Project: TECHNOLOGY PROJECTS

Title: Proposal for Research in High Assurance Systems

Ref: DS/FMU/RBJ/171 *Issue:* *Date:* 1990-02-26

Status: Informal *Type:* Proposal

Authors:

Name	*Location*	*Signature*	*Date*
R B Jones	WIN01		
R Stokes	WIN01		

Abstract: This paper presents a proposal for research into issues relating to the development of systems which can be relied upon to very high degrees of assurance to satisfy certain identified critical requirements.

Distribution:

0 DOCUMENT CONTROL

0.1 Contents list

0.2 Document History

Issue: , Date: , Status: Informal

This is the first externally issued draft.

0.3 Approval Authority

This document is not subject to formal approval.

0.4 Changes Forecast

This document is expected to be significantly revised as a result of further discussion both in substance and in form.

0.5 References

[CESG 87] CESG *Handbook of Computer Security Evaluation* CESG Computer Security Memorandum No. 2, 1987

[CESG 89] CESG *UK Systems Security Confidence Levels* CESG Computer Security Memorandum No. 3, 1989

[TCSEC] DoD *Trusted Computer System Evaluation Criteria* DoD 5200.28-STD 1985

[Gass] Gasser M *Building A Secure Computer System* Van Nostrand 1988

[Gog] Goguen J A and Meseguer J *Security Policies and Security Models* Proc 1982 Symposium on Security and Privacy, IEEE 1982

[MJCG] Gordon M J C *HOL: A Proof Generating System for Higher Order Logic* in: VLSI Specification, Verification and Synthesis, Kluwer 1988, eds Birtwistle and Subrahmanyam.

[HOL] Gordon M J C et. al. *The HOL System - DESCRIPTION* version 1 (for HOL88.1.10) December 4 1989, available from SRI International (at Cambridge) or Cambridge University Computer Labs.

[BJH027] Homer B J *A Definition of Security Based on Classification* ICL DS/FMU/BJH/027 1988

[BJH035] Homer B J *Unwinding the Classification Property* ICL DS/FMU/SM/035 1988

[CBJ 80] Jones C B *Software Development A Rigorous Approach* Prentice-Hall 1980

[CBJ 86] Jones C B *Systematic Software Development Using VDM* Prentice-Hall 1986

[Spi 88] Spivey J M *Understanding Z* Cambridge U P 1988

[Spi 89] Spivey J M *The Z Notation A Reference Manual* Prentice Hall 1989

[Wor] Wordsworth J B *A Z Development Method* IBM UK 1987

1 GENERAL

1.1 Scope

This document addresses the development of high assurance systems. In principle the document may be relevant to a very broad class of system developments, but the approach to the subject is through studies of specific classes of system and through relatively specific development methods. The reason for approaching this very general subject through rather more specific studies is simply that there is as yet insufficient experience in developments and methods to regard a less specific study as likely to bear fruit.

The application domain to be used as exemplar is the development of secure computer systems. Methods which will receive particular attention involve the use of Higher Order Logic and the proof development and proof checking tool HOL, as means of specifying precisely what requirement is to be satisfied by the system, and as a general mathematical modelling language with which models of proposed solutions can be shown mathematically to meet the identified requirements.

Notwithstanding the focus on these specific problem domains and development techniques, the scope of the document and of the proposed research remains very broad.

There are within its scope:

- philosophical problems - about what *high assurance* is and how it is to be obtained
- metamathematical problems - about the meaning of statements in formal languages and the demonstration of the soundness of rules of reasoning for such languages
- secondary verification problems - concerning our assurance of the correctness of the tools used to develop and check proofs, and of other tools which may effect our confidence that any theorems supposedly proven and checked by machine are indeed theorems of the relevant formal system

- application problems - which include the question whether the propositions held to assert conformance of the implemented system to the identified requirements do indeed assert this

The problem is in some ways like a hologram. When a part of the problem is broken off, on closer examination there may unfold within this part the entirety of the original problem. One sub-problem is that of obtaining highest levels of assurance that theorems accepted by a proof checker as proven are valid. This is the 'high assurance' problem in a different problem domain (proof development system), but one which shares a useful number of characteristics with the original problem domain (secure systems).

1.2 Introduction

2 BACKGROUND

ICL collaborates with the University of Kent and other partners in a research programme partly funded by the DTI and SERC entitled 'Foundations and Tools for Formal Verification' (IED proposal 1563). This project extends over a period of three years from 1st January 1990.

The applicant is overall manager for this research project and expects to make a substantial contribution to the research effort. The research described below will be conducted primarily under the auspices of this project, and the researcher will therefore have at his disposal all the necessary machine resources. Appropriate library facilities will be available on site, supplemented by access to the library at Reading University. Materials not accessible directly in these libraries may be acquired through inter-library loan, or by personal visits to the British Library, within reasonable travelling distance of ICL Winnersh.

Though registrartion is applied for as a part time student, teh applicants previous work in this field together with the fact that most of the research will be conducted during normal working hours mean that more rapid progress may be expected than would otherwise be the case. It is hoped that it may therefore be possible to make a case for permitting submission of a dissertation for the degree of PhD in the minimum exceptionally permitted period of 3 years from registration.

3 AREA OF RESEARCH

It is now generally recognised that formal methods have a role to play in the development of computer systems. Where particularly high levels of assurance are sought that a computer system meets certain critical requirements the use of formal methods may be mandated, and formal machine-checked proofs may be required.

Machine checked proofs add to the assurance of correctness of the system under development only if some confidence can be placed in the correctness of the computer system which is undertaking the proof checking. Most such proof checking systems have originated in University research, and the evidence for their correctness is at best the observation that no means of proving false claims has been discovered for some (generally quite modest) period of time since the last such means was eliminated.

The proposed research will address some aspects of the problem of obtaining the highest achievable

levels of assurance that a proof development system meets its critical requirement, viz: that it never accepts as proven a proposition which is not in some appropriate sense "true".

Such proof development systems are at risk of error in two quite disctinct ways. Firstly the may accept as proven invalid propositions because the logical system which it was inteded to implement was in fact unsound. Secondly they may accept as proven an invalid proposition despite the soundness of the logical system, because the implementation is incorrect and does not in fact implement the intended system.

It is therefore necessary to establish for such a system:

- Whether the logic supported is 'consistent'
- Whether that logic is correctly implemented in the proof checker.

For highest levels of assurance these proofs would be expected to be conducted within some formal system with good tool support for machine checking of formal proofs, e.g. HOL, VERITAS+.

The proposed research will address both the above subproblems, with a view to providing for the target system, formal proof that all alleged theorems proven using the supporting tool are 'valid'.

To have success in such an endeavour, given the tendency of formal proofs to prove large and time consuming, it will be necessary to chose a formal system which is economic in its primitive structures, and an implementation technique which permits careful separation of the critical from non-critical code. The HOL logic implemented using the LCF technique of defining an abstract data type is ideal for such an investigation. The HOL language would itself be used as a metalanguage in which all specifications would be written (though versions in Z might also be prepared). The implementation of the inference engine would be undertaken in SML and its verification would be attempted using techniques for embedding programming languages in HOL developed from suggestions by MJCG.

If reasonable success attends the attempt to treat the HOL system in this way then more difficult languages might be attempted. The primary contenders for these will be Z and VERITAS. The reasons for these two choices are, in the case of Z that the language is more widely used as a specification language than HOL, but lacks effective proof tools. In the case of VERITAS+, that though an effective proof tool is available, the meta-theory is as yet incomplete, and hence assurance of consistency could be improved. The primary merit of VERITAS relative to the two other languages mentioned is its provision of 'dependent types'.