©1996 Copyrights reside with the original authors who may be contacted directly

ISBN 0-620-20568-7

Cover printed by Natal Printers (Pty) Ltd, Pietermaritzburg
Copying by the Multicopy Centre, University of Natal, Pietermaritzburg
Binding by Library Technical Services, University of Natal, Pietermaritzburg

The views expressed in this book are those of the individual authors
FOREWORD

This book is a collection of papers presented at the National Research and Development Conference of the Institute of Computer Scientists and Information Technologists, held on 26 & 27 September, at the Interaction Conference Centre, University of Natal, Durban. The Conference was organised by the Department of Computer Science and Information Systems of The University of Natal, Pietermaritzburg.

The papers contained herein range from serious technical research to work-in-progress reports of current research to industry and commercial practice and experience. It has been a difficult task maintaining an adequate and representative spread of interests and a high standard of scholarship at the same time. Nevertheless, the conference boasts a wide range of high quality papers. The program committee decided not only to accept papers that are publishable in their present form, but also papers which reflect this potential in order to encourage young researchers and to involve practitioners from commerce and industry.

The organisers would like to thank IBM South Africa for their generous sponsorship and all the members of the organising and program committees, and the referees for making the conference a success. The organisers are indebted to the Computer Society of South Africa (Natal Chapter) for promoting the conference among its members and also to the staff and management of the Interaction Conference Centre for their contribution to the success of the conference.

On behalf of the Organising Committee
Vevek Ram
Editor and Program Chair
Pietermaritzburg, September 1996
Organising Committee

Conference General Chairs
Mr Rob Dempster and Prof Peter Warren (UNP)

Organising Chair
Dr Don Petkov (UNP)

Secretariat
Mrs Jenny Wilson

Program Chair
Prof Vevek Ram (UNP)

Program Committee
Prof Peter Wentworth, Rhodes
Dr Milan Hajek, UDW
Prof Derek Smith, UCT
Prof Anthony Krzesinski, Stellenbosch
Dr Don Petkov, UNP
Mr Rob Dempster, UNP
Prof Peter Warren, UNP
# Table of Contents

- Foreword
- Organising Committee
- List of Contributors

## Keynote Speaker

**The Role of Formalism in Engineering Interactive Systems**
M D Harrison and D J Duke

## Plenary

**Industry-Academic-Government Cooperation to boost Technological Innovation and People Development in South Africa**
Tjaart J Van Der Walt

**Checklist support for ISO 9001 audits of Software Quality Management Systems**
A J Walker

**The IS Workers, they are a-changin'**
Derek Smith

## Research

**Examination Timetabling**
E Parkinson and P R Warren

**Generating Compilers from Formal Semantics**
H Venter

**Efficient State-exploration**
J. Geldenhuys

**A Validation Model of the VMTP Transport Level Protocol**
H.N. Roux and P.J.A. de Villiers

## Intelligent Systems

**Automated Network Management using Artificial Intelligence**
M Watzenboeck

**A framework for executing multiple computational intelligent programs using a computational network**
H L Viktor and I Cloete

**A Script-Based prototype for Dynamic Deadlock Avoidance**
C N Blewett and G J Erwin

**Parallelism: an effective Genetic Programming implementation on low-powered Mathematica workstations**
H. Suleman and M. Hajek

**Feature Extraction Preprocessors in Neural Networks for Image Recognition**
D Moodley and V Ram
Real-Time Systems

The real-time control system model - an Holistic Approach to System Design
T Considine

Neural networks for process parameter identification and assisted controller tuning for control loops
M McLeod and VB Bajic

Reference Model for the Process Control Domain of Application
N Dhevcharran, A L Steenkamp and V Ram

Database Systems

The Pearl Algorithm as a method to extract information out of a database
J W Kruger

Theory meets Practice: Using Smith's Normalization in Complex Systems
A van der Merwe and W Labuschagne

A Comparison on Transaction Management Schemes in Multidatabase Systems
K Renaud and P Kotze

Education

Computer-based applications for engineering education
A C Hansen and P W L Lyne

Software Engineering Development Methodologies applied to Computer-Aided Instruction
R de Villiers and P Kotze

COBIE: A Cobol Integrated Environment
N Pillay

The Design and Usage of a new Southern African Information Systems Textbook
G J Erwin and C N Blewett

Teaching a first course in Compilers with a simple Compiler Construction Toolkit
G Ganchev

Teaching Turing Machines: Luxury or Necessity?
Y Velinov

Practice and Experience

Lessons learnt from using C++ and the Object Oriented Approach to Software Development
R Mazhindu-Shumba

Parallel hierarchical algorithm for identification of large-scale industrial systems
B Jankovic and VB Bajic
List of Contributors

Vladimir B Bajic  
Centre for Engineering Research, Technikon Natal, P O Box 953 Durban 4000

C N Blewett  
Department of Accounting University of Natal King George V Avenue Durban 4001

Justin Cansfield  
Department of Accounting University of Natal King George V Avenue Durban 4001

Tom Considine  
Apron Services (Pty) Ltd P O Johannesburg International Airport 1600

Eric Cloete  
School of Electrical Engineering Cape Technikon Box 652 Cape Town

I Cloete  
Computer Science Department University of Stellenbosch Stellenbosch 7600

O A Daini  
Department of Computer Science University of Botswana Gaborone Botswana

Nirvani Devcharan  
Umgeni Water Box 9 Pietermaritzburg 3200

P J A de Villiers  
Department of Computer Science University of Stellenbosch Stellenbosch 7700

Ruth de Villiers  
Department of Computer Science and Information Systems UNISA Box 392, Pretoria, 0001

G J Erwin  
Business Information Systems University of Durban-Westville Private Bag X54001 Durban 4000

G Ganchev  
Computer Science Department University of Botswana P Bag 0022 Gaborone, Botswana

J Geldenhuys  
Department of Computer Science University of Stellenbosch Stellenbosch 7700

Louise Gibson  
BIS, Dept Accounting & Finance University of Durban Pvt Bag X10 Dalbridge 4014

Mike Hart  
Department of Information Systems University of Cape Town Rondebosch 7700

M. Hajek  
Department of Computer Science University of Durban-Westville P Bag X54001 Durban 4000

A C Hansen  
Dept of Agricultural Engineering University of Natal Private Bag X01 Scottsville 3209

J M Hattingh  
Department of Computer Science Potchefstroom University for CHE Potchefstroom 2520
Boris Jankovic  
Centre for Engineering Research  
Technikon Natal  
P O Box 953,  
Durban 4000  

Paula Kotze  
Department of Computer Science and  
Information Systems  
UNISA  
Box 392  
Pretoria, 0001  

J W Kruger  
Vista University  
Soweto Campus  
Box 359  
Westhoven 2124  

A C Leonard  
Dept of Informatics  
University of Pretoria  
Pretoria  
2000  

Ben Laauwen  
Laauwen and Associates  
P O Box 13773  
Sinoville  
0129  

Mari Le Roux  
Information technology, development: project  
leader  
Telkom IT 1015  
Box 2753  
Pretoria 0001  

P W L Lyne  
Dept of Agricultural Engineering  
University of Natal  
Private Bag X01  
Scottsville 3209  

Rose Mazhindu-Shumba  
Computer Science Department  
University of Zimbabwe  
Box MP167  
Harare, Zimbabwe  

Meredith McLeod  
Centre for Engineering Research,  
Technikon Natal,  
P O Box 953  
Durban 4000  

D Moodley  
Computer Management Systems  
Box 451  
Umhlanga Rocks  
4320  

Andrew Morris  
P O Box 34200  
Rhodes Gift  
7707  

R C Nienaber  
Technikon Pretoria  
Dept of Information Technology  
Private Bag X680  
Pretoria 0001  

E Parkinson  
Department of Computer Science  
University of Port Elizabeth  
Box 1600  
Port Elizabeth 6000  

Don Petkov  
Department of Computer Science and  
Information Systems  
University of Natal  
PBag x01  
Scottsville 3209  

Olga Petkov  
Technikon Natal  
Box 11078  
Dorpspruit 3206  
Pietermaritzburg  

N Pillay  
Technikon Natal  
Box 11078  
Dorpspruit 3206  
Pietermaritzburg  

vii
REAL-TIME CONTROL THE SYSTEM MODEL - AN HOLISTIC APPROACH TO SYSTEM DESIGN

TN CONSIDINE
APRON SERVICES (PTY) LTD
JOHANNESBURG INTERNATIONAL AIRPORT
PO KEMPTON PARK
1600
(011) 978-3493

The Real-time Control System Model is a system design methodology which provides robust systems based on Object technology. The paper will detail the components of the model including the control layers and the time-based levels into which the model is divided. The application of the Model in the design of an operational system is described in detail particularly the sensor, user interface and world model layers.

In the United States 35% of all software projects are cancelled before completion[3]. 91% of large projects fail as do 84% of small projects[4]. Indeed a snap survey of 14 magazines published in the past year showed 16 articles published on Software Quality or the lack of it. I do not know how South Africa compares to the American situation but I suspect that it is much the same.

Why does software fail?

A recent study by Software Productivity Research showed that the most significant factors were the lack of automated software cost estimating tools, automated software project management tools, effective quality control and effective tracking of software development milestones[5]. All of these are needed because too much software is written from scratch.

It seems logical to assume use of standard modules or objects in systems can promote successful development. However, the use of standard software modules seems, in many cases, to be beyond the reach of developers. Why should this be? Taking an extreme view of software development it could be said that software development is similar to building a house. To build a house you simply buy the objects, bricks, window frames, baths and so on, assemble them and you have a house. This does not work properly since there are choices in housebuilding, floor finishes, for example. So a house is not simply a collection of objects but more accurately, a collection of standard objects, customisable objects and new objects. Over many thousands of years the building industry has standardised on customisation of objects. Cost features in the equation as well, the more expensive the house, the less the use of standard objects and the more the use customised or new components.

Use of standard modules in house building has it's place but is subject to the factors outlined before. Similarly, use or re-use of components in a system is a blend of standard components, customisable components and new components. It follows then, that there is a hierarchy of re-usability which depends on the system under consideration. Or to put it another way, if you were creating special effects for the Jurassic Park versus Godzilla film you would use mostly new components whereas if you were doing an accounting system for your hairdresser you would use standard components almost everywhere. Re-usable components may then be graded by their complexity. Simple components like multiplication and division (yes, once upon a time, these were programmed by programmers) are normally re-usable. Complex components used in advanced client server applications are not. Can we not sub-divide components until we get totally re-usable components? Not really. Many problems are not sub-divisible and others would, if sub-divided, increase the complexity of the system. And complexity would increase running time, a real problem in graphics systems, for example. Less of a problem in a single-user accounting system and no problem at all a word-processing system. If re-use is a function of
complexity and complexity is a function of running time, is it possible to derive rules that will guide us in the use or re-use of standard components?

We need a methodology that will sub-divide systems into efficient blocks or methods or components.

The Real-time Control System does just this. It accomplishes this by formalising the structure of an entire system or organisation in such a way that data flows are easily visualised and temporally analysed.

**What is the Real-time Control System Model?**

It is a reference model for intelligent real-time systems - loosely, systems in which the database is updated as events occur. It sub-divides organisation management into seven basic elements or layers: planning, execution, control, world modelling, sensory processing, user interface and communications - Albus [2]

The Real-time Control System clusters these elements into computational nodes that control specific subsystems, and arranges these nodes in hierarchical levels so that each level has a characteristic granularity. The levels are graduated in levels depending on the granularity of the level. The granularity of each level in the control hierarchy is derived from the characteristic timing and functionality of that level. For example, in a manufacturing environment, a dozen hierarchical levels are more or less standard. The actual timings for each level and the levels used may vary from business to business but the general principle remains unchanged. By sub-dividing the organisation = system in this way, we are able to determine the complexity of each sub-system. The Real-time Control System gives the system architect the ability to measure complexity along any axis he or she chooses. Returns on Re-use Investment are quick because system management can focus on specific problem areas. Yes; Re-use is an investment, in the same way that developing a new process or installing Statistical Quality Control in the factory is an investment.

The layers into which the work is divided are as follows:

**Planning**

This layer is where the strategic and tactical planning takes place. The time horizon at this level varies from one day to three or more years depending on the business.

**Execution**

This layer is where the commands are issued to the operating level based on planned requirements and task decomposition.

**Control**

This layer controls the execution of the process and is where value judgement is performed. The value judgements may be made by humans or by computers using Expert Systems.

**Communications**

This is where messages are passed between levels.

This layer includes electronic mail within the organisation and communication with customers and external databases outside the organisation. Local and Wide area networks will be used between all elements of the organisation so that data is continually kept current.
User interface

This layer includes the software that enables a manager to scan the business. The manager may be provided with standard reports and may be able to select additional information. The information will be presented in graphical (in its broader sense) format for easy assimilation. The layer will also offer analysis functions. The current situation will be presented in real-time as will future projections based on current events. In SynQontrol, a Statistical Process Control system, the latest release displays test results in graphical format, as they enter the system, in a Real-time Status Management Block that is continually available to management. From which it can determine the quality of the current and past production. These and other information displays enable management to control the company within its envelope. Managing proactively rather than reactively and conserving energy.

Sensory processing

This is the system picks up input. Depending on the layer different inputs are collected and processed. Some messages are passed to higher layers, while some are processed at the level at which they are received. It is worth noting, at this point, that 85% of all errors which occur in a system, computer or manual are system errors and cannot be fixed by the users. It is thus necessary to have direct and quick communication to the Command and Control layers. And good error recording routines so that you can analyse the problems when enhancement or extension time comes around.

World Model

This is the repository of static and dynamic data and the business rules of the organisation.

Static database

In this database are kept all or most of the fixed information of an organisation. Some data entities that will be kept: customers' addresses, recipes, routings, part data including CAD data and historical sensor data for analysis. This data will probably be kept in an Object Relational Database. Data enabling Virtual Reality functionality will also be kept in this database.

Dynamic database

In this database, which can easily be a conventional Relational Database, is stored the transient data that is the lifeblood of the business. This is composed of such data as orders, stock holdings, money in other words those entities that change dynamically. It also includes external information like market research, benchmarking comparatives, political and economic analysis.

Business Rule set

The rules that govern the business are kept in this database. Note this particular database may, in practice, be a virtual database in that all the rules may be spread over different datasets. For example, the Business Rules relating to credit management may be kept in a separate dataset from those which relate to product quality. The format of the database dictates storage of Business Rules. In an Object Database the Rules are encapsulated with the data while in a Relational Model Rules may be stored as data in tables accessed by functions.

How are the levels organised?

I will now describe the 12 levels into which a system may be divided. 1 level is described in some detail, the others in less.
Level 12 - Strategic planning

This level schedules and controls the company for a period of about 3 years. Major capital equipment and new product decisions are made at this level. How does a 3 year planning horizon fit in with a real-time control system? The answer is simple. Decisions made today will affect the company’s energy in the future. Energy is the sum of the men, machines, money, methods and measurements that is affected by the milieu in which the company operates. (Milieu = Environment. All of the others begin with the letter M.) If a company squanders its energy, it may be unable to compete in its environment in future. The layers found in this level are planning, execution, control, communications, user interface, sensory processing, and the World Model. Let us examine each of these in turn.

Planning has already been mentioned, the Execution layer implements what has been planned. The Control layer monitors the situation and either takes corrective action or passes information to the Execution layer for display in the User Interface layer. The Sensors at this level are concerned with long-term planning and will include monitoring the political, economic and competitive environment. The Strategic Planning level is primarily concerned with Modification of the World Model especially the Business Rules.

Level 11 - Medium term planning

This level schedules and controls the company for a planning period of about 1 year. Some capital equipment decisions and product extensions are made here.

Level 10 - Short term planning

This level schedules and controls the company for a period of about 3 months. Sales promotion decisions are made at this level.

Level 9 - Scheduling

This level schedules and commands the company for a planning period of about 1 month. Forecast orders are assembled into batches, raw materials are ordered and preventative maintenance planned. Sensor data at this level may be obtained from Market Research.

Level 8 - Factory

This level schedules and controls the factory for a planning period of about 1 week. Schedules are drawn up to meet delivery requirements for accepted and expected orders. Raw material deliveries are scheduled and instructions issued to suppliers to deliver at planned times and locations. Sensory Processing at this level includes bar-code scanners that provide data for raw material and warehouse management.

Level 7 - Shop

The level schedules and controls the activities of one or more manufacturing cells for about 24 hours. Commands are issued to cells to prepare schedules for the batches. The World Model at this level accesses the dynamic database of orders and the resources needed to process the orders.

Level 6 - Cell

This level schedules and controls the activities of several workstations for approximately 1 hour. Batches of ingredients are delivered to workstations and commands are issued to workstations to mix production batches. The World Model at this level accesses the database of equipment and materials needed for the immediate 1 hour planning period.
Level 5 - Workstation
This level schedules tasks and controls the activities within each workstation for about 5 minutes. The World Model at this level accesses the databases which contain works instructions, parts and tools held at the workstation and equipment readiness.

Level 4 - Equipment task
The equipment task level schedules task and controls the activities of a machine for a 30 second planning period. This level decomposes each equipment task into elemental moves for the subsystems that make up each resource. The World Model references the static database which for this level contains part attributes.

Level 3 - Elemental move
The elemental move level schedules and controls simple machine operations that require a few seconds. The World Model at this level will access the static database that contains part detail including some which may be in a CAD format.

Level 2 - Primitive
This level plans trajectories for tools, manipulators and inspection probes to minimise time and to optimise performance. The World Model at this level accessing the Static Database that supplies data for tool movement as related to items to be manufactured.

Level 1 - Servo
This level transforms commands from tool path to joint actuator co-ordinates. Planners interpolate between primitive trajectory points with a 30 millisecond look ahead. The World Model at this level contains data relating to state variables such as velocities, temperature and equipment reaction times. Data from reaction time sensors is plotted on a graph so that the technical manager becomes intuitively aware of potential problems such as hydraulic failure and can taken preventative action. Expert systems may highlight problems.

Standard component use related to levels.
The temporal granularity of each level will determine the possibility of use of standard components. At the lowest level where execution speed is important, for instance, in an aircraft control system, the developer would be working toward a custom solution that is tuned for efficiency. Components will probably be written in C++ or even Assembler. At the highest level where speed is not of great importance we would work towards a more generalised approach capable of solving many problems. In practice, we find that this is the case. Systems or sub-systems that are not time-critical tend to be the ones that have the greatest availability. Spreadsheets and graphs are good examples of off-the-shelf components. One would not think of flying a passenger aircraft using a spreadsheet package but making a purchase decision with one is normal.

The Real-time Control System, therefore, helps the system designer to plan re-usability. By modelling the entire organisation, the designer can determine the areas in which re-use will have the quickest payback.

How the Real-time Control System was used at Apron Services
The ground handling management system in use at Apron Services captures all customer aircraft movements and services provided to those customers. These services are cargo and baggage loading and unloading, passenger transport for those aircraft not parked at airbridges, assistance for sick passengers and aircraft towing. In addition it captures equipment, labour and fuel usage. Customer invoices as well as operational statistics are presented by the system. Monitoring of operations is currently at level 7. Extensions will include scheduling of operations (levels 6 and 7) and forward planning - level 8 and above. Payback for the system was less than 3 months.
Productivity in the organisation increased. Error rates in one particular area dropped from an average of 25% to less than 1%. Invoices that took up to 14 days after month-end to prepare are down to 1 day. The MIS department now has the ability to evaluate the productivity of operational departments and provides daily feedback to these departments in respect of errors and resource utilisation.

The system also provides South African Airways (a customer of Apron Services) with accounting and operational management information not available from their own information system.

A Relational database is used for the system. Some data would be more efficiently stored in an Object database but implementation of this is not a priority.

The implementation of the Real-time Control System followed the following phases:

1) User Requirements definition using the Real-time Control Model.
   Since aircraft handling is, at this point, largely a manual operation, the system was designed from level 5 upwards. Initial implementation was at level 7. This was chosen because it was the interface between the manual and the computer-based components of the system. Insufficient reliable data was available for the levels below 5. However, the system was designed to record data for future level 4 development. Once South African airports are mechanised to the extent of that of, say, Denver (was that a good example?) development will progress to lower levels.

   PLANNING at level 7 relates to staff and equipment availability and scheduling to optimise resource use.

   EXECUTION is not used at this stage owing to the volatility of the industry. 1 late 747 or bad weather can destroy a schedule.

   CONTROL is performed at this level by supervisors

   COMMUNICATION is done through a file server. Hand-held computers are not considered to be useful because of the labour intensive nature of the work. Process data is, therefore, captured from input forms.

   The USER INTERFACE is through the Relational data base using its query facilities or exception reports.

   There are altogether 74 exception and warning reports. The warning reports cover “Fuzzy” problems. An example might be that an Airbus300 used fewer buses than might be expected.

   Investigation of this possible problem revealed serious deficiencies in the bus control system. The depth of problem reporting has provided ample opportunity for customising reports in response to ad-hoc enquiries. Turnaround for ad-hoc enquiries is generally minutes.

   SENSORY INPUT is done manually by operating staff. However, designing the system with RCS means that upgrading sensors to, for example, barcode scanners or vehicle tracking devices will necessitate only changing that interface - the rest of the system will remain unchanged.

   The WORLD MODEL STATIC DATABASE contains amongst others, aircraft data like the number of passengers, the number of steps required and type of cargo handling. Other static data relates to customers (airlines), especially the aircraft used. Every registration number of every aircraft handled in the last 18 months is kept in the database. This data is used to assist input accuracy.

   DYNAMIC DATA is operating data like arrival times, registration numbers of equipment used and quantity of baggage handled.

   The BUSINESS RULES SET contains operating standards like baggage delivery and other handling times. Non-conformance with Business Rules is advised to management and staff for both corrective action and recognition of good performance.

2) Failure Mode and Effect Analysis. (Developed by Ford Motor Company.) This is an effective tool for estimating the cost and effect of potential problems. It provides focus to the system
designer, ensuring that important potential problem areas are covered in the design. We used Byzantine Failure Analysis - Turek and Sasha [6] to supplement the FMEA since we suspected that one or more of the processors or the software itself might fail and cripple the system.

3) Executable system modelling using Tool Abstraction proposed by Garlan, Kaiser and Notkin. [7]

The finished system specifications were presented to one of the top programming houses in the country who programmed the system in 3 weeks. It went live immediately and has not failed during 18 months of use.

Additions have been made to the original design extend its facilities and to accommodate changes in the environment. These have not impacted on the system because The Real-time Control System model methodology ensures design robustness and opportunity for re-use.

Summary

The challenge for South Africa in the next century will be to utilise our resources efficiently. This will entail structural changes in our organisations. Management will have to become immersed in the system. The system will be the organisation. Management will spend most of its time monitoring company health. Expert systems will analyse data arriving through the sensor layers, comparing this data to standards derived from internal and external research. Any variation in, for example, Productivity will manifest itself at an early stage and corrective action can be taken immediately. This is not futuristic, military aircraft are currently being designed to have an integrated structural health-and-usage monitoring system. “The system is used to perform real-time airframe-fatigue calculations and to monitor significant structural events and flight performance parameters.” [1] Virtual Reality will enable the MD to “walk through” even remote plants on a daily basis. Access to data will not be restricted to senior management, the information system will allow everyone to know what everyone else knows. This will eliminate knowledge hierarchies in which knowledge is jealously guarded from other workers, staff and bosses alike.

The Real-time Control System will enable response to market forces to be made in hours rather than days or weeks as is the case now.

References
