SPIRIT: Integrating Knowledge-Based Techniques Into Well Test Interpretation [1]

I.W. Harrison, J.L. Fraser
Artificial Intelligence Applications Institute
The University of Edinburgh
Edinburgh EH1 1HN
Scotland, U.K.
email: aiai@ed.ac.uk

Abstract

This article describes how SPIRIT – a computer program for well test interpretation (WTI) – integrates knowledge-based techniques for pattern matching and decision modelling. Existing conventional WTI computer software assists reservoir engineers by removing some of the more mundane tasks as well as automating some of the manual interpretation techniques. However, the software generally provides no guidance to the user on the selection of an appropriate WTI model, which is a crucial stage in WTI. The goal of the SPIRIT project was to develop a prototype of the next generation of WTI software, that included a decision support tool for WTI model selection.

SPIRIT makes use of several different types of information for interpreting a well test: pressure, seismic, petrophysical, geological and engineering. SPIRIT's knowledge-based approach to type curve matching is to generate several different feasible interpretations by making assumptions about the possible presence of both wellbore storage and late time boundary effects. SPIRIT fuses information from the type curve matching and the other data sources, using a knowledge-based decision model developed in collaboration with a WTI expert.

These knowledge-based approaches were integrated with novel smoothing and segmentation algorithms as well as standard WTI presentation and analysis methods. The resulting prototype system has been judged to be a success by sponsors of the work.

Introduction

The motive behind the SPIRIT project was the need to improve the quality of well test interpretation (WTI) as it is carried out within oil companies. Reservoir engineers have access to existing WTI software that help in parameter estimation. However, in order to perform a full interpretation, an analytical model (called a WTI model) has to be selected, based on the well test pressure response data and the known geological and engineering

[1] This paper was written as a chapter for the book "Novel Applications of Artificial Intelligence in Petroleum", B. Braunschweig and R. Day (eds.), Editions Technip, (unpublished).

data about the well being tested. Selecting the most appropriate WTI model is a complex task, requiring significant expertise.

Various researchers had investigated the use of artificial intelligence (AI) techniques for the WTI model selection task [1, 2, 3, 4, 9]. However, this previous work had investigated using only the pressure response to help select a WTI model. Du [8] and Harrison [10] went further and developed a demonstrator WTI model selection system, that used *external* (geological, geophysical, petrophysical) data, as well as pressure data. The goal of the SPIRIT project was to build on this previous work and to develop a prototype of the next generation of WTI software, that would include a decision support tool for WTI model selection.

SPIRIT was a collaborative project between the Artificial Intelligence Applications Institute at The University of Edinburgh (AIAI) and the Department of Petroleum Engineering at Heriot-Watt University, and the project was managed overall by the Petroleum Science and Technology Institute (PSTI). The project began in January 1991 and was separated into three phases. The first phase was a design phase in which the requirements of potential users were elicited and the overall architecture and functionality of SPIRIT was decided upon. The second phase was a research phase in which the implementation of the modules that make up SPIRIT was explored, as were ideas concerning the user interface. At the end of this second phase a working demonstrator was produced, which gave sponsors a feel for the potential capability of the final prototype. The third phase of the project entailed agreement on the design of SPIRIT and the implementation of the final prototype. SPIRIT was completed in April 1993.

In this article the well test interpretation task is first described and some current difficulties with WTI are identified, as well as with the software support for WTI. The KADS methodology for developing knowledge-based systems is then introduced and it is shown how this was used for the SPIRIT project. Following on from this, the overall design of SPIRIT is given and the design of the knowledge-based modules, together with some related parts of the user interface, are detailed. Implementation issues are then addressed and finally the conclusions from the project are presented.

Well Test Interpretation

Overview

Reliable information about *in situ* reservoir conditions are important in many phases of reservoir engineering, *e.g.* to analyse and simulate reservoir performance. Pressure transient testing (commonly called well testing) can provide much of this information, and for many pieces of information is the most cost effective technique. Not only can well testing determine physical parameters about the reservoir rocks, such as permeability, but it can also be used to support information about the structure of the reservoir.

Well testing can be viewed as the application of a known signal, which is usually a perturbation to the oil flow rate, to an unknown oil reservoir, giving an output signal which is the corresponding change in pressure with time. Many different analytical (WTI) models have been derived to describe the flow of oil to a wellbore. The first, and most crucial stage in well test interpretation is to select the most appropriate WTI model.

Problems with WTI Model Selection

Each WTI model has a distinctive pressure response signature. However, identifying the correct WTI model to select given the known pressure transient test data, together with any other known geological and engineering data is not straightforward. The WTI model selection task has specific difficulties:

Non-uniqueness

Different WTI models can have the same pressure response, dependent on the reservoir parameter values. Hence the shape of the pressure response curve is not sufficient on its own to uniquely identify a WTI model. In addition the pressure data can be distorted by many different kinds of engineering phenomena.

Uncertainty in data

Much of the external (geological, geophysical and engineering) data that are used to select the WTI model are uncertain in some way. In addition there is uncertainty in the match between the theoretical response of a WTI model and the actual pressure test response, due to the problems of missing and noisy data.

Knowledge limited to a small range of reservoir types

Selecting the most appropriate WTI model requires significant amounts of expertise. Most interpreters only have experience of interpreting well tests in one or two reservoirs. There is a tendency for engineers to select a WTI model with which they are familiar and which fits some of the data, without fully exploring all the data in order to see whether other models are more appropriate.

Communication problems

The data necessary to correctly select the most appropriate WTI model come from a variety of sources and need to be integrated to produce a complete picture of the reservoir. This requires engineers, geologists, geophysicists and petrophysicists all communicating the relevant data to the well test interpreter and for the interpreter to fully understand the significance of the data. This is a difficult task, for which engineers are not trained.

The introduction of integrated, multi-disciplinary reservoir teams has helped to remove some of these problems, but there is still a perceived communication barrier between engineers and others.

Problems with Conventional WTI Software

The most common use of computers in WTI have been aimed at replicating the various manual techniques for interpreting well tests, such as the straight line and type curve techniques. The introduction of high resolution electronic bottomhole gauges has increased the quality of pressure test data significantly, such that it is now possible to detect small changes of pressure quite accurately. The introduction by Bourdet [6] of the pressure derivative type-curve has allowed the power of the pressure derivative to be recognised. Since the derivative enhances the differences in the shapes of the various flow regimes it enhances the diagnostic capability of the engineer. Even so, this type of software only aids the engineer by removing some of the mundane tasks of data preparation, graph plotting and parameter calculation. This results in a faster analysis but does not necessarily improve the analysis.

Many of the current software packages also include "black box" regression techniques, which when given a WTI model, can adjust the value of the model reservoir parameters until the shape of the actual and the model pressure responses match. The danger with such systems is that data can be force fitted to incorrect WTI models. This is especially true with some of the more complex WTI models which have up to 6 parameters. Expert interpreters would recognise this but non-experts might not, which would result in an erroneous interpretation.

Conventional WTI software, thus, is reliant on the correct WTI model being selected, in order that a correct analysis can be performed. However, conventional software provides no guidance in the selection of the WTI model. In addition, once a pressure transient test has been analysed, conventional software provides little in the way of guidance for the validation of the derived reservoir parameters, *e.g.* checking parameter values derived from other information sources.

Use of KADS Methodology in SPIRIT

KADS – A KBS Development Methodology

KADS is the name of a methodology for the development of knowledge based systems (KBS). KADS is the result of a long (and still on-going) ESPRIT sponsored collaborative project. KADS, although specifically a KBS development methodology, shares many features of the traditional software engineering approach, in that a rigorous analysis and requirements stage is carried out before any design decision is made, with implementation following design. Within KADS rapid prototyping is viewed as a technique to assist in particular aspects of development. The first stage in KADS is the analysis, which consists of the internal and external analysis of the system to be built [12]. Internal analysis in the KADS methodology is concerned with developing a model of expertise and a model of cooperation (modality), which together constitute the *conceptual model* of the task. External analysis in the KADS methodology is concerned with the specification of the external requirements and the constraints imposed by the working environment. Following on from the analysis stage, a functional and physical design can then be determined, and finally software can be implemented.

A library of generic models of expertise (called *interpretation models*) have been developed as part of the KADS project by abstracting out these models from analyses carried out on real problems. The first stage in KADS analysis is to select an interpretation model, or if none exists which describes the task then an interpretation model has to be developed. This interpretation model then acts as a focus for knowledge acquisition, to help develop the model of expertise.

KADS Internal Analysis for SPIRIT

The first stage in the analysis was to select or develop an interpretation model for the WTI model selection task, which could be used to focus knowledge elicitation. Early on it was felt that the task was fairly diagnostic in nature, but also seemed to contain synthetic elements, with regard to the way the unknown WTI model could be thought of as being configured from identifying it's parts (model components) and parameter values [11]. Various versions of the interpretation model were developed, with the main difference being in how the solution was developed – either incremental refinement of a set of hypotheses, or synthesis of data to produce a set of verified WTI models.

The main reason for the problem in developing the interpretation model for the WTI task is the fact that experts are not truly able to verbalise how they go about identifying WTI models. Many of their replies were post hoc justification of how they solved prototypical cases. Finally it was decided that the WTI task could best be described as a systematic diagnostic task, incorporating multiple diagnoses, whereby a set of hypotheses (the WTI models known to SPIRIT) can be ranked using both the shape of the well test pressure curve, as well as any known engineering and geological information. Then this set of competing models can be further verified, by gathering more engineering and geological data and by checking the models using the known pressure data (*e.g.* using various specialised plots of pressure against time).

KADS External Analysis for SPIRIT

External analysis in the KADS methodology is concerned with the specification of the external requirements of the KBS. The aim is look at the client's and users' requirements and the constraints imposed by the working environment. It also provides users with a set of expectations of what the new system will help them achieve.

Interviews were held with reservoir engineers and geoscientists at several oil companies, who were felt to be representative potential end users of SPIRIT. The goals were to find out how WTI was currently carried out; to find out where the main problems with the current WTI methodology were perceived to be; to identify potential solutions to these problems; to identify what kinds of computer software support they wanted for the WTI task.

From these interviews it was clear that there existed many problems in interpreting well tests. Some of the problems associated with WTI model selection were discussed earlier in this article. The major problems with respect to current WTI *software* were identified to be:

- No help in WTI model selection.
- No integration of external data in the overall WTI task.
- Inability to follow multiple lines of enquiry.
- Handling noise in data due to real reservoir effects.
- Unfriendly software – not geared for a wide range of skill levels.

Overall, the aim of the SPIRIT project was to design and develop a prototype of a new generation of WTI software; one that provided a decision support capability for WTI model selection, as well as showing how WTI software could be integrated with reservoir characterisation software. One of the main foci in the SPIRIT project was to tackle all of the problems, highlighted previously, associated with WTI model selection, through the integration of knowledge-based techniques with conventional WTI software functionality.

Summary

The use of the KADS methodology for the analysis stages of SPIRIT was viewed as a great success. The methodology gave a strong framework which helped in eliciting just what kind of system was required. The resulting documentation greatly aided communication within the project team and helped give focus for the design and implementation stages of the project. This was especially useful given the multi-site and multi-discipline makeup of the project team.

Overall System Design

From the initial project scoping it was clear that SPIRIT would have to be able to reason with uncertain information, from many different data sources, and be able to rank WTI models in terms of how well they matched the data. In addition, the system would have to be able to provide an explanation to the user of how it came to its conclusions.

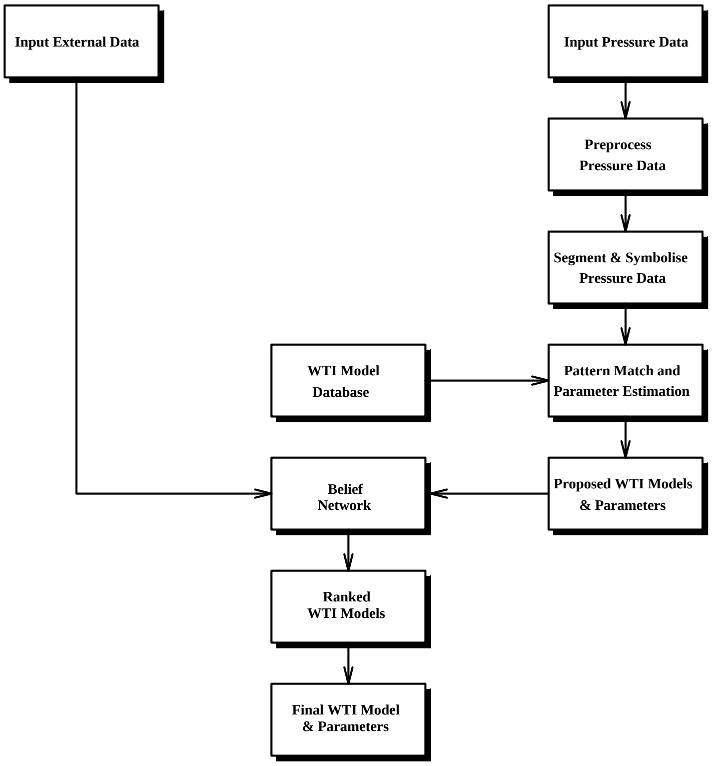

Figure 1: Overall design of SPIRIT tool

The overall design of SPIRIT was of two principal data sources feeding into a *belief network*, which modelled the expertise of a world expert in WTI model selection (see Figure 1). The user of SPIRIT interacts through a graphical user interface to enter pressure data or *external data* (engineering, geological, geophysical).

The steps in processing the pressure data are as follows:

- **Input pressure data** – similar to loading these data into conventional WTI software packages.
- **Pre-process pressure data** – filter the pressure data to extract a diagnostic curve (log-log plot of pressure logarithmic derivative).
- **Segment and symbolise pressure data** – translate the well test signal into a symbolic representation.
- **Pattern match and parameter estimation** – match the segmented and symbolised pressure data with a library of theoretical WTI model type curves and automatically compute model parameters from the well test data.

The output from the pattern matcher, together with the external data, feeds into a *belief network*. This belief network represents the WTI model identification task for a large set of WTI models. Inferences based on the input data are propagated through the network, and result in all WTI models in the network being assigned a level of belief. Figure 4 shows a tiny fraction of the full belief network in SPIRIT.

The output from SPIRIT is displayed in a window in the graphical user interface, and is a ranking of all the WTI models known to SPIRIT. For each of these models the following information is also available:

- Explanation of how the test pressure data match the model type curve.
- Well and reservoir parameters automatically calculated from the test pressure data, as well as collected from external sources.
- Explanation of how external data have supported the WTI model.

There is also the capability within SPIRIT for generating reports that include the data that were input to SPIRIT and the conclusions drawn (including the pattern match results) in PostScript format. SPIRIT's results can also be taken and used in conventional WTI software, *e.g.* for performing non-linear regression analysis on the parameter values determined within SPIRIT for a WTI model.

Pattern Matcher Design

The input data to the pattern match module is the output from the segmentation and symbolisation process (this process is described in [18]). The pattern match module in SPIRIT takes the segmented and symbolised transient pressure data from the well test and matches it against the theoretical responses of all the WTI models known to SPIRIT. Each WTI model has several "free" parameters which govern the shapes of its theoretical responses[2]. Each model therefore has several theoretical responses associated with it: each theoretical response corresponds to a "volume in parameter space", *i.e.* a set of parameter ranges within which a change in parameter value makes no change to the shape of the curve. Pattern matching thus requires the ability to judge whether the actual test response is close enough in shape to any of the many possible theoretical responses. The difficulty of the model selection task is compounded by the non-uniqueness of type curves: two quite different WTI models can have the same shape of theoretical response, depending on the parameter values.

The output from the pattern match module is a ranked set of possible interpretations of the well test pressure data. If the test response curve is deemed to be close enough to a theoretical response then the reservoir engineer needs to know two things: first of all, how close the match is and, secondly, which segments of the test curve correspond with which segments of the theoretical response. The main reason for this segment correspondence being needed is to indicate which portions of the test curve relate to the early-, middle-

[2]Note that the parameters referred to here are not "physical" parameters like permeability. They are "pseudo-parameters" like dimensionless wellbore length or dimensionless transmissibility.

and late-time regions (ETR, MTR and LTR), so that parameters like permeability and skin can be calculated with some confidence. Figure 2 shows an example of the output from the pattern matcher. In this figure the bottom left-hand window shows a log-log plot of the actual well test pressure data, together with the segmentation and symbolisation of the pressure logarithmic derivative; the top right-hand window shows the theoretical response for a selected WTI model; and the bottom right-hand window shows the explanation of how the actual pressure data matches the theoretical model response (WBS in this figure stands for wellbore storage).

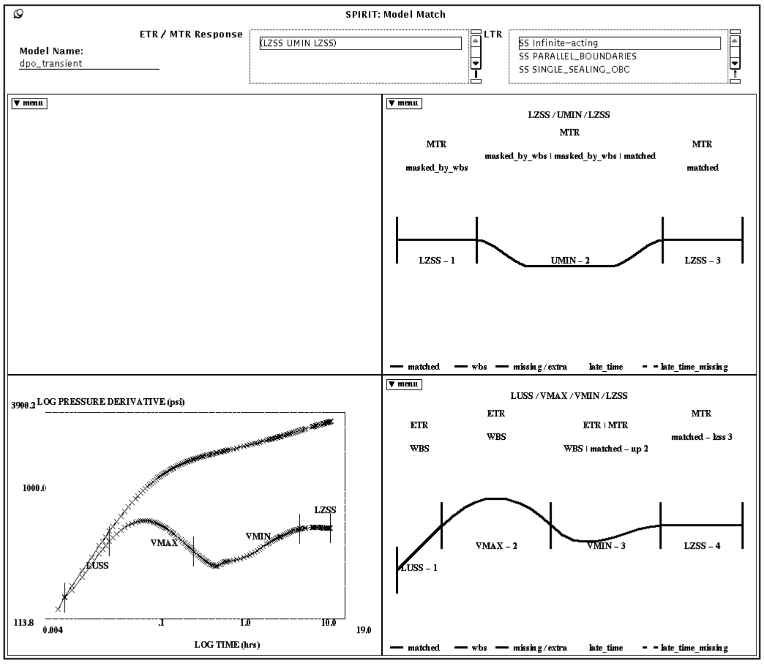

Figure 2: Pattern match results window

There are several further aspects to WTI model response matching which significantly increase the complexity of the problem addressed by the pattern matching module:

- Missing data – both at the beginning of the test (data not recorded for early part of test) and end of the test (test stopped too early).
- An engineering phenomenon called wellbore storage, if present, occurs at the start of the test curve and it may mask none, some or all of the segments present in the theoretical type curve response.
- Late time effects, due to outer boundaries, can mask some of the type curve segments.

- Uncertainty in segmentation of the test curve – the WTI engineer can under- or over-smooth noisy data, or select inconsistent sequences of segments, so the pattern matcher must infer corrections to these errors.

Belief Network Design

Belief Networks

Belief networks (Figure 3) represent the probabilistic dependency amongst a set of variables and have become an increasingly popular knowledge representation for uncertain reasoning [16]. Amongst other names given to belief networks are Bayesian (belief) networks, knowledge maps, probabilistic causal networks and qualitative probabilistic networks. The nodes in a belief network represent a random variable, or uncertain quantity, that can take two or more possible values. The arcs signify the existence of direct influences between the linked variables, and the strength of these influences are quantified by forward conditional probabilities. Calling an arc link an influence can be misleading. In general the links should be thought of as relevance links, but in other cases the links can be seen to have a causal meaning.

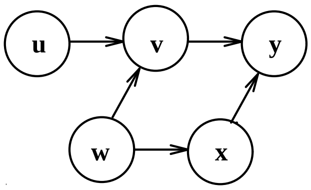

Figure 3: Example belief network

Within a belief network the basic computation is to calculate the belief of each node (the node's conditional probability) based on the evidence that has been observed. Various methods have been developed for evaluating node beliefs and for performing probabilistic inference. The most popular methods are due to Pearl [15] and Lauritzen and Spiegelhalter [13]. Similar techniques have been developed for constraint networks in the Dempster-Shafer formalism [17]. In addition to numerical representations of uncertainty other work has concentrated on non-numerical uncertainty handling, *e.g.* [5, 7, 14]. However, all these schemes are basically the same – they provide a mechanism to propagate uncertainty in the belief network, and a formalism to combine evidence to determine the belief in a node.

Belief Network for SPIRIT

As stated previously an important aspect of SPIRIT was the requirement for managing uncertainty. From speaking to the expert it became clear that much of the uncertainty in data was expressed in non-numerical terms, and the expert used symbolic terms to describe how likely a conclusion was based on a given set of data. From a review of the literature on AI techniques for reasoning with uncertain information, the conclusion reached was that the work of Cohen [7], with his MUM system, was the most applicable for use within SPIRIT. Cohen's work was further explored and a scheme for representing and reasoning with uncertainty within SPIRIT was derived. It was decided that within SPIRIT a symbolic representation of uncertainty would be adopted. Seven levels of belief were decided upon – *confirmed, strongly supported, supported, unknown, detracted, strongly detracted, rejected.*

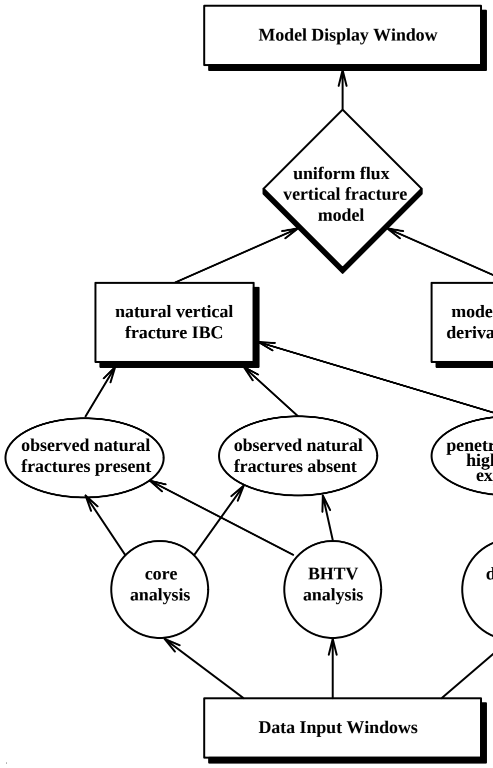

Figure 4: Schematic diagram of part of the belief network within SPIRIT

Based on MUM the WTI model selection task was represented as a belief network. Figure 4 shows a schematic view of how the belief network in SPIRIT works. At the bottom of the network are data nodes (circles) which relate to data that can be entered via the graphical user interface. Evidence nodes (ovals) represent the mapping of raw data or quantitative interpreted data into qualitative interpreted data. Cluster nodes (rectangles) are engineering or geologically significant groupings of evidence. Data nodes can support or weaken evidence nodes, which, in turn, can support or weaken cluster nodes. Cluster nodes can, in turn, support or weaken WTI model nodes (diamonds).

Propagation of inferences through the belief network comes about due to the fact that attached to each node is a combining function. A combining function is simply a function that calculates the level of belief of that node, given the levels of belief of its contributing nodes. In general there is a unique combining function for each node in the belief network. A combining function is fired when the level of belief in a connected lower node changes. Thus, as data are entered by the user, related evidence nodes will have their combining functions fired. This will in turn cause combining functions of cluster nodes linked to these evidence nodes to be fired and so on.

The result is that the level of belief in all the WTI models known to SPIRIT are constantly kept up to date, given the data that have been input. The WTI models are then ranked and displayed in the graphical user interface (see Figure 5). This window has three scrolling lists, displaying WTI models which have a level of belief of confirmed/strongly supported; supported; and other. The user can select any of these WTI models and can view why the model has the level of belief it does (Figure 6), as well as viewing how the WTI model matched the well test pressure data (Figure 2). In Figure 6, the levels of belief in a WTI model and the components of this model are displayed graphically using a thermometer-like analogy. A high level of belief is displayed as a long dark line stretching to the right.

External Data Interface Design

From discussions with reservoir engineers and geoscientists it became clear just what kinds of external data were of use when interpreting well tests. It also became clear that much of the necessary data was available either within existing geoscience software packages, such as seismic, mapping, or petrophysical packages, or was contained within company reports (available either in electronic or paper form).

For the external data interface of SPIRIT it was decided that it would be desirable to show how the next generation of WTI software could be integrated with such existing software packages. Hence it was decided to provide several different types of external data interface, which could act as a medium for dialogue between reservoir engineers and geoscientists.

Rather than trying to replicate the functionality of existing software packages, the external data user interfaces were designed to allow the output from other software packages to be displayed and manipulated for the particular needs of WTI. Four windows for the four main types of external data were designed:

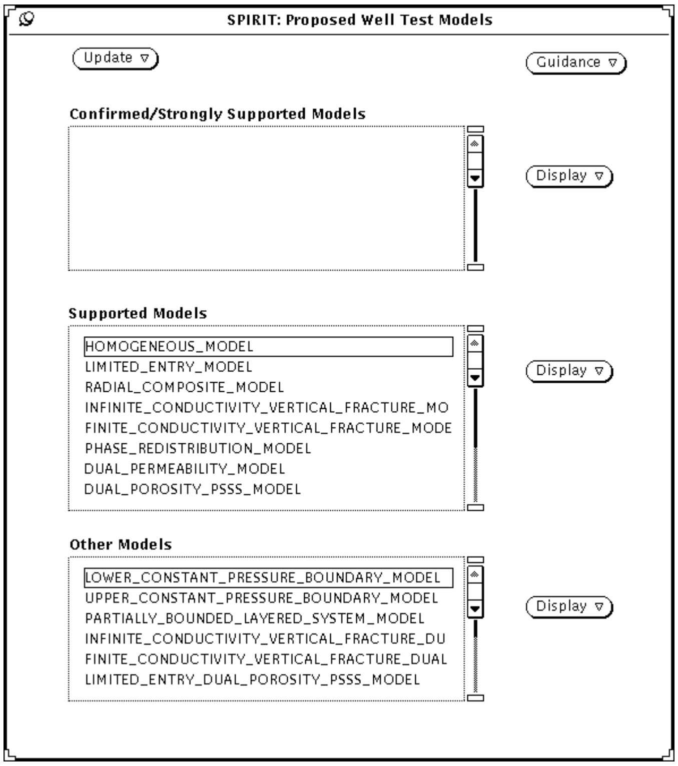

Figure 5: Ranked list of WTI models

Engineering data

These data would normally exist in various reports available to a reservoir engineer. These data include near wellbore information, such as perforation, drilling, completion and stimulation data, together with production history and production drive mechanisms. The user interface allows an engineer to select from the possible values for such data (including the value “unknown”).

Geology data

These are quite high level geological data and a reservoir geologist should be able to give a reservoir engineer the data. These data include structural fabric and depositional environment data. Again, like the engineering data interface, an engineer can select from the possible values for such data.

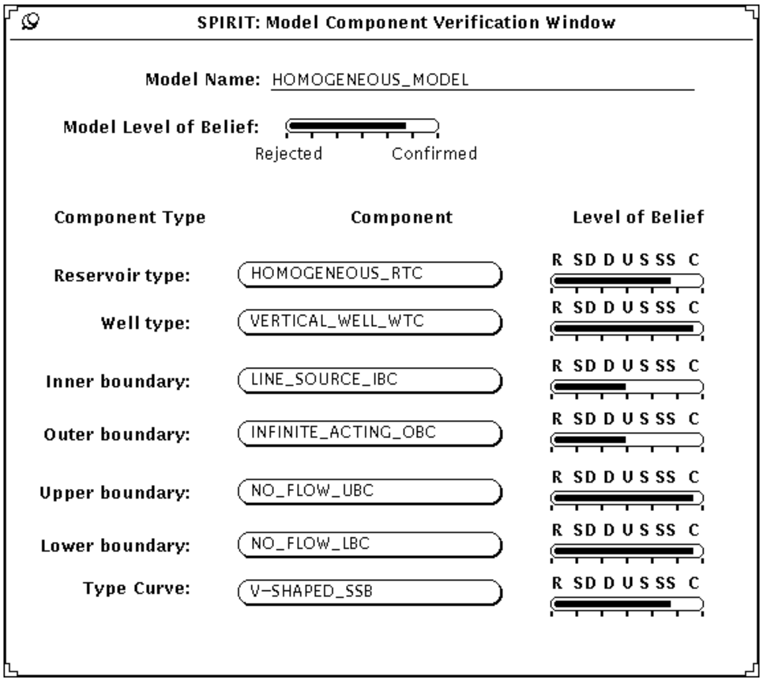

Figure 6: Explanation for level of belief in a WTI model

Structure data

Top structure contour data, fault data and well location data can be output from existing mapping packages in ASCII format. SPIRIT can read such data in and display them. A reservoir engineer, together with a geologist, can then look at these data and tell SPIRIT what the likely outer boundary conditions are for the well being tested. Figure 7 shows the structure data window.

Zonation data

Petrophysical log packages can output interpreted log data (such as volume of shale, porosity, water saturation, PLT and RFT). The zonation data window can display these data, together with other data such as perforation intervals and core permeability. An engineer can then, with the help of a geologist, zone up the reservoir in terms of the significant flow units for the reservoir being tested. For each of these flow units, the user can add interpreted data about things such as flow unit continuity. In addition, various parameters such as flow unit perforation and average flow unit permeability can be automatically calculated. Figure 8 shows the zonation data window.

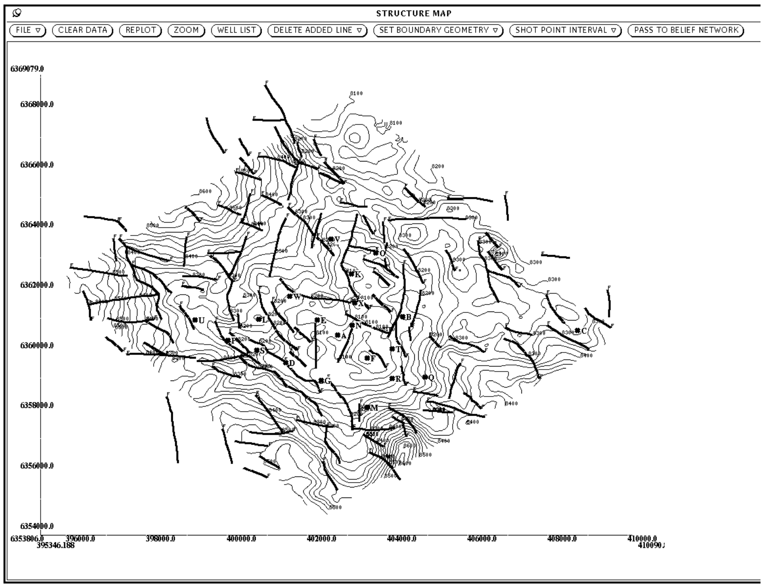

Figure 7: Structure data window

WTI Model Browser Design

As part of the SPIRIT project a database of WTI models was developed. In order that the data on WTI models could be made available to users of SPIRIT a browser window for the database was designed. As well as allowing users to browse the database to learn more about a WTI model of interest, users could also add new models to this database (although currently these models would not then automatically be known to the WTI model selection decision support tool). The WTI model database, as well as being available on-line from windows within SPIRIT, would also be accessible outside of SPIRIT. This means that the SPIRIT prototype has potential as a training aid for learning about WTI models.

System Implementation

User Interface

SPIRIT was designed to be an X Windows application for use on Unix workstations. The user interface of SPIRIT was implemented using the OpenLook toolkit XView 3.0. The user interface was developed incrementally, using the OpenWindows Developers Guide

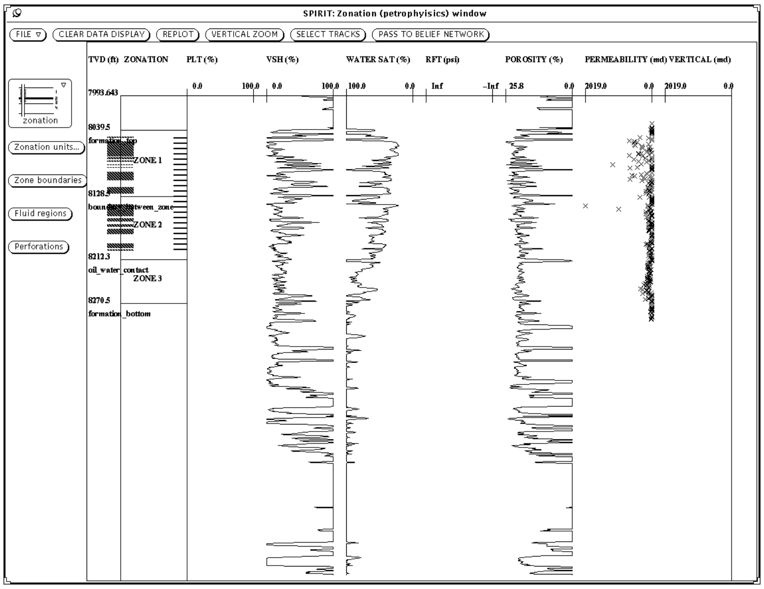

Figure 8: Zonation data window

(DEVGUIDE) to produce mock-ups. DEVGUIDE proved to an extremely useful tool, as mock-ups could be developed quickly and were used to generate discussion and ideas within the project team as well as with the WTI expert. In addition, once the mock-up user interfaces became mature, DEVGUIDE could generate the XView code for the user interface windows automatically, leaving the project team to code only the system callbacks.

Knowledge-based Modules

The pattern match module was developed using COMMON LISP, and the WTI model selection belief network was initially developed using KEE[3] – an AI toolkit. The powerful debugging and object browsing facilities of the toolkit proved to be extremely useful during early prototype development, but it became clear from sponsors that it would be desirable for the final deliverable software not to be reliant on KEE. Hence it was decided that the final deliverable knowledge-based component of SPIRIT would be implemented using purely COMMON LISP and CLOS[4] code.

[3] KEE (Knowledge Engineering Environment) is an advanced software environment for rapid development of KBS. It is produced by IntelliCorp Inc.

[4] CLOS is the object system that is part of COMMON LISP

Knowledge acquisition sessions with the expert were arranged, but problems in getting sufficient time with the expert meant that the first demonstrator of SPIRIT did not have much specialist WTI model selection knowledge built into it. In addition the complex nature of the WTI model selection task meant that knowledge acquisition took longer than expected.

The belief network was developed using the hyper-diagramming tool HARDY[5]. HARDY greatly helped knowledge capture by allowing a graphical representation of the belief network to be interactively developed by the knowledge engineer in collaboration with the WTI expert, reviewed by the expert and modified as necessary. The expert found the belief network representation scheme to be very intuitive. In addition to being a useful knowledge capture tool, HARDY can automatically generate KEE or CLOS code for the network, with only the combining functions for each node in the network having to be hand written in CLOS. Using HARDY to capture the nodes and arcs of the belief network in an intermediate representation meant that it was easy to move eventually from KEE to CLOS. The KEE code for the belief network, generated from HARDY, together with the combining function code, was tested using KEE. Once the belief network was working correctly, the finished network was then output from HARDY as CLOS code, and loaded into SPIRIT, together with the combining function code.

System Testing

Sponsor companies provided various field data sets which were used to test various parts of the system. The expert also went through several other real test cases of his own, in order to test the pattern matcher and the belief network further. SPIRIT was found generally to agree with his expectations. The final prototype of SPIRIT was demonstrated to all the sponsor companies at several workshops, allowing the project team to get feedback. Sponsors were pleased with SPIRIT and have been keen to start fully evaluating it.

Conclusions

SPIRIT was designed to be a prototype of a new generation of well test interpretation (WTI) software, showing how WTI could be integrated with reservoir characterisation software. Rather than replicating much of the functionality of existing geoscience software packages, the emphasis was on handling output from these packages and providing sufficient functionality within SPIRIT to demonstrate how these data could be displayed and manipulated for the particular needs of WTI. Using such data SPIRIT can, in conjunction with conventional WTI software, provide a more complete interpretation of a well test.

Major new features of SPIRIT over conventional WTI software include:

[5] HARDY is a visual modelling tool developed by AIAI.

- A knowledge-based decision support tool for WTI model selection, integrated with some conventional WTI software functionality;
- Reduction of the non-uniqueness problem, while ensuring all possibilities are considered;
- Integration of external geological and engineering information with pressure data;
- Semi-automation of pattern recognition in the pressure data;
- A database of WTI models – accessible for a variety of uses;
- Various methods for converting build-up and variable rate drawdown tests into their constant rate drawdown equivalents;
- Potential for SPIRIT to be used as a WTI training tool;
- Potential for SPIRIT to be used to carry out some well test design calculations.

The SPIRIT project was also judged to be a success by the project team and produced some valuable conclusions about developing knowledge-based systems (KBS). The KADS methodology was found to be extremely useful in the analysis stages of KBS development, as it provided a clear framework for knowledge acquisition. Devguide proved to be a very useful tool for developing graphical user interface (GUI) mock-ups and speeded up the overall implementation time of SPIRIT's GUI.

Belief networks were shown to be an intuitive way of representing and reasoning with uncertainty and HARDY proved to be an excellent tool for the belief network knowledge acquisition. Hardy enabled the model of expertise to be captured in a graphical manner such that it was understandable to the expert, in a way which rules alone could not be. In addition the network was reviewed and modified interactively with the expert and the output was used directly in SPIRIT.

Spirit has been demonstrated to project sponsors, who have judged it to be a success. Spirit meets their original requirements and is viewed as a valuable decision support tool for WTI model selection. Sponsors will soon take delivery of the software and begin in-house evaluation.

Acknowledgments

The authors would like to acknowledge the contributions of other colleagues at AIAI, in particular Julian Smart for his help and expertise in tailoring HARDY for use on SPIRIT. In addition, the authors would like to acknowledge the contribution of the other SPIRIT team members, based at Heriot-Watt University: Kuifu Du, Alan Bennett and Andrew Exton, together with Prof. George Stewart, who provided the WTI expertise for SPIRIT. Thanks also go to the project sponsors: The Petroleum Science and Technology Institute (PSTI) who also acted as project manager; Amoco; Bow Valley, Elf-Enterprise Caledonia, Enterprise Oil; Shell Expro.

References

[1] A.U. Al-Kaabi and W.J. Lee. Using artificial neural networks to identify the well test interpretation model. In *5th SPE Petroleum Computer Conference*, 1990. SPE Paper 20332.

[2] A.U. Al-Kaabi, D.A. McVay, S.A. Holditch, and W.J. Lee. Using an expert system to identify the well test interpretation model. In *63rd Annual Technical Conference and Exhibition of the Society of Petroleum Engineers*, 1988. SPE Paper 18158.

[3] O. Allain and O.P. Houze. A practical artificial intelligence application in well test interpretation. In *European Petroleum Computer Conference*, 1992. SPE Paper 24287.

[4] O.F. Allain and R.N. Horne. Use of artificial intelligence for model identification and parameter estimation in well test interpretation. In *63rd Annual Technical Conference and Exhibition of the Society of Petroleum Engineers*, 1988. SPE Paper 18160.

[5] Z. An, D.A. Bell, and J.G. Hughes. Res: A relative method for evidential reasoning. In D. Dubois et al., editor, *8th Conference on Uncertainty in Artificial Intelligence*, pages 1–8. Morgan Kaufmann, San Mateo, California, 1992.

[6] D. Bourdet, T.M. Whittle, A. Douglas, and Y.M. Pirard. A new set of type curves simplifies well test analysis. *World Oil*, pages 95–106, 1983.

[7] P.R. Cohen, D. Day, J.D. Lisio, M. Greeberg, R. Kjeldsen, D. Suthers, and P. Berman. Management of uncertainty in medicine. *International Journal of Approximate Reasoning*, 1:103–116, 1987.

[8] K.F. Du. *Knowledge Based Well Test Interpretation.* PhD thesis, Department of Petroleum Engineering, Heriot-Watt University, Scotland, 1990.

[9] S.S. Guedes, E.A. Campagnolo, and C.R. Souza. An expert system for model identification in well test interpretation. In *Conference on Artificial Intelligence in Petroleum Exploration and Production*, pages 43–50, 1989.

[10] I.W. Harrison. Development of a knowledge based system for well test model identification. Master's thesis, Department of Artificial Intelligence, The University of Edinburgh, Scotland, 1990.

[11] I.W. Harrison and J.L. Fraser. Pre-selecting well test interpretation models using external data. In *European Conference on Artificial Intelligence in Petroleum Exploration and Production*, 1991.

[12] F.R. Hickman, J.L. Killin, L. Land, T. Mulhall, D. Porter, and R.M. Taylor. *Analysis for Knowledge-Based Systems.* Ellis Horwood, Chichester, 1989.

[13] S.L. Lauritzen and D.J. Spiegelhalter. Local computations with probabilities on graphical structures and their applications to expert systems. In G. Shafer and J. Pearl, editors, *Readings in Uncertain Reasoning*, pages 415–448. Morgan Kaufmann, San Mateo, California, 1990.

[14] T.Y. Leong. Representation requirements for supporting decision model formulation. In B.D. D'Ambrosio et al., editor, *7th Conference on Uncertainty in Artificial Intelligence*, pages 212–219. Morgan Kaufmann, San Mateo, California, 1991.

[15] J. Pearl. Fusion, propagation and structuring in belief networks. *Artificial Intelligence*, 29(3):241–288, 1986.

[16] R.D. Shachter, B. D'Ambrosio, and B.A. Del Favero. Symbolic probabilistic inference in belief networks. In *AAAI-90*, volume 1, pages 126–131, 1990.

[17] G. Shafer. Perspectives on the theory and practice of belief functions. *Int. J. of Approximate Reasoning*, 4:323–362, 1990.

[18] G. Stewart and K.F. Du. Feature selection and extraction for well test interpretation by an artificial intelligence approach. In *64th Annual Technical Conference and Exhibition of the Society of Petroleum Engineers*, 1989. SPE Paper 19820.