore, two kinds of DB components were produced in the IDB application approach:

- *Application DB*: the repository of user descriptions of rock samples, expressed according to the domain ontology;
- *Knowledge DB*: storage of meta-data about the schema and ontology of the domain (concepts, relations, chunks and *K*-graphs). These data about problem-solving structures are queried by the IDB application and the inference process is implemented without a strong coupling with the RDBMS query engine.

The *application DB* and the *knowledge DB* are treated as data repositories, without any deductive or active role ([Parsaye et al., 1989](#)) within the IDB application. The mapping process that facilitates the production of the application DB was based on a semi-automatic classification approach, using only two simplified levels of distinction: *main concept* and *secondary concept*.

The main concepts are the main semantic parts of the root concept (in this case, the object *Sample*, shown in [Fig. 1a](#)). The *secondary concepts* are the descriptive components associated with these *main concepts*: slots, facets, etc. Also, it is necessary to design the application DB taking into account query efficiency of the IDB applications and organizing the generated DB entities accordingly.

Using this basic approach of concept distinction on the modeled concepts in the domain knowledge, the representation primitives can be mapped using the following rules:

- R1: *main concepts* are mapped to main entities. For example: the Diagenetic Composition concept is mapped to a Diagenetic Composition entity;
- R2: *part-of* relationships between *main concepts* are mapped as aggregation relationships between main entities. For example: the Diagenetic Composition concept is a part of the Sample concept. Then, the Diagenetic Composition entity is an aggregation of the Sample entity;
- R3: *set-of* relationships between *main concepts* and *secondary concepts* are mapped to multiple cardinalities between mapped entities. For example: the Diagenetic Composition concept is a set of instances of the Diagenetic Constituents concept. Then, the Diagenetic Composition entity is associated with multiple occurrences of the Diagenetic Constituents entity;
- R4: single-value facets of *main concepts* are mapped to attributes of main entities. For example: the Diagenetic Composition concept has a Location slot characterized by a one-of facet. This Location slot is directly mapped to an attribute Location of the Diagenetic Composition entity;
- R5: multi-value facets of *main concepts* are mapped as secondary entities associated with the main entities already mapped. For example, the Macroscopic Description concept has a Structures slot characterized by a list-of facet of name and scale of macroscopic structures. This Structures slot is mapped to a Macroscopic Structure entity with name and scale attributes and the Macroscopic Description entity is associated with multiple occurrences of the Macroscopic Structure entity;
- R6: structured facets of *main concepts* are mapped to multiple attributes of main entities. The Diagenetic Composition concept has a ParageneticRelation slot, which is described by a ParageneticRelation string name and the ConstituentName involved. Thus, the Diagenetic Composition entity is formed by two mapped attributes: a ParageneticRelation string name and a rock ConstituentName.

The *K*-graph structure of concepts were also mapped as three DB entities and related attributes. The first entity—*K-graph entity*—is used to store structural concepts of *K*-graphs, such as reservoir rock interpretation strings and threshold indices. The second entity—*chunk entity*—is used

to store structural concepts of chunks, such as chunk names and weights to related interpretations. The third entity—*domain concept entity*—is used to store the CAV leaves of the structure of chunks. AND and OR relationships between CAV leaves were modeled as additional attributes in the *domain concept entity*. Finally, the *K-graph* entity was associated with multiple occurrences of the chunk entity, and each chunk entity was associated with multiple occurrences of the domain concept entity, representing cardinality attributes between RDBMS entities.

4. PetroGrapher: an intelligent database application

The main components of the IDB application *PetroGrapher* (Fig. 4) are an application-oriented visual interface and an object-oriented (OO) control system, both implemented in Java and attached to an RDBMS, through a standard SQL interface. Other software components are also associated with the system to provide particular functions, such as quantitative analysis or visual evaluation of the *K-graph* structures (Fig. 5). The IDB application and the RDBMS are tightly coupled, showing low interaction during the input phase and high interaction to accomplish the inference process. External XML interfaces are provided to allow the export of data to foreign systems, such as text editors, spreadsheets or external DB applications.

The tightly coupled architecture permits combination of the high expressiveness of representation of KBSs with the capacity of RDBMSs for robust and safe storage, both necessary characteristics for effective knowledge management of oil exploration. Our application allows many users in geographically distant places to input rock descriptions supported by highly informed interfaces and methods for

applying expertise-level interpretations to these descriptions. Furthermore, the descriptions are safely stored and shared through well-known RDBMSs, which allow many different kinds of cross-consultations and statistical operations over the data.

The *PetroGrapher* IDB application provides a user-driven and reliable environment incorporating the domain nomenclature, methods and restrictions of sedimentary petrography, in order to help geologists in identifying petrographic features in sedimentary oil reservoir rocks. The interface guarantees the standard format and nomenclature of the description, suggesting a natural sequence of the description and making available the petrographic terms for each feature of the reservoir rock, e.g. via rock compositional diagrams, visual tables of grain roundness and size. The resulting description of a rock sample is kept in main memory until the user explicitly asks for the information to be saved. Whenever this happens, the OO control system generates the corresponding DB query commands to store the information in the application DB.

The OO control system is the main component of the *PetroGrapher*. Its function is shared among three subsystems or components. The concept component implements the domain knowledge model as classes and methods, which preserve the organization of domain concepts, their attributes and relationships. Most of the integrity constraints expressed in the slots and facets are managed by the RDBMS, mainly the constraints related to primitive types or possible sets of values. Some of them, however, are too complex to be managed by traditional RDBMS procedures and they are kept by the concept component, especially those characterized by multi-value facets.

The OO control system also includes a set of subsystems that support many aspects of the tasks of petrographic

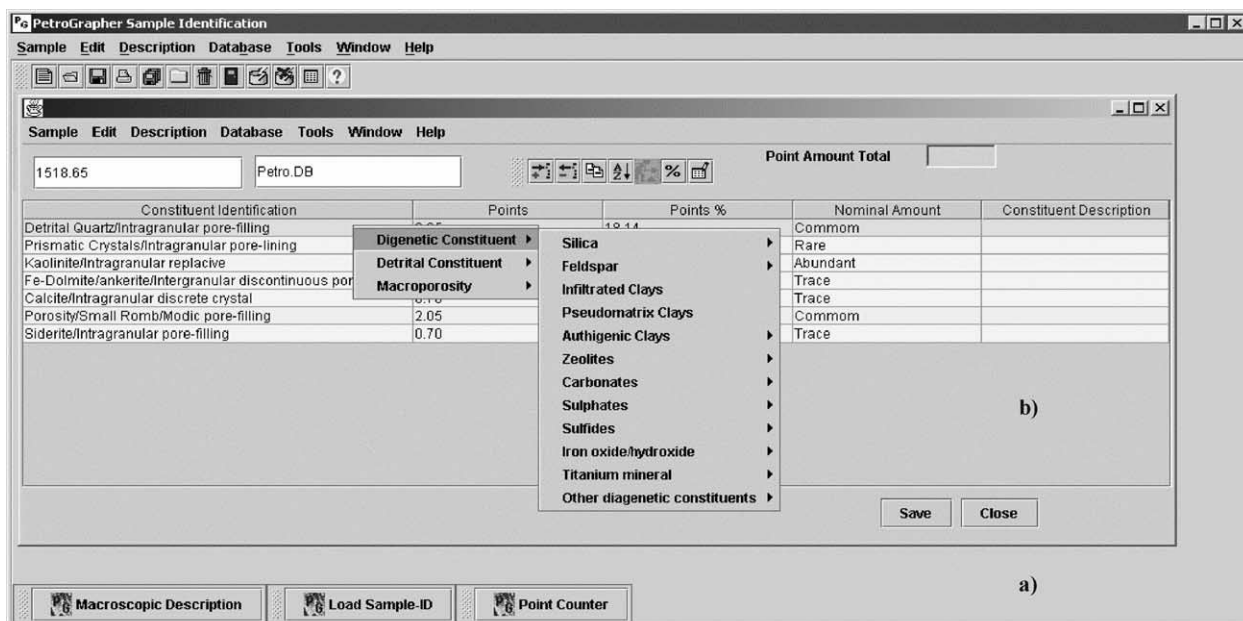


Fig. 4. PetroGrapher: an IDB application to petrographic data and knowledge management.

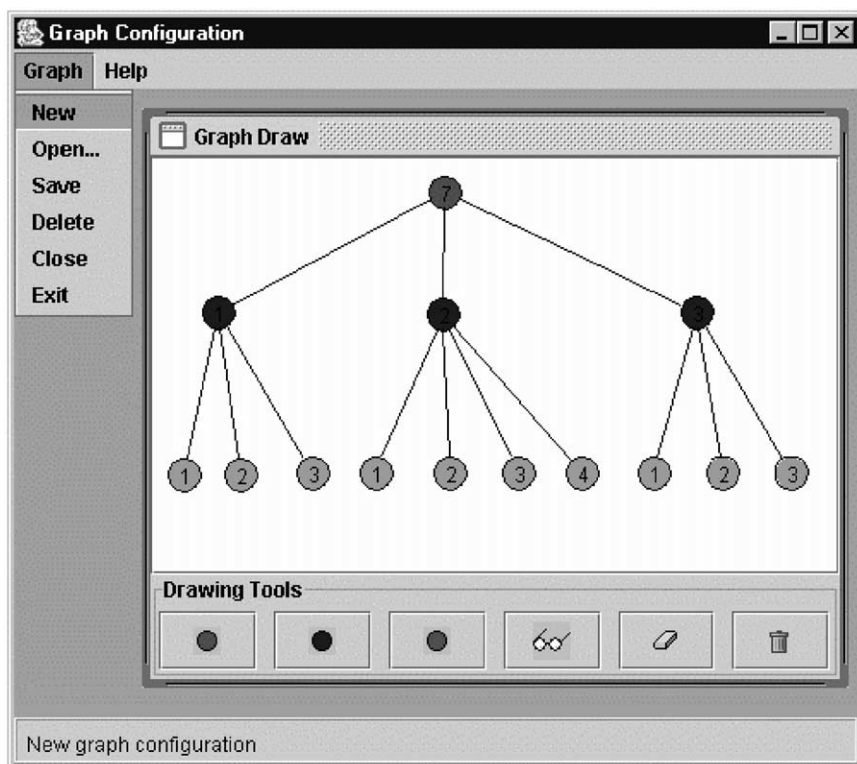


Fig. 5. PetroGrapher: the component for supporting causal relation extraction.

description. These include several forms of statistical analysis of the amount of the rock components, and the formal sandstone classification obtained from quantitative analysis based on different authors and methods. The current version of the system provides its compositional classification according to the methods of Folk (1974) and McBride (1963), with both the actual and original composition (i.e. before diagenesis). Also, the system suggests the provenance of the sediments, according with the Dickinson method (Dickinson, 1970). In each method, the user can visualize the correspondent triangle of mineral composition.

The OO control system has an inference component, which implements the inference process based on *K*-graph models, by querying their data structure description in the knowledge DB and building the *K*-graph structure in main memory. The automated suggestion of a diagenetic environment/process interpretation, achieved by matching the several parts of the description stored in the application DB against the nodes of the *K*-graphs, can optionally be started by user command at the end of the process of image description of a rock sample. The heuristics for diagenetic interpretation were provided by the expert and can be found in part in Anjos et al. (2000), Garcia et al. (1998) and Spadafora et al. (1998).

Fig. 4 shows the *PetroGrapher* system's main interface (Fig. 4a) and the point counter component (Fig. 4b). Fig. 5 shows the causal relation extraction component. This interface was developed for allowing the user to select, visualize and qualify a substantial set of distinct ontological

concepts (in this case, constituents are represented by constituent sets such as Silica or Feldspar). Each rock feature and its attributes are displayed as concepts in the interface. However, the great number of elements in the screen and the many properties to be defined make the task of interface design a challenge. The causal relation extraction component visually represents the causal relations that exist between the ontological concepts and the possible domain interpretations.

The general inference process is forward chaining and bottom-up, driven by the *K*-graphs (from CAV triples to interpretation) matched against the CAV triple expressing a user's description. It is not expected that the whole set of *K*-graphs will be matched against the information given by the user. The best match is calculated by the number of chunks found in user features which match against the chunk representation in the *K*-graphs, combined with their relevance indexes. The larger this number, the better the match. In these processes, each *K*-graph is loaded from the knowledge DB and tested in a predefined order. This high interaction between the two systems causes degradation in the overall computational system performance during the inference process. Even so, the desired independence of data justifies such degradation, since it can always be expected to be confined to a few moments of the interaction. The final interpretation is composed by the interpretations and the chunks found in the *K*-graphs that are fired, which is then stored in the user DB along with the description of the sample.

The inference algorithm was validated using a set of 23 cases previously described by an expert petrographer. The expert had also analyzed each inference path to adjust the weights and thresholds of the chunks and *K*-graphs. The efficiency of the inference component is limited by the user's capacity to recognize and describe the diagnostic features in the reservoir sample under analysis. Non-expert geologists may not be able to recognize specific diagnostic rock features, even if they are part of the ontology of petrography, which therefore limits the applicability of this inference algorithm. Nevertheless, according to both the expert and the (significantly less expert) test group, any suggestion of correctly derived interpretation is a useful insight for such a complex task as the interpretation of oil reservoir rocks.

The OO control system is attached to an RDBMS which stores all descriptions managed by the IDB application, plus the structure of the domain knowledge base. The DB schema for the application keeps a correspondence with the format of the cases in the OO control system, which is implemented in such a way that the two systems can translate the concept and attribute representations between themselves. Moreover, the knowledge DB keeps all nomenclature terms of the domain knowledge model and additional information needed by the system to carry out the reservoir rock description tasks. There is no domain knowledge in the system that is not stored by the RDBMS. This facilitates the development of other related applications using the same knowledge.

5. Conclusions

We have presented here the modeling process employed to build an IDB application for petrographic analysis. The resulting system, *PetroGrapher*, has been designed to support a complex visual analysis task. The system was developed by acquiring the domain ontology from an expert and the further definition of an application which applies the acquired ontology to manage a large amount of heterogeneous data and complex knowledge. The knowledge acquisition techniques and the proposed system architecture can be reused in task-related domains.

In order to support the task of petrographic visual analysis, diagenetic analysis and data and knowledge management in the petrographic domain, we have proposed a tightly coupled IDB application which integrates an OO implementation with an RDBMS. The data and knowledge requirements were acquired from an expert in sedimentary petrography and further modeled in ways that take proper account of the sometimes quite complex semantics of the knowledge, and were mapped to a DB model. Integration between the KBS and RDBMS features is accomplished dynamically in the OO control system which maps the conceptual objects of the domain and problem-solving model to the DB model, generating the related DB query commands.

The system was validated in three distinct stages and groups:

- The expert validated the knowledge domain model after each stage of development: knowledge acquisition, when the models were made explicit only on paper; implementation, which was tested the completeness of the models and correctness of the inference algorithm; validation; when the system was made available for thorough use by a test group. At each stage, the model was subjected to modifications and adjustments;
- A group from a petroleum company was selected to evaluate the system (in its alpha-test version) for correctness and completeness of its model and adequacy of the interface for 20 days with the support of one of the system developers;
- A group of sedimentary petrography students used the system as a supporting tool for the course whenever the opportunity presented itself (that is, defined by their university schedule rather than the interests of the project).

The IDB application proposed in this work supports KBS and RDBMS functions with high flexibility, providing a strong tool to support decision-making in a knowledge-intensive environment. The *PetroGrapher* IDB application is able to represent the relevant knowledge, while preserving the necessary semantic richness of the domain and offering a user-friendly environment to accomplish the complex task of petrographic description. The acquired and produced information is then made available for consultation or sharing by any user or external system through the DB resources. Despite this success, several problems related to the distinct programming paradigm integration still remain: for example, the system degrades in performance when the level of interaction becomes high, mainly during the inference process. The decrease in performance is a direct result of the orthogonal form in which RDBMS and KBS refer to the data set. RDBMSs are intended to manage large group of data of the same type, while KBSs deal with pieces of heterogeneous information. The integration of the two paradigms requires trade-offs between these different views and leads to an extra overhead of processing. Such issues open interesting opportunities for future research in both the *PetroGrapher* project and in knowledge-based computing in general.

Acknowledgements

This work is supported by the Brazilian Project Financing Agency—FINEP and Council of Research—CNPq through the founding program CTPETRO. We are pleased to record our gratitude to the Petrobras group of geologists who have given help during the evaluation phases of this project.

References

- Abel, M. (2001). *Estudo da perícia em petrografia sedimentar e sua importância para a engenharia de conhecimento. Programa de PG em Ciência da Computação*, Porto Alegre: UFRGS, (pp. 239).
- Anjos, S. M. C. d., De Ros, L. F., Schiffer de Souza, R., de Assis Silva, C. M., & Sombra, C. L. (2000). Depositional and diagenetic controls on the reservoir quality of Lower Cretaceous Pendencia sandstones, Potiguar rift basin, Brazil. *AAPG Bulletin*, 84(11), 1719–1742.
- Benjamins, R. (2000). *Knowledge engineering and ontologies*. <http://www.swi.psy.uva.nl/usr/richard/home.html>.
- Bertino, E., Catania, B., & Zarri, G. P. (2001). *Intelligent database systems*. Reading: Addison Wesley.
- Conwy, P. (2002). *Petrographic data collection software*. Conwy Valley Systems Ltd, <http://www.petrog.com>.
- De Groot, A. D. (1965). *Thought and choice in chess*. The Hague: Mouton.
- Dickinson, W. R. (1970). Interpreting detrital modes of graywacke and arkose. *Journal of Sedimentary Petrology*, 40, 695–707.
- Fernández-Breis, J. T., & Martínez-Béjar, R. (2002). A cooperative framework for integrating ontologies. *International Journal of Human-Computer Studies*.
- Folk, R. L. (1974). *Petrology of sedimentary rocks*. Austin: Hemphills.
- Gaines, B. (1998). *Sisyphus III and IV knowledge-engineering studies. Proceedings of the 11th workshop on knowledge acquisition, modeling and management (KAW98), Alberta, Canada*, <http://ksi.cpsc.ucalgary.ca/KAW/KAW98/KAW98Proc.html>.
- Gappa, U., & Puppe, F. (1998). *A study of knowledge acquisition—experiences from the SISYPHUS III experiment for rock classification. Proceedings of 11th workshop on knowledge acquisition, modeling and management (KAW98), Banff, Alberta, Canada*.
- Garcia, A. C. B., Maciel, P. M., & Ferraz, I. N. (2000). *ADDGEO: an intelligent agent to assist geologist finding petroleum in offshore lands. International conference on industrial and engineering applications of artificial intelligence and expert systems (IEA/AIE 2000)*, New Orleans, Louisiana, USA: Springer.
- Garcia, A. J. V., Morad, S., De-Ros, L. F., & Al Aasm, I. S. (1998). *Palaeogeographical, palaeoclimatic and burial history controls on the diagenetic evolution of reservoir sandstones; evidence from the Lower Cretaceous Serraria sandstones in the Sergipe-Alagoas Basin, NE Brazil (Vol. 26). Carbonate cementation in sandstones; distribution patterns and geochemical evolution, Morad-Sadoon*, Oxford: Blackwell, (pp. 107–140).
- Gómez-Pérez, A., & Benjamins, V. R. (1999). Overview of knowledge sharing and reuse components: ontologies and problem-solving methods. In V. R. Benjamins, B. Chandrasekaran, A. Gomez-Perez, N. Guarino, & M. Uschold (Eds.), *International joint conference on artificial intelligence (IJCAI-99), workshop on ontologies and problem-solving methods (KRR5)*, Stockolm, Sweden, <http://sunsite.inforamtik.rwth-aachen.de/Publications/CEUR-WS/Vol-18/>.
- Gruber, T. (1992). *Ontolingua: a mechanism to support portable ontologies*. Stanford: Knowledge System Laboratory, Stanford University, 2002, <http://ksl-web.stanford.edu/KSL-91-66.html>.
- Guarino, N. (1995). Formal ontology, conceptual analysis and knowledge representation. *International Journal Human-Computer Studies*, 43(2/3), 625–640.
- Jansen, M. G., Schreiber, A. T. H., & Wielinga, B. J. (1998). *Rocky III—Round 1 a progress report. Proceedings of the 11th workshop on knowledge acquisition, modeling and management (KAW98), Banff, Alberta, Canada*, <http://ksi.cpsc.ucalgary.ca/KAW/KAW98/KAW98Proc.htm>.
- Leão, B. d. F. (1988). *Construção de uma base de conhecimento de um sistema especialista de apoio ao diagnóstico de cardiopatias congênitas. Pós-Graduação em Cardiologia da Escola Paulista de Medicina. São Paulo*, (pp. 230).
- Mattos, N. M. (1991). *An approach to knowledge base management*. Berlin: Springer.
- McBride, E. F. (1963). A classification of common sandstones. *Journal of Sedimentary Petrology*, 33, 664–669.
- Nonaka, I. (1994). The dynamic theory of organizational knowledge creation. *Organization Science*, 5, 14–37.
- Parsaye, K., Chignell, M., Khoshafian, S., & Wong, H. (1989). *Intelligent databases—object-oriented, deductive hypermedia technologies*. New York: Wiley.
- Schreiber, G., Akkermans, H., Anjewierden, A., Hoog, R. d., Shadbolt, N., Velde, W. v. d., & Wielinga, B. (1999). *Knowledge engineering and management—the CommonKADS methodology*. Cambridge: MIT Press.
- Silva, L. A. L. (2001). *Aplicando métodos de solução de problemas em tarefas de interpretação de rochas. Programa de PG em Ciência da Computação*. Porto Alegre: UFRGS.
- Spadafora, E., De Ros, L. F., Zuffa, G. G., Morad, S., & Al Aasm, I. S. (1998). *Diagenetic evolution of synorogenic hybrid and lithic arenites (Miocene), Northern Apennines, Italy (Vol. 26). Carbonate cementation in sandstones; distribution patterns and geochemical evolution, Morad-Sadoon*. Oxford: Blackwell, (pp. 241–260).
- Studer, R., Benjamins, V. R., & Fensel, D. (1998). Knowledge engineering: principles and methods. *Data and Knowledge Engineering*, 25(1/2), 161–197.
- VanLehn, K. (1989). Problem-solving and cognitive skill acquisition. In M. I. Posner (Ed.), *Foundations of cognitive science* (pp. 526–579). Cambridge: MIT Press.