SOFTWARE FOR SAFETY CRITICAL APPLICATIONS

Martin Kropík, Karel Matějka, Monika Juřičková, Robert Chudý
Czech Technical University
Faculty of Nuclear Sciences and Physical Engineering
Department of Nuclear Reactors
V Holešovičkách 2, CZ180 00 Prague 8, Czech Republic
kropik@troja.fjfi.cvut.cz

ABSTRACT

The contribution gives an overview of the project of the software development for safety critical applications. This project has been carried out since 1997. The principal goal of the project was to establish a research laboratory for the development of the software with the highest requirements for quality and reliability. This laboratory was established at the department, equipped with proper hardware and software to support software development. A research team of predominantly young researchers for software development was created. The activities of the research team started with studying and proposing the software development methodology. In addition, this methodology was applied to the real software development. The verification and validation process followed the software development. The validation system for the integrated hardware and software tests was brought into being and its control software was developed. The quality of the software tools was also observed, and the SOSAT tool was used during these activities. National and international contacts were established and maintained during the project solution.

1 INTRODUCTION

The paper deals with the description of activities during the solution of the project of the software development for safety critical applications. This project, financially supported by the Czech Ministry of Education, was carried out at the Department of Nuclear Reactors, Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University in Prague during the period from 1997 to 2000. The principal goal of the project was to establish a research laboratory for the development of highly reliable software. This laboratory had to be equipped with proper hardware and software to support software development. Furthermore, a research team of predominantly young researchers for software development had to be created. The research activities were primarily focused on software for the safety systems of nuclear devices, but the investigated methodology of the highly reliable software production will also able be used in other branches of software utilization with high quality and reliability demands (medicine, transport, military, etc.).

The further goals of the project were to integrate undergraduates and PhD students of the department into the carrying out the project, to incorporate new lectures dealing with the quality assurance of the software into the educational process at the department and faculty, to establish and maintain the national and international co-operation and to publish the results of the project.
2 THE LABORATORY FOR THE DEVELOPMENT OF HIGHLY RELIABLE SOFTWARE

The laboratory for the development of highly reliable software was established at the Department of Nuclear Reactors. The department granted convenient rooms for the laboratory. The rooms were rebuilt to fulfil the new laboratory requirements. The laboratory was equipped with proper hardware and software for the project solution.

2.1 The computer network

The laboratory was equipped with a separate computer network. The reason to use a separate network with its own server was to guarantee the network integrity and configuration management control. The network server is based on a personal computer with the Pentium II microprocessor, 256 Mbytes RAM, 18 Gbytes SCSI hard discs and a CD ROM unit. The operating system of the server is the Novell Netware 3.2. The choice of this operating system was influenced by a very good operational experience. The operating system provides the possibility of access rights setting to individual users according to the configuration management plan.

![Network server, switch and tape unit](image)

Figure 1: Network server, switch and tape unit

To assure high reliability of stored data on the server, the hard discs mirroring technology is used. This technology, inherently supported by the Novell Netware operating system working with two hard discs, writes parallel data to both discs. If one disc crashes then data are stored on the other one. This solution consumes more disc space in comparison with other solutions (e.g. RAID), but there are simple hardware requirements. Furthermore, the server is equipped with the HP DAT40 tape unit to backup server on the DAT DDS-4 tapes, up to 40Gbytes on one tape. The utilized backup software is Cheyenne Arcserve 6.6.
2.2 The validation system

The purpose of validation is to check that the integrated hardware and software of the checked system satisfies the defined requirements. The tested system should be exercised through static and dynamic simulations of input signals during normal operation, anticipated operational occurrences and accident conditions. The validation system at the laboratory utilizes standard individual measuring instruments and does not develop any specific hardware. The reason is to achieve a simple structure of the system, reasonable maintenance, simple upgrade, clear evidence of the system configuration and also the possibility to utilize instruments from the validation system in other applications in research or education.

![Validation system](image)

**Figure 2:** Validation system

The main task of the validation system is to simulate signals of the neutron flux (reactor power) and to collect and check final data from the system, which is being validated. Because of the neutron chamber simulation for inputs of checked system, it is necessary to simulate a signal representing the neutron chamber either in current or pulse range according to the power range. The pulse range signal is simulated by generation of pulses with the proper frequency and the chamber current range simulation is achieved by a current with the proper amplitude according to the simulating reactor power course.

The validation system is controlled by the computer. The communication between the computer and individual instruments is based on the IEEE488 (HPIB) bus. The interface is supported by the majority of measuring instrument producers. The choice of this communication interface was a compromise between required performance, price, availability and the support of the instruments’ manufacturers.

To simulate neutron pulses, the HP8110A Pulse Generator is used to generate basic frequency (TTL), which can control the HP33120A Function/Arbitrary Waveform Generator to produce a typical response of the neutron chamber. It is possible to use either the TTL output direct to the digital logic inputs or the neutron pulse waveform to the analogue chamber input. The frequency modulation ability of the HP33120A generator supports the
simulation of the Poisson’s distributed pulse signal. The HP3245A Universal Source is intended to simulate the current range of neutron chambers for the validation.

2.3 Research team

The research team of primarily young researchers was created. The research team consists of the research team’s head, technical manager, two researches, one technician and two PhD students. Moreover, the department’s undergraduates take part in particular tasks in the project. During activities concerning setting requirements, the research team strongly co-operates with nuclear specialists of the department.

There are some difficulties to stabilize the research team staff. The principle problems are lack of reasonable accommodation and the concurrency of private industry, which can offer substantially higher salaries for the specialists. On one hand, the migration of staff causes some difficulties. On the other hand, the interest of the industry for our graduates and PhDs confirms the correctness of the research and educational activities.

3 RESEARCH ACTIVITIES

The activities of the research team started with studying and proposing the software development methodology. In addition, this methodology was applied to the real software development. This software was developed according to the proposed methodology. The verification and validation process followed software development. The validation system for integrated software and hardware tests was brought into being and its control software was developed. The tests of a high level programming language compiler were carried out utilizing the SOSAT tool.

3.1 Software development methodology

Finding the methodology started the activities of the research team. The appropriate literature was studied. Requirements and recommendations were defined for the software life cycle, necessary documentation, verification and validation process, configuration management and safety assessment.

To achieve the required quality and reliability of the developing software, the following three principles [1] should be followed. The first principle is fault avoidance through good software engineering and quality assurance throughout the complete life cycle of the software. The second principle is fault detection through verification and validation activities. The third principle, fault tolerance, means that the target system should be designed so that a failure will not jeopardise safety.

The software life cycle consists of the following phases that are to some extent self-contained, but will depend on other phases: requirements, design, coding, verification, integration of HW/SW, validation, installation and maintenance. Each phase of the software life cycle should be completed by the verification process to prove that all activities and requirements for this phase were successfully fulfilled. The static testing, systematic and random data testing, and real time aspects testing can be performed. Any faults, bugs and errors found during this phase should be carefully documented.

3.2 Power limiting system software development

The previously established methodology was applied to the power limiting system software development. The power limiting system is independent of other subsystems of the
control and safety system and performs reactor shut down if the maximal power of the reactor is exceeded.

Three fundamental documents were prepared before starting the software development. The first document is the “Quality Assurance Plan”. This document covers the methodology of the software production in all phases of the life cycle and defines measures to ensure high quality and reliability of the developing software. The second document, the “Verification and Validation Plan”, defines activities of verification and validation during the whole software life cycle. The plan explicitly sets out what should be checked in every phase of the software development, gives procedures on how to treat detected failures and errors, and defines what to do, if the errors arose in earlier phases of the software development, etc. The third document, the “Configuration Management Plan”, specifies approaches and methods for identifying software product items, for the control and implementation of changes, for secure storage of the developing software and documents, and for revision control and access rights of individual research team members to the software, data and documentation.

The software requirements of the power limiting system were deduced from the safety system requirements. The requirements for accuracy and time response of the power and period evaluation were established. Because of neutron flux statistics (Poisson’s distribution), different accuracy and time response requirements were set for different neutron flux levels. The behaviour of the system was set for typical neutron flux changes. All these requirements were carefully discussed with specialists in neutron physics, reactor engineering and neutron detection. Suitable mathematical representation of these requirements was found to facilitate verifiability, traceability and modifiability of the requirements. The requirements for the safety signal generation (reactor shutdown control signal) - fixed set limit values and response time - were also defined.

During the design of the software, modules for calculating power and period (the most important part of the developing software, because the safety signal generation depends on their values) were proposed. Three basic approaches of power calculation were studied: constant regression, linear regression and exponential regression. Linear regression and exponential regression appeared to be similarly accurate, constant regression is slightly less accurate. It was decided to use linear regression because of its easier implementation. After the design of the algorithm for power and period calculation, this algorithm was thoroughly tested to check all requirements.

In the next phase, the designed algorithms were converted into a programming language description. The high level programming language C was selected because of experience with it and availability of a compiler for the target hardware. The software was coded in bottom-up manner, from single functions to more complex modules.

All the above-mentioned software life cycle phases were accompanied by verification. Verification assures that during every individual phase of the software life cycle all necessary activities have been completely and correctly fulfilled and no errors have been introduced into the developing software.

### 3.3 Validation system control software

The validation system control program was prepared with the HP VEE graphical programming language. The advantage of a graphic programming language is to reduce program development time, to make the control programs easier to understand and to remove the need to study in detail instrument programming through built in drivers for most measuring instruments. The programs in the HP VEE are constructed by the connection of icons. The icons are data inputs and outputs, functions, instrument drivers, flow control
objects, displays, etc. The program can be made modular by usage of user defined objects and functions. The HP VEE provides direct support for the HPIB and VXI interfaces.

The first part of the validation system control software prepares data for the simulation of the reactor power. It is possible either directly to set the combination of simple frequency changes – linear, exponential and/or jump - or to read data from an external ASCII file prepared by other software. This file should contain time and suitable power value. The data for the power simulation can be displayed.

![Figure 3: Example of the panel view during validation](image)

The next part of the software controls instruments to produce simulation signals. After the start of the validation, the initial setting is performed, and then the control data are sent through the HPIB to the instruments in the required time steps.

During the simulation, the data from the tested system are received, decoded and stored for further checking. The safety signal, after conversion from fibre optics, is tested by the Universal Counter in the pulse counting mode over 0.1 sec. because of its pulse character. The validation system software calculates, according to the requirement’s definitions, expected values for power, period and status, and compares them with regard to the required accuracy and time response with received data. The status of the validation is displayed in the software presentation window, and the results of the validation with time stamps are stored in an ASCII file. The file contains data to document validation and to evaluate the cause of incidental failure of the performed validating test.

### 3.4 C compiler tests utilizing the SOSAT tool

The SOSAT tool, developed by the Halden Reactor Project and TÜV Northern Germany, granted the research laboratory, was used to check the correctness of the C compiler Borland C++ 3.1. The reason was to study the properties and abilities of the SOSAT and prepare us for its real programs future usage.

The SOSAT tool supports the evaluation of binary code programs. It carries out disassembling, program structure analysis and the generation of the pseudocode. Then it is
possible to compare the generated pseudocode with original source code of the program and to compare them. The SOSAT also provides program metrics.

The test of the C compiler was limited on the ANSI C abilities because of its intended use in ‘embedded’ systems. The tests were carried out in two steps. In the first step, small programs using individual checked keywords were established, translated to the binary code, evaluated with SOSAT tool, and the SOSAT output pseudocode was compared with original source code. Furthermore, some more complex programs were prepared to check different data types, function calls with diverse arguments and returning values, pointers etc.

It was found that the SOSAT tool is very helpful for binary code program evaluation. But it is not possible to assume the fully automatic run of the evaluation. Manual interactions are often necessary, also the comparison of the pseudocode output cannot be carried out automatically. It is clear that from the binary program code it is not possible to produce its complete original source code. Therefore, man’s activity is unavoidable during the evaluation. In addition, adequate knowledge of the microprocessor, its instruction set, assembly language, C language conventions (arguments, variables allocation, data types etc.) are essential for the successful utilization of the SOSAT tool. The SOSAT tool is now available for X86 microprocessor programs; it would be helpful to also have the SOSAT tool for other microprocessors.

4 UTILIZATION OF THE PROJECT RESULTS IN EDUCATIONAL PROCESS

The intention of the project was also to utilize results of the project in the educational process at the faculty. First, the undergraduates and PhD students took active part in the project solution. Their activities were usually connected with same particular task elaboration – annual research works, diploma theses for undergraduates and dissertations for PhD students. In addition, new lectures were included into the study plans of students. The lecture “Digital control and safety systems of nuclear devices” deals with particular aspects of the usage of computer based systems for nuclear safety applications with special respect to the quality assurance of software. The software life cycle, configuration management, required documentation, V&V process, FAT, SAT, etc. are included in this lecture. Furthermore, new lectures relative to the project’s subject are “Computer control of physical experiments” exploiting gained knowledge about graphical programming tools during the validation system software development and “PLA design” based on experience with programmable logic.

5 NATIONAL AND INTERNATIONAL CO-OPERATION

Before and during the project, many national and international contacts were established and maintained. On the national level it can be mentioned Nuclear Research Institute in Řež, which incorporated the Department of Nuclear Reactors and the established laboratory into its membership in the Halden Reactor Project. Close co-operation exists between the department and State Office for Nuclear Safety Czech Republic, which comprises expert reviews and participation in proposals of safety standards. A traditional partner is the Škoda Pilsen company (division of Nuclear Engineering). Common interests include the upgrade of the contemporary VR-1 reactor control and safety system and the conversion of a research reactor in Bulgaria to an educational one with the nuclear reactor I&C reconstruction. In both projects, the laboratory actively participate.

The Halden Reactor Project (HRP) should be mentioned as the principle subject of international co-operation. Through the HRP we gained access to literature, experience, important international contacts and we also appreciate the opportunity to find out new knowledge and to publish our results at HRP conferences. Co-operation with the
Atominstitute in Vienna was focused on the joint testing of hard-wired and computer based nuclear reactor safety system properties and on the behaviour of the systems during accident conditions because of the higher reactor power in Vienna.

Members of the research team take part in the training courses of the IAEA as lecturers. The subjects of lectures are control and safety systems of nuclear devices, physical protection and quality assurance of software for nuclear safety applications. The research laboratory is preparing a training course in the maintenance of the electronic and digital systems of experimental nuclear reactors with support of the TU Vienna.

Recently, the research team has become a corresponding member of the EWICS (European Workshop on Industrial Computer Systems) TC7 group.

6 CONCLUSION

The purpose of this paper was to summarize the activities carried out during the project of the software development for safety critical applications. The properly equipped laboratory was established, and the research team was formed. The activities of the research team were focused on software development methodology, the utilization of this methodology for the power limiting system software development, bringing the validation system into being and using it for integrated hardware/software validation and application of the SOSAT tool for high level programming language compiler tests. The collected experiences from the project were also incorporated in the educational process at the faculty and undergraduates and PhD students took part in it.

The main aims of future research activities are the innovation of the VR-1 training reactor control and safety system in co-operation with Škoda Pilsen company, the investigation of the reactor power calculations utilizing Fourier transformation and other up-to-date methods, the study and application of the new standard IEC 61508 “Functional Safety of Electronic Safety Related Systems”, the investigation of formal methods utilization for the software development and proposal and application of the quality assurance procedure for PLA design.

REFERENCES


