



## Editorial: special issue on verification and validation

ROBERT PLANT

*Department of Computer Information Systems, University of Miami, Coral Gables, FL 33124, USA. email: rplant@umiami.ir.miami.edu*

ALUN D. PREECE†

*University of Aberdeen, Computing Science Department, Aberdeen AB9 2UE, Scotland. email: apreece@csd.abdn.ac.uk*

As for any software, users of knowledge-based systems (KBS) need to know that they can rely on the system to do its job properly. Assuring the reliability of knowledge-based systems has become an important issue in the development of the knowledge engineering discipline. The processes employed directly to assure the reliability of software are called verification and validation (V & V). Roughly speaking, *validation* is the process of determining if a KBS meets its users' requirements; *verification* is the process of determining if a KBS has been constructed to comply with certain formally-specified properties, such as consistency and irredundancy. Implicitly, validation includes verification.

Verification and validation techniques for KBS have been discussed and debated in workshops at many of the predominant artificial intelligence conferences in recent years. The purpose of this special issue is to provide "snapshots" of the current state of the V & V area for KBS, by collecting together representative works from three of the most recent workshops:

- at IJCAI-93 in Chambéry, France (Chairman: Marc Ayel, Université de Savoie, France);
- at AAAI-94 in Seattle, USA (Chairman: Robert Plant, co-editor of this issue);
- at ECAI-94 in Amsterdam, The Netherlands (Chairman: Alun Preece, co-editor of this issue).

These workshops succeeded in highlighting many of the significant issues and trends within their area of concern. These issues and trends are reflected in the articles selected for this issue, the authors of which have expanded and updated their original workshop papers. The purpose of this introduction is to highlight some of the issues and trends in KBS V & V, to put this collection in its context.

© 1996 Academic Press Limited

### 1. KBS verification and testing

In recent years, the main technological theme in the KBS validation area has been the development of tools for automatic verification of knowledge bases. Within this sub-area, the dominant concern has been with the "first generation" type of rule-based systems, and the verification has been aimed at detecting *anomalies*—such as subsumed or conflicting rules—which are symptomatic of logical faults in the knowledge base. The paper by Murrell and Plant offers a mature view of this kind of verification, wherein the rule base is modelled using graph theory, and the anomalies are defined in graph-theoretic terms.

† Editor to whom all correspondence should be sent.

A second “classical” area of concern in KBS V & V has been the testing of a knowledge-based system using test cases. Among the chief difficulties in testing KBS are ensuring that the coverage of the testing is sufficiently thorough, and determining the acceptability of the system’s response to the tests. The paper by Preece *et al.* addresses both these topics, employing a structural model which can be used to assess test coverage, and presenting a number of experimental methods with which to assess the acceptability of the system’s performance on the tests.

The foundations of the research in validation and verification primarily focused on the classification and formalization of testing techniques that could be applied to knowledge-based systems. This area led O’Leary to consider a further problem: what is the relationship between knowledge-based system size and the number of errors in the knowledge base, as well as the number and existence of different kinds of errors. The correlations between these relationships are presented in his paper.

## 2. KBS V & V with formal methods

One of the most significant issues currently facing researchers in the KBS V & V area is of how to exploit work done in the area of formal specification languages. In recent years several languages have been developed which allow knowledge engineers to create an initial, formal description of a KBS, and progressively refine it towards an implementation. Clearly, formal specifications can play a significant role in KBS V & V.

The paper by van Harmelen and Aben demonstrates how the (ML)<sup>2</sup> specification language can be used to bridge the gap between an informal description of a KBS, and a completely formal description, in such a way as to provide a degree of assurance that the formal description reliably captures the users’ requirements. In this way, validation is “built-into” a rigorous process of specification and refinement, so that the final product will have been validated without requiring testing.

In contrast to the previous paper, which uses a specification language developed specifically for KBS, Gamble uses specification techniques derived from conventional software engineering. Gamble considers the problem of building hybrid knowledge-based systems that combine object-oriented, frame-based, and rule-based paradigms, presenting a methodology for prototyping these hybrid systems from a partial specification expressed in a formal language.

An open question arising from these works is that of whether languages designed specifically for KBS are more appropriate than “wide spectrum” specification languages from software engineering, for supporting KBS V & V.

## 3. V & V and KBS refinement

Reflecting the fact that V & V are merely sub-processes in the overall KBS development processes, a growing number of researchers are concerned with using the results of V & V techniques to “drive” subsequent KBS refinement and improvement. The articles by Craw and Ho Kang *et al.* offer different perspectives on how this may be done with assistance from automatic tools.

Craw’s approach is to couple the output of “traditional” V & V tools—such as an anomaly detector—to a machine learning (ML) system. The ML system employed,

KRUST, uses the anomalies to localise faults in the KB, and then makes use of test cases to propose knowledge base refinements to remove the anomalies. The process is designed to be interactive, to assist a human knowledge engineer.

The technique described by Ho Kang *et al.* uses a KBS maintenance technique, *Ripple-Down Rules*, to allow a KBS to be refined incrementally when faulty behaviour is discovered on test cases. The perspective on V & V underlying this approach is that, rather than seeking to eradicate faults prior to fielding a KBS, the system should be given a mechanism whereby unacceptable performance can be detected and rectified “on the fly”.

Taken together, we believe that this collection of papers provides an accurate and illuminating picture of the current state-of-the-art in KBS verification and validation. We would like to thank all the authors for their considerable work in revising and expanding their papers, all of the reviewers, and the attendees of the original workshops, for stimulating the growth and maturing of the V & V area.