# The Visual Development of Rule-Based Systems

By Charles Langley and Clive Spenser

## Introduction

In the late 1980's Knowledge Based Systems (KBS) were seen to be leading edge software technology. Developers thought that the simplest KBS paradigm, Expert Systems, perhaps combined with probabilistic and fuzzy logic extensions would soon revolutionise the way that software was used throughout business and other sectors of the economy.

KBS software was built on rules which encoded the knowledge of experts in any given domain. Computers would then use this encoded knowledge to make decisions on behalf of their human users.

It was not long however, before the bubble of hype surrounding these systems began to burst. Something was wrong, but what was it?



## The Knowledge Acquisition Bottleneck

Apart from the limited power of the computers available at the time, the major problem was the difficulty of acquiring implicit knowledge from the minds of experts and then representing it explicitly. This so-called Knowledge Acquisition Bottleneck was believed to be the limiting factor on building systems that could do complex, useful tasks.

By the end of the twentieth century however, university departments were working hard at this problem. Curiously it was often Psychology departments rather than Computer Science departments which had the most impact in this area.

In particular, Ethnography (by then seen as a core part of Cognitive Psychology) was being used to study behaviour in situ with the aim of identifying the cognitive processes underlying that behaviour. Just as Margaret Mead (an early ethnographer) had lived amongst native tribes in Papua New Guinea in order to study their cognitive behaviour, so Psychology departments were sending researchers (often under cover) into workplace environments to discover how people approached problem-solving activities.

This work was, and continues to be, very successful. Knowledge acquisition is no longer the 'black art' it was deemed to be. Despite this, KBS has continued to be underused. Why might this be?

## A Knowledge Representation Bottleneck?

It is my contention that the problem was not primarily with how we obtained knowledge, but with how we represented it. I am not arguing that rules (or Bayesian networks and other knowledge representation methods) are inadequate to the

task, but rather that it is the way in which these rules and other representational formalisms are themselves represented that is the limiting factor.

At first a simple rule-base is relatively transparent, especially if properly documented. Certainly such systems were easier to comprehend than procedural code and were subsequently easier to update and amend. As such rule bases became larger and more complex however, a simple syntax error, perhaps only involving one word, could prevent them from operating correctly. The complexity of these rulesets also meant that it was difficult to get an overview of what was intended, thus impeding their maintenance and extension.

## A Picture is Worth a Thousand Words

The problem of rulebase comprehensibility, I would argue, is the fact that we have primarily represented knowledge using text based structures rather than visual ones. No matter how close to natural language a knowledge representation language is, you cannot see at a glance what a complex system is trying to do.

## Visual Rule Generation

Rule generation via a graphical interface is a hot topic right now, with offerings from a number of companies small and large. This is being driven in part by current interest in so-called 'business rules management' which is arguably a reawakening of the KBS paradigm we mentioned earlier.

London based Logic Programming Associates is an appropriate company to enter this market as it has been producing rule-based software since the mid 1980s. Its latest product, VisiRule, enables rule-based systems to be automatically generated from a flowchart drawn on the screen.

Consider the following business rule (Ross 2003):

Rule: An order must be credit-checked if any of the following is true:

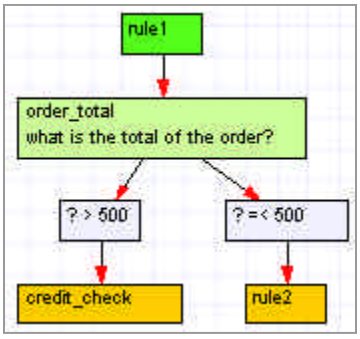


**Figure 1**

* The order total is more than $500
* The outstanding balance of the customer's account plus the order amount is more than $600
* The customer's account is not older than 30 days
* The customer's account is inactive
* The customer is out of state