2005

An investigation into the use of B-Nodes and state models for computer network technology and education

Gurpreet Kohli
Edith Cowan University

An Investigation into the Use of B-Nodes and State Models for Computer Network Technology and Education

A thesis submitted in fulfilment of the requirement for the award of DOCTOR OF PHILOSOPHY

By
Gurpreet Kohli

Faculty of Computing, Health and Science
Edith Cowan University
2005

USE OF THESIS

Acknowledgments

I should like to thank my principal supervisor Associate Professor S.P. Maj for his assistance, sincere dedication, critical thinking, insight, and encouragement in helping bring this undertaking to its conclusion. I should also like to thank Associate Professor T. Fetherston of the School of Education at Edith Cowan University for his valuable input in data collection and analysis of my work. I should also like to thank all collaborators of the published papers and especially my colleagues Dr D. Veal and Mr G. Murphy for research collaboration and the need for feedback from time to time. My thanks also to the research support Officer Mr S. Cikara for his help with hardware and software. Finally, to my family without whose co-operation love and support this work would have been impossible to complete.

Declaration

I certify that this thesis does not to my knowledge and belief:

i. Incorporate without acknowledgement any material previously submitted for a degree or diploma in any institution of higher education;

ii. Contain any material previously published or written by another person except where due reference is made in the text; or

iii. Contain any defamatory material.

Signed:

Date:

Abstract

This thesis consists of a series of internationally published, peer reviewed, conference research papers and one journal paper. The papers evaluate and further develop two modelling methods for use in Information Technology (IT) design and for the educational and training needs of students within the area of computer and network technology.

The IT age requires technical talent to fill positions such as network managers, web administrators, e-commerce consultants and network security experts as IT is changing rapidly, and this is placing considerable demands on higher educational institutions, both within Australia and internationally, to respond to these changes.

Computer and Network Technology systems are complex, and considerable understanding is required to design and operate them. Models based on abstraction are widely used in the field to assist in providing that understanding. Of all the models evaluated in the design of IT systems in the course of this research, none provided a simple yet quantitative technique that could be used to model the computer technology with appropriate units. In an attempt to overcome this absence, Maj and Veal proposed the use of Bandwidth-Nodes (B-Node) to model the Personal Computer (PC) (e.g. hard disk, microprocessor, RAM). A PC is a complex collection of heterogeneous devices interconnected by a range of bus structures. Using this model, each node (microprocessor, hard disc drive etc) can be treated as a data source/sink capable of, to various degrees, data storage, processing and transmission. Significantly, it employs a common unit, regardless of device, from which other user-oriented units may be derived. Previous work suggests that the B-node model may provide the basis of a suitable conceptual map, and hence a pedagogical framework for an introduction to computer technology. This model is conceptually simple and controls the level of detail by abstraction. Hence the model can be used to examine PC performance and also has pedagogical value. ***The first aim of the research was to develop further and evaluate the use of B-nodes to model hardware running the IT system.***

The B-Node model may be used for resource modelling (server hardware, switches and routers) and hence may provide an improved way to plan the capacity of infrastructure running critical applications, such as e-commerce sites.

Furthermore, using the B-Node model, it is relatively simple to estimate the bandwidth of networking devices such as switches and routers.

All models have limitations. PC hardware and operating systems are complex, and this makes obtaining accurate bandwidth measurements inside the PC difficult to obtain. A formula which relies upon Efficiency rating or E-coefficient (**B**andwidth = **C**lock speed * **D**ata path width * **E**fficiency) need to be evaluated. However, evaluating the E-coefficient proved to be problematic. Despite this, it was felt the general principles employed by the B-Node model could be applied to internetworking devices such as switches and routers. It is, relatively, easier to make measurements on devices such as switches and routers compared to PCs.

Hence, the second aim of this research was to investigate the use of the B-Node model and evaluate other conceptual models used to improve understanding of computer networks.

There are a wide range of equally valid approaches to teaching networking. One approach is to teach the actual internetworking technologies such as switches, and routers. Both within Australia and internationally there is a demand for this practical, hands-on approach to networking curriculum. Accordingly, some universities have adopted the Cisco Network Academy Program (CNAP) and hence obtain access not only to vendor-specific curriculum and certification (Cisco Certified Networking Associate (CCNA) and Cisco Certified Networking Professional (CCNP)), but also to low cost equipment (switches and routers). Routers and switches are managed using a Command Line Interface (CLI) in conjunction with various software diagnostic tools such as the Cisco Discovery Protocol (CDP), PING, TRACE, IP ROUTE and Telnet. The CLI allows the user to determine and modify the status of the various components of a device, such as routing table entries, Address Resolution Protocol (ARP), table entries, interface status etc. However, there are problems with this approach. For example, a single CLI command may produce extensive output that requires considerable skill to interpret and evaluate; in addition, learning the CLI command structure is often difficult for novices. Furthermore, the status information of the many different device tables, interfaces etc must typically be obtained by a number of different CLI commands. This may be problematic during teaching, when it is often necessary to integrate all of the information from a number of different CLI commands. It was also found that routers and switches are usually treated as 'black boxes'. This is contrary to educational theory that supports the need for a conceptual model as the foundation of curriculum design. This research further developed the B-Nodes into state models, which can be used in promoting students' learning of computer and network technology. Furthermore, state models can be developed further to model various protocols e.g. Spanning Tree Protocol (STP), Open Shortest Path First (OSPF), etc.

A possible major contribution to new knowledge of these papers relates to the integration of state models of switches and routers with students' learning.

Within educational research it is well documented that after successfully completing an examination it is not uncommon for the majority of students to demonstrate very poor retention of not only factual information but also of concepts presented during the course of the study. Students whose learning was based upon state models clearly demonstrated that they had internalized the model and were able to produce valid network models after the elapse of a significant period. This conclusion is made for two reasons: the students' model was not an exact copy of the state model provided in class; and the students demonstrated an understanding of the concept they had been taught. However, further work is needed.

The state models appear to provide a richer understanding of device operations and hence may provide better teaching models for both undergraduate and postgraduate units in computer and network technology. Furthermore, state models may also provide an improved way of troubleshooting complex routing protocols and hence provide a better way of managing routers and switches.

The two main research questions addressed by this thesis are:

1. Can B-Nodes theory be applied to model hardware infrastructure supported by computer and network technology?
2. Can state model developed from B-Node theory be applied to the teaching of internetworking and hence promote student learning?

Table of Contents

TABLE OF FIGURES

1. INTRODUCTION

The computing discipline has evolved rapidly over the last 40 years. Computing is now integrated into almost every aspect of modern higher education, from the delivery of educational material to the use of technology in learning and training. Computer applications are currently integrated into many aspects of human life such as paying bills using the Internet, and video communication over the Internet. Rapid changes in Information Technology place significant demands on curriculum content. The curriculum must not only provide students with knowledge and skills relevant to employer expectations, but also provide a theoretical foundation that will support further learning when required by subsequent technical development (Abelman, 2000; Barnett, 2000). This suggests the need for professional bodies to regularly monitor and update the curriculum. As Khan et al notes:

> The phenomena of establishing and operating a university specially awarding a computer science degree need strict checking both by the monitoring and regulatory bodies of the donor and the recipient countries. Especially in case of computer science education because the rate of migration of software professional and rate of transition of technology is significantly high if compared with other subject areas (Khan, Kadir, Shaikh, & Kidwai, 2001, p.4).

Professional bodies are variously responsible for the accreditation of awards and defining curriculum content. According to the Association for Computing Machinery (ACM):

> The expansion of the discipline beyond the traditional boundaries of computer science certainly has a significant impact in the broad domain of computing education. At the same time, the problem of developing a coherent curriculum for the computing field as a whole is an extremely difficult undertaking, given the enormous breadth of specialties within the field (ACM, 2001, p.9).

The ACM further states:

> Professional bodies already exist for those disciplines, and it is certainly important that curriculum design in those specialties be undertaken by people with the relevant expertise. Review committees in several of these areas have recently published new curriculum studies, such as the MSIS 2000 curriculum for information systems

> and the Software Engineering Body of Knowledge (SWEBOK) definition (ACM, 2001, p.9).

There have been ongoing efforts to design computing curricula since the 1960s. The first report on curriculum design appeared in the Association of Computing Machinery (ACM) journal in 1968 (ACM68, 1968). In 1976, the Computer Society published a model curriculum for computer science education (ACM78, 1979). During the late 80s and early 90s there was a major shift in emphasis in the computing industry, due, in large measure, to the need to form a bridge between the software and hardware oriented approaches (ACM, 1991). The mid 90s witnessed the evolution of the Internet. The growth of the Internet as a communications vehicle since 1995 has been nothing less then spectacular. According to Dell "*The Internet is like a weapon sitting on a table, ready to be picked by either you or your competitors*" (Dell, 1999). Due to the dynamic nature of technology, there is considerable demand for computer and network curricula to be able to adapt to these changes. The infrastructure that supports Information Technology (IT) requires technical talent to fill various specialist positions such as network managers, web administrators, e-commerce consultants and network security experts. These advances in computing technology resulted in a joint review of computing curricula by the ACM and the Institute of Electrical and Electronic Engineering (IEEE). The ACM /IEEE proposed a Model Curriculum for Computing in 2001 (ACM, 2001). This curriculum was designed to match the developments in computing technologies in the past decade and sustain development through the next decade (Chang, Cross, Hang, Mei, Mulder, Shackelford & Chang, 2000). This resulted in the inclusion of ACM/IEEE recommendations for Net-Centric computing as a part of the Computer Science Undergraduate Body of Knowledge (ACM2004, 2004). With respect to this the ACM notes:

> Today, networking and the web have become the underpinning for much of our economy. They have become critical foundations of computer science, and it is impossible to imagine that undergraduate programs would not devote significantly more time to this topic. Modern networking technology enhances everyone's ability to communicate and gives people throughout the world unprecedented access to information. In most academic programs today -- not only in computer science but in other fields as well -- networking technology has become an essential pedagogical tool (ACM, 2001, p.12).

Accredited curriculum therefore is typically the basis of computer science courses within Australian and other universities worldwide. Furthermore, similar programs, particularly those based on ACM recommendations, have typically been adopted and recommended by the Australian Computer Society (ACS) in the design of tertiary courses in Australian universities (Hughes, 2003). However, even with accredited curriculum, there are other problems as noted by the British Computer Society (BCS):

> Computer systems are getting increasingly complex and in some cases there is no overall control or knowledge of exactly how they work or how they are performing. Such huge increases in complexity throw into question traditional engineering techniques for system development. These are based on breaking a problem into sub-problems and those into sub-sub-problems (BCS, 2004, para 2).

This introduces another issue, which is the use of scientific methods in the design of computer science curricula. Again according to the BCS:

> Computer science graduates are not sufficiently educated in the breadth of various scientific paradigms. They will need to be literate in related sciences as computer science becomes a field that is essentially horizontal: they need to understand the physics, biology, economics and social pressures of complex adaptive systems which might be working on a global scale. They will need to understand how to approach and investigate problems scientifically, how to compare two scientific methods (BCS, 2004, para 2-3).

Computer science faculties have been facing additional challenges in curriculum design. West notes:

> Even with increasing enrolments, the number of graduates in computer science and information systems has been inadequate to meet worldwide industry demand (West & Bogumil, 2001, p.34).

Stein and Criag point out the need for new curricula as part of Information Technology courses:

> The dot.com generation enters the university with an intensive education in technology. Experiences in Australian universities reveal that incoming students exhibit increased computer knowledge, have more confidence in their skills, and use IT applications more extensively than prior generations (Stein & Criag, 2000, p.220).

In this context, Laurillard points out problems with teaching methods in computer network education:

> Teaching methods have not evolved to meet the needs of students and employers in these rapidly changing technical fields (Laurillard, 2002, p. 2).

A Western Australia based study by Maj and Veal found that computer technology may be managed as a modular system that demands skills other than those provided by the computer science curriculum (Maj & Veal, 2000a). Veal points out:

> Computer science students when they graduate may fail to meet the employer's expectation in this rapid changing world of IT technology (Veal, Maj, & Swan, 2000, p.3).

Another problem facing the design of computer network courses is that computer network technology tends to change faster than most educational institutions can respond (Abelman, 2000; Naraymann & Neethi, 2001).

1.1 ACM Guidelines

Computer Science is a relatively new discipline, and, given the rapid advances in technology, is subject to ongoing debate, development and fragmentation. The ACM / IEEE (2001) recommended several principles that should underline computer science curricula. These principles, listed below, address the problem of rapid change in technology and education pedagogies:

- System-level perspective: Graduates of a computer science program must develop a high-level understanding of the system as whole;
- Appreciation of the interplay between theory and practice: Graduates of computer science program must not only have a theoretical understanding of the discipline but also how the theory influences practice;
- Adaptability: Graduates of computer science program must possess a solid foundation that allows them to maintain their skills as the field evolves; and
- Utilization of concepts from many different fields: All computer science students must learn to integrate theory and practice, to recognize the importance of

abstraction, and to appreciate the value of good engineering design. (ACM, 2001).

Their recommendations suggest a bottom-up approach and the use of abstraction in order to control complex systems.

1.2 Abstraction

With regards to abstraction, the IEEE defines it as: "*A view of a problem that extracts the essential information relevant to a particular purpose and ignores the remainder of the information*" (IEEE, 1983, p.47). Rumbaugh et al define abstraction as: "*Abstraction is the selective examination of certain aspects of a problem. The goal of abstraction is to isolate those aspects that are important for some purpose and to suppress those aspects that are unimportant*" (Rumbaugh, Blaha, Premerlani, Eddy & Lorensen, 1991). Furthermore, the ACM notes:

> Levels of Abstraction: the nature and use of abstraction in computing; the use of abstraction in managing complexity, structuring system, hiding details, and capturing recurring patterns; the ability to represent an entity or system by abstraction having different levels of detail and specificity (ACM, 1991, p. 75).

As noted by Jackson and Chapin abstraction can be applied to computer science courses:

> From an educational point of view, our experience suggest that undergraduate computer science courses should emphasize basic notions of modularity, specification, and data abstraction, and should not let these be displaced by more advanced topics, such as design patterns, object oriented methods, concurrency, functional languages and so on (Jackson & Chapin, 2000, p. 69).

Abstraction is important because it permits the simplification of complex situations to a level where analysis and understanding can take place. A good abstraction is one that emphasizes details that are significant to the reader or user and suppresses details that are, at least for the moment, immaterial or diversionary (Shaw, 1984). Due to rapid changes in technology one needs to develop new high-level models that can support abstraction in computer science. A report titled "*BCS Thought Leadership Debate IT Systems: Scale, Complexity and Risk*" by the British Computer Society notes:

> Computer systems are getting increasingly complex and in some cases there is no overall control or knowledge of exactly how they work or how they are performing (BCS, 2004, para 2).

Furthermore Clements notes:

> While the knowledge base academics must teach is continually expanding, only a fraction of that knowledge can be taught during a student's time at university. Consequently, academics must continually examine and update the curriculum, raising the level of abstraction (Clements, 2000, p. 11).

In order to avoid problems with the underlying technology there are advantages in creating models based on the principles of abstraction, so that they are not dependent on the low level details (Maj & Veal, 2000b). Such abstract models could be used to respond to the changes in computer and network technology (ACM, 2001; Ramsden, 1992; Staggers & Forrcio, 1993). Floyd notes:

> We want models that apply to the Internet of the future, as well as to the Internet of today. Due to the Internet's vast heterogeneity and rapid rate of change, we must pay close attention to what seems to be invariant and what is rapidly changing, or risk building dead-end models (Floyd & Kohler, 2003, p.30).

Success in a given field requires a mastery of the basic principles and in-depth knowledge of all component areas (Lovett & Greenhouse, 2000). This theme is developed by Robbins who notes that:

> Science is not merely about applying rules but engaging with creatively, especially in Computer Science, where we are not merely observing the rules of nature but developing our own. Science teaching needs to get back to a more objective approach. Conversely, increasing accessibility and producing creative, autonomous individuals puts the emphasis back on subjective needs (Robbins, Schagaev, & Yip, 2003, para 1).

Additionally the ACM notes:

> Enduring computer concepts include algorithms, complexity, machine organisation, information representations, modelling and abstraction…. While skills are fleeting, fundamental concepts are enduring and provide long lasting benefits to students, critically important in a rapidly changing discipline (ACM, 2001, p.70).

In addition to curriculum content it is important to look at the teaching of computer and network technology.

1.3 The need for models based upon Abstraction

Many aspects of computer science education are based upon abstraction. For example, when discussing the hardware used in digital systems, Veal notes:

> Digital techniques and modelling provide an abstraction that is independent of the underlying details of semiconductor switching. Such combinational or sequential circuits can be described without the complexity of their implementation in different switching technologies such as Transistor-Transistor Logic (TTL), Complementary Metal Oxide Semiconductors (CMOS), etc. Similarly details of semiconductor switching may be modelled using abstractions independent of the underlying details of quantum mechanics. Computer technology can therefore be described using a progressive range of models based on different levels of detail e.g. semiconductors, transistors, digital circuits. (Veal, 2004, p.238).

Within computer technology education, the ACM suggests a bottom-up approach. In essence, this means that each level in the system hierarchy can use the principle of abstraction. For example, digital techniques are taught via the abstraction of Boolean algebra. Furthermore, combinational and sequential circuits may be modelled as registers and functional blocks. Having a background in digital electronics and microprocessor architecture and programming may provide student with more fundamental understanding of new technologies as they appear on the scene (Beeson & Gay, 2000). However, with a bottom-up approach students may find it difficult to see the overall picture by being initially introduced to large quantities of underlying detail (Staggers & Forrcio, 1993), This may result in a large conceptual gap that students would need to bridge in order to understand the operation of an actual computer. However, due to rapid changes in computer and network technology it is recognised that a new high-level abstraction model is now needed (Maj & Veal, 2000b; Maj, Veal, & Boyanich, 2001).

In order to address the problems of complexity, rapid change in computer technology, and to meet the needs of students and employers, there is a need to create models based on the principles of abstraction as recommended by ACM /IEEE.

What may therefore be needed are new high-level models that are not dependent upon low-level technical details, since it is this detail that is subject to rapid changes. Although this may not always be possible, one strategy would be to avoid complex models based upon foundations of low-level technologies. A model is required that can help to bridge this gap.

1.4 Modelling characteristics

According to Maj (2000), characteristics considered of particular importance when modeling computer and networking technology are that models are:

- Diagrammatic;
- Easy to use; and
- Able to control detail by means of hierarchical top-down decomposition.

According to Cooling (1991) there are two main types of diagram: high-level and low-level. High-level diagrams are task oriented and show the overall system structure with its major sub-units. Such diagrams describe the overall function of the design and interactions between both the sub-systems and the environment. The main emphasis is 'what does the system do' and the resultant design is therefore task oriented. For example, the Data Flow Diagram (DFD) model (Hawryszkiewycz, 2001) enables a complex system to be partitioned (or structured) into independent units of an amenable size so that the entire system can be more easily understood. It is possible to examine a system in overview and with increasing levels of detail, whilst maintaining links and interfaces between the different levels. DFD's are not only simple, but also graphical; hence they serve not only as documentation but also as a communication tool (Pressman, 1992). DFD's are therefore a top-down diagrammatic representation of information flow within a system, and are a means of defining the boundaries and scope of the system being represented, checking the completeness of the analysis and providing the basis for program specifications. This technique is relatively simple to use, yet powerful enough to control complexity during the analysis and design of both small and large systems.

Computer models are often based on conceptual models (Maj, 2000; Menasce, Almeida, Fonseca & Mendes, 1999). Models should be easy to use and understand. They should not depend too much on the underlying technology, and should provide the user with a more meaningful way of looking at the system. Finally, models should provide the ability to hide information, yet also provide top-down decomposition so that they allow the user to construct meaningful information.

An important aspect of models in the past, one that has not received much attention, is their ability to adapt to change. Software engineering studies have shown that over half of software development effort is spent on maintenance (Kain, 1999). When models are embedded in the software applications it is likely that considerable effort will be needed to maintain the models.

Abstraction is one of the many approaches used for understanding complex computer and network technology (Maj & Veal, 2000b; Veal, 2004). Models based upon the principles of abstraction are required in the teaching of computer and network technology.

1.5 Modelling Approaches

There are two methods used to address the problem of complexity within computer and network technology: the two methods are the bottom-up approach and the top-down approach. Both of these methods address complexity and allow the use of abstraction as recommended by the ACM curriculum (ACM, 1991).

The bottom-up approach helps in identifying all relevant resources associated with the system. Furthermore, it also provides the ability to represent an entity or system by abstraction having different levels of details (ACM, 1991). It should be noted that the ACM syllabi outlines are presented in a bottom-up sequence (ACM, 2001), where the Computer Architecture syllabus and the Architecture and Operating Systems syllabus both commence with digital logic (ACM, 2001). In describing the "hardware first" approach the ACM note that: "*The first course in the sequence covers the computer from the bottom-up*" (ACM, 2001, p.34). Furthermore, with respect to computer networking, the bottom-up approach may include how data is packaged on physical media, such as on copper or optical cables and, in recent times, wireless. This approach provides detailed explanations of how communication works between two nodes e.g. two PCs. Error detection and correction are also provided in this approach. Furthermore, it may also include an introduction to the different protocols communications, which provide guaranteed, reliable communication over error-prone network connections. The advantages of the bottom-up approach within computer and network technology are that it:

- Provides fundamental understanding of how communication works.
- Uses concepts which are consistent with students' learning cycles.

More and more computer networking courses are using this approach (Commer, 2004). However, with a bottom-up approach students may find it difficult to see the overall picture by being initially introduced to large quantities of underlying details (Staggers & Forrcio, 1993).

The top-down approach to computer networks focuses on network applications. It helps in controlling complexity, structuring systems and hiding details. Furthermore, it also provides information about packet switching, and the web (Kurose & Ross, 2004). Management Information courses tend to favour this approach (Huy & Chae, 2004). Furthermore Gupta and Wachter (1998) agree that Management Information students require effective and efficient applications of IT to solve business problems.

Both the bottom-up and top-down approach can be used in controlling complexity, structuring systems, hiding details, and capturing recurring patterns. Both these methods can be integrated in the teaching of computer and network technology.

1.6 Summary

Rapid changes in Information Technology place significant demand on curriculum content. Professional bodies such as the American Computer Machinery (ACM), the Australian Computer Society (ACS), the British Computer Society (BCS), and Institute of Electrical and Electronics Engineering (IEEE) are working actively to create a model curriculum to address these concerns.

The principles of abstraction can be applied to computer and networking technology to help provide students with common fundamental concepts regardless of the particular underlying technological implementation. These principles are designed to avoid the rapid redundancy of a detailed knowledge of modern computer and networking technology. However, models employing both top-down and bottom-up methods are needed. Teaching methods need to include good conceptual models that will assist students to get a sound grasp of the complex technology they are studying.

An overview and analysis of the top-down and bottom-up models are discussed in the next section.

2. DETAILS OF MODELLING METHODS

The importance of abstraction has been very significant in understanding complex technology. The following section introduces and evaluates a range of models used in the design and development of Information Technology (IT) systems.

In order for any IT system such as E-business to provide a prompt and possibly worldwide service they must be based on a complex infrastructure of hardware and software. Furthermore, this system must provide a Quality of Service to meet the Service Level Agreement.

There exist a wide range of methods that can be used for the analysis and design of IT systems. Some methods are based on a rigorous mathematical foundation such as the Vienna Development Method (VDM). Other, less formal methods provide a clearly defined structured framework with associated tools and techniques. Uniquely, soft system methodologies attempt to capture the complexity of human interactions with IT systems. Regardless of which type of method is used the final product must be a fully functioning IT system that meets the requirements of the Service Level Agreement (SLA).

To illustrate the use of models, e-business is selected for the following reasons. E-business applications depend upon the core of IT, which involves the use of:

- Computer hardware
- Software
- Computer Networking

As a consequence of the large volume of e-business transactions, computer and network technology applications are putting greater pressure than ever on Information Technology (Bailey & Bakos, 1997; Bakos, 1998). This has resulted in system and network overload. Organizations need to estimate load, especially as networks proliferate and take over more of the critical data processing. E-business applications have a tendency to saturate server and network resources quickly (Dennis & Hofer, 2000). Capacity planning is the process of predicting when future load levels will saturate the system and of determining the most cost effective way of delaying the onset of system saturation (Menasce, Almendia, & Dowdy, 1994).

dequate capacity planning if not undertaken can easily put companies in an endless cycle of upgrades and migration (Menasce & Almedia, 1999).

E-business applications have encountered transaction delays as systems and computer networks become overloaded (Devlin, Gray, Laing, & Spix, 1999). Surveys and studies indicate that slow downloading is the most frequently cited reason for online customers leaving a vendor’s site and looking for another site instead (Bakos, 1998). As more online businesses face this problem, computer scientists are investigating the issue to evolve meaningful quantifications that will enable prompt service and meet desired Quality of Service (QoS) to provide a level of service that users of commercial web sites will find acceptable.

The Software Development Life Cycle (SDLC), as defined by Pressman (2000) and Pfleeger (2001), comprises systems planning, design, implementation, testing and maintenance, and thus provides an analytical framework for issues in the systems domain. A company may have placed great effort into an online business strategy and business process re-engineering scheme, yet may fail to investigate key SDLC performance measures. The following areas have been routinely identified as sources of systems level failure in online business systems:

- User interface design;
- Operational functionality;
- User testing; and
- Information architecture (Nielsen, 2000).

Having analyzed these common causes of system failure, Jakob (2000) developed a theory of usability that emphasizes the importance of simplicity in design, and comprehensive user testing. However, most time and expenditure is spent on site design and business requirements. According to Shklar:

> Sites have been concentrating on the right content. Now, more of them specially e-commerce sites realize that performance is crucial in attracting and retaining online customers (Shklar, 1998).

The problem associated with infrastructure design is non-trivial. According to Fenik: “*Being able to manage hit storms on commerce sites requires more then just buying more plumbing*” (Fenik, 1998).

Capacity planning relates to the prediction of when the future capacity of an existing system will become insufficient to process the required workload with a given level of performance (Ferrari, Serazzi, & Zeigner, 1983). A key objective of the capacity planning process is to ensure that adequate capacity is provided without unacceptable reductions in system service levels.

In the final analysis, capacity planning relates to predicting when the service level will be violated as a function of the work load evolution, as well as determining the most cost-effective way of avoiding system saturation (Menasce, Virgilio, & Almeida, 2000).

A critical aspect of any e-business application is its ability to guarantee Quality-of-Service (QoS) to its networked users, and to provide the same guarantees on internal data delivery as it expects from outside sources. A Service Level Agreement is a contract between a service provider and a customer in which the service provider promises to deliver a specified level of performance across the network (Dennis & Hofer, 2000).

Several metrics are used to assess SLAs. Among them are availability and application response time. According to a study conducted by Cahners (1997), most business executives rank response time as the second most important service factor after availability. Sluggish response time is a major problem facing e-business (Menasce, Almendia, & Dowdy, 1994). Load testing is another important metric guaranteed in Service Level Agreement. Key principles of successful e-business are the ability to be open to all potential consumers when they come to browse and buy products, and to be able to handle all service requests and demands. Performance bounding analysis (Denning & Buzen, 1978; Lazowska, Zahorjan, Scott & Graham, 1984) is a key technique used for scalability problems in e-business. It is hard to estimate traffic; a designer of e-business sites must know a priori of limitations in the system (Menasce, Barbara, & Dodge, 2001).

In the final analysis it is essential to map e-business requirements to technical specification for hardware selection. To address the problem for hardware selection the following section provides an overview of some of the modelling methods and techniques used in the design of computer and network technology.

2.1 Systems Analysis and Design

Systems Analysis and Design mainly deals with the software development activities. The main objectives are:

- Understanding of the system;
- Understanding the different phases of system development life cycle;
- Design and Implementation.

There are a wide range of system analysis methods. A method consists of phases or stages that in themselves may consist of sub-phases. Systems analysis and design methods include: waterfall (Royce, 1970), prototyping (Naumann & Jenkins, 1982), incremental (Gibb, 1988), and spiral (Boehm, 1984). Regardless of the underlying theme of each information system, all methods must provide techniques for modelling data, processes and system functions. The majority of the structured methods have well-established tools and techniques for modelling processes such as Data Flow Diagrams (DFD) and Entity Flow Diagrams (EFD). However, none of the structured methods analyzed have any simple technique for deriving the hardware specifications in order to meet service level agreements. Maj et.al. (2001b) conducted an in-depth investigation of the Structured Systems Analysis and Design Method (SSADM) (Ashworth & Goodland, 1990) as a method for developing an information system. SSADM is mandatory for United Kingdom (UK) central government software development projects. SSADM provides tools that allow the estimation of storage requirements. From the Composite Logical Data Design (CLDD) and Logical Design Volumes (LDV), detailed information about the data volumes may be extracted. It is possible to obtain detailed information about data space for each data group, volumes of each data group, volumes of relationships, variance of volumes over time, etc. Furthermore, SSADM employs a range of different, heterogeneous performance metrics that includes Millions of Instructions per Sec (MIPS), CPU time, disk access time, number of instructions per database call, etc. Other than this, there are no simple tools or techniques that can be used to control complexity and hence model the hardware required to support infrastructure.

It was then decided to investigate the design approach used by commercial vendors such as Oracle and Cisco.

2.1.1 Database design using Oracle

Oracle provides technical specifications for the minimum platform to run the database e.g. Uni-processor Pentium 300MHz, 128Mbytes RAM and 1.2 Gbytes Hard Disc space for the database installation. Other architectural solutions include: Symmetric Multiprocessor Systems (SMP), Clusters, Massively Parallel Processing Systems (MPP) and Non-uniform Memory Access Computers (NUMA) Table 2.1 (Greenwald, 1999). However, beyond this, little guidance is provided for the selection of hardware.

Ranking	Scalability	Manageability	Availability	Price
Best	MPP	Uni-processor	Cluster	Uni-processor
	Cluster	SMP	MPP	SMP
	NUMA	NUMA	NUMA	Cluster
	SMP	MPP	SMP	NUMA
Worst	Uni-processor	Cluster	Uniprocessor	MPP

Table 2.1: Processor Architectures (Greenwald, 1999)

Standard Entity Relationship Diagrams (ERDs) are used to determine the number of database tables needed, from which it is possible to calculate the upper and lower table-space limits for the users, applications, roll backs etc. Hence the size of the hard disc can be calculated. The main emphasis here is database sizing for current and future needs. It is then possible to tune the Oracle database using performance views such as “V$SYSTEM_EVENT” and “V$SESSION_WAIT”. Solutions to Input / Output (I/O) bottlenecks include disk striping technologies to spread I/O across spindles (e.g. Redundant Array of Independent Disks (RAID) technology), use of table-space to segregate and target different types of I/O, and distribute system overheads evenly across spindles.

Following this it is possible to optimize the database using either cost-based or rule-based criteria. The cost-based optimizing depends upon logical reads, CPU cycles, and number of network transactions. The cost-based optimizer has access to statistics relating to the tables and indexes, such as the size of the table, the minimum and maximum value in indexed columns. The cost-based optimizer uses the "INIT.ORA" parameter "DB_FILE_MULTI_BLOCK_READ" to estimate the number of I/O's required to perform a full table scan.

It can be concluded that Oracle does provide techniques for database optimization and tuning but it fails to provide any simple approach to map specification to hardware selection.

2.1.2 Cisco Three Layered Model

Cisco Systems has defined a hierarchical model know as the hierarchical internetworking model (Cisco, 2002, 2004) (Figure 2.1). The model was designed to simplify the task of building reliable, scalable, and hierarchical Internetworks. The model focuses on the three areas of the network:

Core layer: This layer is considered the backbone of the network and includes the high-end switches and high-speed cables, such as fiber cables. The core layer is not responsible for routing traffic at the Local Area Network (LAN). This layer is concerned with speed and reliable delivery of packets between different locations on the network. High speed WAN and other inter campus links are employed here, but little or no traffic filtering should take place. Generally, devices that operate at layer 3 or layer 2 of the Open System Interconnection (OSI) model would be used here.

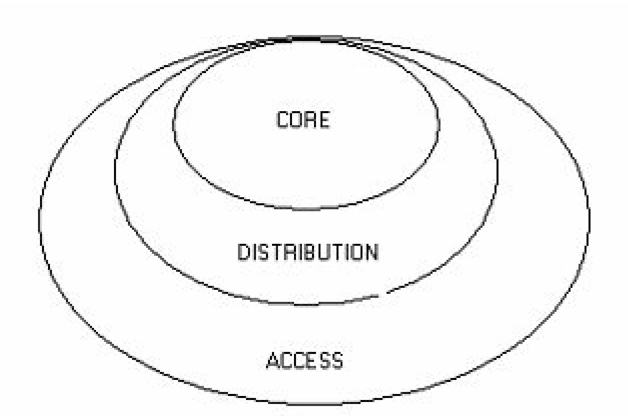


Figure 2.1 Cisco three layered model (Cisco, 2002)

Distribution layer: This layer is implemented at Layer 3 in the Open System Interconnection (OSI) model. The emphasis is on the routing of packets between different subnets and Virtual Local Area Networks (VLANs) in the enterprise (Doyle, 2002). This layer is also called the Workgroup layer (Grice, 2001a). Practical manipulation (policy based routing) is also performed by this layer.

Access Layer: This layer includes hubs and switches. This layer is referred to as the desktop layer (Doyle, 2002) because it focuses on connecting client computers. The main responsibility is the delivery of packets to end user computers.

Although the Cisco three layered model provides advantages in terms of design and implementation it also provides simple techniques for equipment selection. However, it does not provide a simple quantitative model for hardware selection.

The above section provided examples of various system analysis and design methods used for information systems. It was then decided to investigate some more quantitative models used in the design of computer and network technology.

2.2 Analytical Modelling (Queuing theory)

Analytical modelling has been described as the use of a set of equations describing the performance of a computer system (Buzen, 1984). Furthermore, it describes a collection of measured and calculated behaviours of different elements over a period of time, which can include workloads, hardware, software and the Central Processing Unit (CPU) (Caliri, 2000). Analytical modelling using queuing theory (Klienrock, 1975) describes IT systems and sub-systems as resources called servers (Lazowska et al., 1984), and the load on each resource as a series of jobs. Jobs not currently being serviced wait in a queue until the current jobs are completed. The two principal techniques used are operational and stochastic analysis.

In operational analysis there are no assumptions about the distribution of the times required to service jobs and the distribution of job arrival times. Using operational analysis (Denning & Buzen, 1978) it is possible to calculate utilization, traffic intensity, and average number of jobs in a queue (Bailey, 1999; Little, 1961). Operational analysis (Sevcik & Mitrani, 1981) can be applied to any system or part of a system and provides a high-level metric of job flow.

Stochastic analysis (Buzen, 1978), however, assumes probabilistic (stochastic) distribution of job arrival and service times. Using stochastic analysis it is possible to write flow equations for single queue, single server systems and more complex single queue, multiple server systems (Kleinrock, 1975). Analytical modelling has strong mathematical foundations that are actively being researched, and has an associated large body of literature. However, it is a model whose accuracy is dependent upon assumptions. In Lilja's terms:

> It is important to bear in mind, however, that the conclusions drawn from a queuing analysis must eventually be applied to a real system if they are to be of any use. The results of this analysis will be accurate only insofar as the assumptions made in the analysis match the system's characteristic. This means that, if the characteristics of the actual system only approximate some of the assumptions made in the queuing analysis, then the conclusions drawn from the analysis will (at best) only approximate what will happen in the actual system (Lilja, 2000, p.20).

For very large systems, queuing analysis is an important technique where the effort required in measuring performance is proportional to the cost of making the wrong decision. However, such considerable mathematical rigor does not lend itself to general use, and may not result in simple, cost-effective solutions for small to medium-sized enterprises.

The application of a specific technique depends upon the problem domain, which may in turn be described as, the cost of getting it wrong. Furthermore, all of these techniques model job flow and hence use performance metrics that include jobs per second, response time, service time. However, hardware is chosen using technical specifications. Whilst performance analysis techniques employ abstraction (i.e. queues and services) there appear to be no guidelines for relating these abstractions to technical specifications. It remains problematic to relate these different units to the performance of an IT system that consists of a heterogeneous collection of different technologies, each with their own performance metrics. In effect it is relevant to ask, 'How can job flow be directly related to technical specifications?' This is important, because it is on this basis that equipment is actually selected.

Bandwidth is an important consideration in the design of Information Technology systems (Baker, 2002; Lea, Tsui, Li, Kwan, Chan, & Chan, 1999; Maj &

Veal, 2001). Bandwidth management, often used interchangeably with the term traffic shaping, can be defined as the appropriate allocation of bandwidth to support application requirements (Goldman & Rawles, 2001, p.548).

2.2.1 Bandwidth-share Model

The design of today's Internet must take into account congestion, traffic flows etc. Congestion control refers to a mechanism that enables the source to match its transmission rate to the available bandwidth (Li, Shor, Walpole, & Pu, 2001). One way of modelling the Internet is to work with the Bandwidth-Share model (Giuli & Baker, 2003; Li et al., 2001). The Bandwidth-Share model works on the principle of minimum share allocation (Giuli & Baker, 2003). Among the various bandwidth share models one can use either deterministic models (Fage et al., 2000; Morgan & Dennis, 2001; Padjen, Lammie, & Edwards, 2002), or stochastic models (Lilja, 2000; Menasce & Alemeida, 2002; Tanenbaum, 2003). Lai and Baker note that:

> Deterministic models are typically easier to work with mathematically than stochastic models, enabling us to find an analytical solution rather than a numerical one. Unfortunately, a deterministic model implies modelling with absolute certainty and many things cannot be known with enough certainty to make this practical" (Lai & Baker, 2000, p. 52).

To allow for more complex configuration in the modelling of networks and internetworks one needs to develop models that can describe the number of flows on all routes, and the performance of the data transfer, depending upon how link capacity is allocated between competing flows (Bonald & Proutiere, 2003).

Hudson et al note the importance of bandwidth utilization with respect to computer and network performance:

> Reducing the number of devices on a segment, reducing the number of protocols in use on the segment, disabling bandwidth intensive applications of protocols, such as those that support video or audio streaming. Relocating the systems consuming the most bandwidth on the segment (Hudson, Caudle, & Cannon, 2003, p.118).

The use of the Bandwidth-Share model is an important consideration in measuring the performance of computer and network technology. Bandwidth-Share model are generally used for large systems such as the Internet. The use of a strong mathematical approach may not be suited for the design of computer and network technology for small to medium size enterprises. The problem remains of mapping bandwidth to technical specification.

Apart from bandwidth, other performance analysis models are used in the design of computer and network technology. The following section provides the strength and weakness of these models.

2.2.2 A Performance Model

Performance models are useful for infrastructure design, resource allocation and the fine-tuning of systems. A Performance Model represents how systems resources are used by different workloads and to capture the main factors determining the actual system performance (Menasce et al., 2000). A Performance Model is based upon a sub-set of queuing theory (Klienrock, 1975) called operational analysis (Denning & Buzen, 1978). A performance model could help in aiding understanding of the quantitative behavior of complex systems (Menasce et al., 2000). In the design of large infrastructure systems, performance modelling is essential. A Performance Model can assist in providing answers when making changes to a production system. Performance Models advantages include methods of quantitatively analyzing the system rather then relying on ad hoc procedures or heuristics. This model does not provide a simple method for hardware selection on which these systems run. Examples of performance models include Zipf’s law (Zipf, 1949), Little’s law (Little, 1961). These laws are widely used in performance analysis of computing systems. The success of E-business applications is dependent upon web performance. The performance models discussed above are currently applied to the Web. The following section provides an overview of Web performance.

2.2.3 Web performance

Web performance depends upon several factors: hardware platforms, operating systems, server software, network bandwidths and workloads. There are various well known methodologies for performance evaluation of computer systems (Menasce, Almendia, & Dowly, 1994). However, the World Wide Web (WWW) has some unique characteristics that distinguish it from traditional systems (Almedia et al., 1996; Almeida, Virgilio Almeida, & Yates, 1997; Arlitt & Williamson, 1996;

Mogul, 1995). Firstly, the number of WWW clients is in the range of tens of millions and rising. The randomness associated with the way users visit pages makes the problem of workload forecasting and capacity planning difficult (Menasce, Almendia, & Dowdy, 1994). The web is also characterized by a large diversity of components: browsers and servers running on a variety of platforms with different capabilities. To determine if memory, Input / Output (I/O), or the Central Processor Unit (CPU) is the bottleneck for a web server depends upon the nature of the workload. However, there will continue to be a demand for server architectures that perform well under heterogeneous workloads. There is a need for new operating system implementations that are designed to perform well when running on web servers (Mogul, 1995).

Among the various models, there are currently two models used to model user characteristics which can then be used to model Web performance. Menasce et al (2000) proposed the Customer Behaviour Model and the Client Server Interaction Graph (Menasce et al., 2000) which are discussed in the following section.

2.2.3.1 A Customer Model

Customers interact with e-business sites through a series of consecutive and related requests during a single visit, which is called a session (Menasce et al., 2001). During a session a customer can login, browse, search etc. Different customers have different patterns of navigating through an e-business site and this can increase the load on the server handling these sessions.

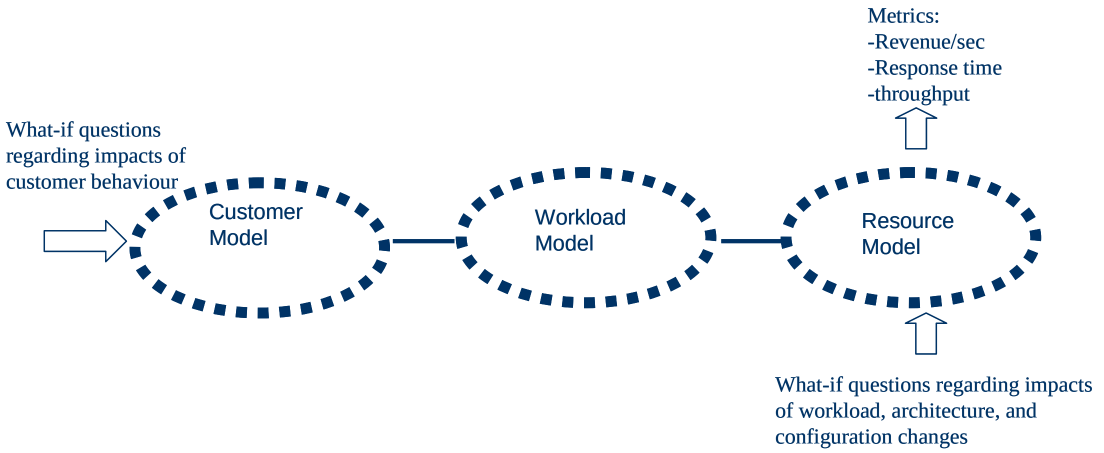


Figure 2.2 Customer, Workload, and Resource Models (Menasce et al., 2000, p.43)

Figure 2.2 shows how a customer model can relate to workload, which can then be used to define the resource model. The workload model describes the workload intensity and the demand on the various resources (e.g. processors, I/O, network) that make up the site (Haar, 1999). The resource model represents the various resources of the site and captures the effect of the workload model on these resources. Understanding customer behavior is used for proper sizing of the resources of an e-business site as well as achieving business goals (Stohr & Kim, 1998a). The Customer Behavior Model Graph (CMBG) model captures navigational patterns, frequency of access and time spent on the site (Menasce et al., 2000). For example during a web site visit a customer may browse the site, followed by search and finally may register on the site to buy a product. The CMBG models represent a graphical notation of user interaction with that site. Furthermore, this model helps to provide information about what resources are required to operate an e-business infrastructure such as application server, payment server, web server etc. The down side of this model is that it lacks a clear definition of CPU, Disk and Memory requirements which can be mapped using the SLA to technical specifications.

In order to provide logical interaction between clients and various servers Client Server Interaction Diagrams (CSID) are used. The following section provides an overview of CSID.

2.2.3.2 Client Server Interaction Diagrams (CSID)

A CSID describes the possible interactions between a client and the servers (Figure 2.2) that implement a transaction.

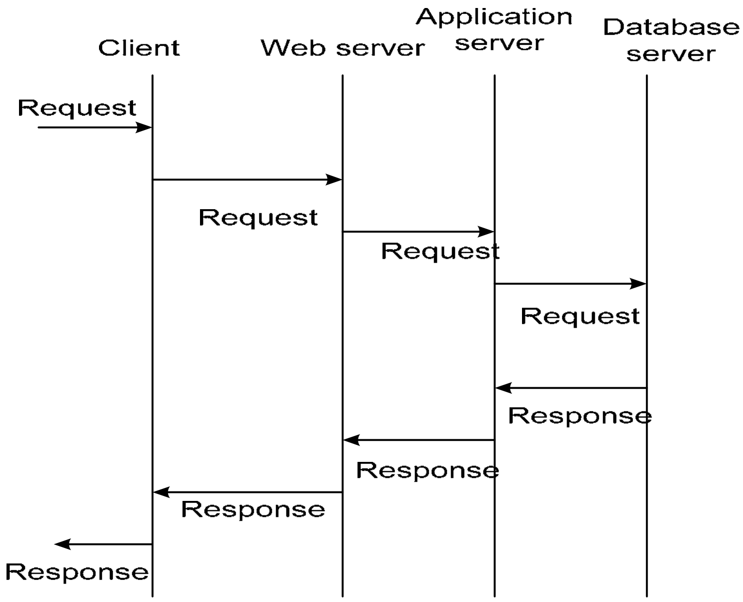


Figure 2.3 Client Server Interaction (Menasce et al., 2000, p.74)

A Client-Server interaction starts when a client process makes a request to a server process, called primary server, and ends when the client process receives a reply from the primary server. The primary server (Web Server) may need assistance from other servers called secondary servers (Application and Database servers). A CSID must be specified for each transaction. The CSID shows all possible Client-Server interactions for that particular transaction. CSIDs are useful in answering quantitative questions about the execution of e-business functions. CSIDs also provide the following benefits:

- Identifying all the nodes in the CSIDs associated with the server;
- Computing the probabilities associated with each node (Menasce et al., 2000, p.89).

The CMBG and CSID are techniques that can be used to capture navigational patterns of customers during site visits and hence obtain quantitative information on workloads. Though these techniques are useful, a wide range of different performance metrics and benchmarks are still used. For example hits per second, page views per day, click-through, transactions per second, CPU time, and processor demand. These models provided no simple technique but a quantitative approach for resource modelling of the actual hardware (e.g. Processor, Memory, Hard disk) which can support the e-business infrastructure.

In addition to these models, benchmark metrics are widely used. The following section provides a brief description of benchmark measurement and its use in e-business infrastructure applications.

2.3 Benchmark Measurement

Benchmarks are standard metrics which are used in measuring the scalability and performance of a given hardware or software system. For example, the Adaptive Computing System (Sanjaya, Chirag, & John, 2000) is a collection of benchmarks that focus upon specific characteristics from the start of a computation to its completion. Benchmarks evaluate the ability of a configurable computing infrastructure to perform a variety of different functions. SPECWEB and TPC-W (Smith, 2000) are notable benchmarks in the e-business environment**.** These benchmarks come close to representing the complex environment of an e-business workload. Benchmark programs are used for evaluating computer and network systems. Different end user applications have very different execution characteristics; hence there exists a wide range of benchmark programs. The four main categories are: science and engineering (for example include Whetstones, Dhrystones, Livermore loops, NAS kernels, LINPACK, PERFECT club, SPEC cpu), Transaction Processing (for example: TPC-A, TCP-B, TPC-C), server and networks (examples include; SPS/LASSIS, SPEC web) and general use (examples include; AIM Suite III, SYSmark, Ziff-Davis PC Benchmark). In February 2000 (Smith, 2000) the Transaction Processing Council introduced the TPC-W to simulate the workload activities of a retail store web site. In the TPC-W an online bookstore is used to provide a real life load situation and the user was emulated via a remote browser that simulated the HTTP traffic that would be seen by a real customer using the browser. As e-business sites have transient saturation, it is difficult using these benchmarks to get reliable estimates of the actual load generated on the web servers. Furthermore, the benchmarks are in themselves problematic because it is not possible to relate the technical specification directly to the metrics used in the SLA. Benchmarks currently in use fail to measure the Web performance characteristics, whilst others may even be incorrectly interpreted (Humphrey, 1990; Lilja, 2000).

2.4 Summary of Modelling Methods and Design

A detailed analysis and investigation was conducted into a wide range of structured models. From the analysis and evaluation of all the models and techniques, none provided a simple model that could used to model the heterogeneous devices such as CPU, I/O, memory etc. used in the construction of computer and network technology infrastructure. Performance analysis techniques (e.g. operational and stochastic analysis) exist and are essential for large, expensive installations. However, their considerable mathematical rigor does not lend them to general use, and for Small to Medium Enterprises (SMEs) this may not represent simple, cost effective solution. Furthermore, most hardware is chosen using technical specifications. Whilst performance analysis techniques employ abstraction (i.e. queues and service centres) there appear to be no guidelines for relating these abstractions to technical specifications. The use of benchmarks remains controversial. Techniques to capture customer navigational patterns using CMBG's and hence obtain quantitative data about workloads have been successfully used and related to the new Transaction Processing Council Web (TPC-W) benchmark. However, they employ a wide range of metrics that cannot simply be related to technical specifications, which are the basis of equipment selection. To address these problems a new modelling technique was developed, Bandwidth Nodes (B-Nodes) (Maj & Veal, 2000a; Maj & Veal, 2000b).

2.5 Introduction to the B-Node Model

Computer and network technology can be described using a progressive range of models based on different levels of details such as semiconductors, transistors, digital circuits, registers, device controllers, clients, servers etc. Each model is valid in the context in which it is used. All such models are designed to progressively hide and hence control detail and yet still provide sufficient information to be useful for communication, design and documentation. This is in keeping with the ACM/IEEE Computing Curricula 1991 in which abstraction is a recurring concept fundamental to computer science (Tucker et al., 1991).

Computer and network technologies are concerned with a wide range of heterogeneous devices (Hard disk drives, electronic memory, CDROM etc) and associated heterogeneous units of measurements (MHz, milliseconds etc). Comparison of the relative performance of each of these different devices is therefore difficult. The fundamental principles of measurement science include: fundamental unit/s (Mass, Length and Time), derived units (e.g. velocity), decimal scaling system and the use of meaningful units relevant to human dimensions and perceptions (e.g. km/hr). B-Nodes are a quantifiable data source/sink capable of, to various degrees, data storage, processing and transmission. This approach allows the performance of B-Node to be assessed by a simple, common measurement: bandwidth[1]. Mbytes/sec[2] is defined as the fundamental unit. In effect the use of a common fundamental unit allows heterogeneous technologies (microprocessor, hard disk drive etc), with heterogeneous units (MHz, rpm etc) to be directly compared using B-Nodes. Though useful, it may not be the best initial, user oriented unit. For general acceptance a unit of measurement must be easy to understand and should therefore be based on user perception of performance. Hence other units may be derived such as transaction per second, images per second, etc.

2.5.1 B-Node Applications

If a web server is modelled as a B-Node then the performance metric is bandwidth with units of Mbytes/s (Maj & Veal, 2001). The sub-modules of a server (microprocessor, hard disk, electronic memory etc) can also be modelled as B-Nodes, again using the same performance metric. The use of fundamental units (Mbytes/s) allows other units to be derived and used e.g. transactions per second (T/s). Assuming the messages in a client/server interaction contain 10kbytes each, the performance of each Information-Node can be evaluated using the unit of transactions/s (T/s) (Table 2.1).

Device	Units	Bandwidth (MBytes/s)	Bandwidth (T/s)	Load (T/s)	Utilization
Processor	400 MHz	1600	160k	250	<1%
DRAM	16 MHz	64	6.4k	250	4%

[1] Where Bandwidth = Clock Speed *Data Path Width * Efficiency (B = C x D x E) with the common units of Mbytes/s
[2] Mbytes/sec units can also be relevant to end users of e-business transaction, medical images, sound etc.

	(16 nsec)				
Hard Disc	60 rps	2.7	270	250	93%
CDROM	32X	2.3	230	250	>100%
ISA Bus	8 MHz	4	400	250	63%
Ethernet	100 MHz	11.25	1.1k	250	23%

Table 2.1: Utilization (Maj & Veal, 2001)

If the demand on this server is 250 Transactions/s it is a simple matter to determine both performance bottlenecks and also the expected performance of the equipment upgrades. From table 2.1 it is possible to determine that the hard disk drive, CDROM and ISA bus are inadequate as the utilization (U) is too high. The metric of T/s can easily be converted to the fundamental unit of Mbytes/s, which can then be used to determine the required performance specification of alternative bus structures, CDROM devices and hard disks. For example, this system would be improved by replacing the hard disk drive by one with a higher performance specification (rpm and higher track capacity). Furthermore, from the derived unit of T/s, other units such as Response Time may also be derived. Response time is simply the reciprocal of T/s.

The B-Node model provides abstraction by use of a top-down recursive approach; furthermore, it also provides a simple way of measuring the performance of the hardware running that infrastructure.

Although B-Nodes provide a simple technique for modelling infrastructure, conducting detailed experiments has proved to be problematic due to the complexity of the hardware and software. The E [3] factor in the bandwidth equation has proved at this point in time difficult to quantify because of the increasing complexity of both hardware and software within the PCs both because of the hardware and software.

On the other hand it is, relatively, easier to measure the performance of internetworking devices such as switches and routers. They operate on a simple Internet Operating System (IOS), and compared to a PC there are relatively few resources to manage on these devices. In addition, switches and routers are typically

[3] Bandwidth = **C**lock speed * **D**ata path width * **E**fficiency

state machines. State machines are systems whose state changes according to events that have occurred during the process.

Furthermore, State-oriented specifications include variables that change each time a state change occurs. The standard approach to documenting this type of specification is to use the Finite State Machine (FSM).

A FSM can be defined mathematically and generally are represented using a graphical notation. The graphical representation is the main design tool for FSMs, and is called the State Transition Diagram (STD). The STD shows the relationship between the system states and the events that cause the system to change from one state to another (Gibson, 2002). The FSM is one technique used for modelling communication protocols. According to Halsal:

> The three most common methods for specifying a communication protocol are state transition diagrams, extended event-state tables and high-level structured programs (Halsal, 1996, p.136).

He further states:

> Irrespective of the specification method, we model a protocol as a finite state machine or automaton. This means that the protocol – or, more accurately, the protocol entity – can be in just one a finite number of defined states at any instant. For example, it might be idle waiting for a message to send or waiting to receive and acknowledgment (Halsal, 1996, p.142).

A communication protocol can therefore be modeled as a protocol entity that can exist in one of a number of defined states. Similarly, Internetworking devices such as routers and switches can exist in different states. The principles used by B-Nodes can easily be extended to model internetworking devices.

2.6 Summary

There exist a wide range of methods that can be used for the analysis and design of computer and network technology. Some methods are based on a rigorous mathematical foundation. Other, less formal methods provide a clearly defined, structured framework with associated tools and techniques. Uniquely soft system methodologies attempt to capture the complexity of human interactions with IT systems. Regardless of which type of method is used, the final product must be a fully functioning IT system that can be used to model the system, and so demonstrate compliance with the SLA. However, a detailed analysis of wide range of top-down and bottom-up methods indicated the lack of a simple method for modelling computer and network hardware. B-nodes can be used for hardware modelling because of the following strengths:

- Bandwidth in Mbytes/s can be considered as a fundamental unit from which other units may be derived;
- B-Node performance, measured in bandwidth (Frames/s) is a user based, easily understood measurement;
- B-Nodes are diagrammatic, self-documenting, easy to use and control detail; and
- Using recursive decomposition B-Nodes can be used to model more complex computer technology devices.

However PC hardware and operating systems are complex and this makes obtaining accurate bandwidth measurements inside a PC difficult to obtain. A rigorous quantitative model based on the B-Nodes requires the “E coefficient” to be evaluated. However, this also proved to be problematic.

We have examined modelling methods which are based upon the top-down or bottom-up approaches. The following section provides an overview of how these models are integrated into the teaching of computer and network technology.

3. INTEGRATION OF DIFFERENT MODELS IN TEACHING OF COMPUTER AND NETWORK TECHNOLOGY

3.1 Different Approaches

There are different, equally valid approaches to teaching computer and network technology. Kurose et al (2002, p.104) identifies three contrasting methods:

1. A quantitative style, similar to the approach used in engineering courses versus algorithmic style, such as used in computer science education;
2. A hands-on laboratory approach versus a more traditional university style of lecture based teaching; and
3. A bottom-up versus a top-down approach

The quantitative engineering approach focuses on the design of computing elements and computer based systems. According to Srimani et al:

> Computer Engineering embodies the science and the technology of design, construction, implementation and maintenance of the hardware and the software components of modern computing systems and computer-controlled equipment. Computer Engineering are solidly grounded in the theories and principles of computing, mathematics and engineering (Srimani, Soldon, Impagliazo, Hughes & Nelson, 2002, para 2).

The quantitative approach involves the use of complex mathematics, which may not be suitable or relevant to an employer's expectation for many computer-networking students (Maj, 2000). However, its advantages include powerful mathematical tools, which may be helpful in the design and implementation of hardware and software. Unfortunately, however, the tools may not be based upon real systems (Smith, 2000). There has been research outlining the lack of poor mathematical skills by computer science students (Hennessy & Patterson, 1996; Keitel, 1992; Veal, Engel, & Maj, 2001; Veal, Maj, & Swan, 2000). The quantitative engineering approach to computer networking is not common in the teaching of computer networking courses (Williams, 2003).

The software / algorithmic computer science approach to teaching primarily deals with UNIX sockets and Windows C ++ programming. Although most computer networking courses are taught with no programming content, some authors are of the opinion that programming should be included as part of the courses (Quinton, 1993; Toll, 1995). However, most students studying computer networking courses will not be placed in situations where a socket-based programming approach will be required (Toll, 1995). The software / algorithmic approach is generally concerned with the top three layers of the seven layer Open System Interconnection (OSI) model.

A third equally valid approach in teaching computer networking is based upon internetworking devices e.g. switches and routers. The main focus of this approach is concerned with the bottom four layer of the seven layer OSI model. Traditionally, computer networks courses have not provided students with access to networking devices. The initial cost of the equipment, and the costs of maintenance and frequent upgrading, as demanded by ever changing technology, may have been some of the reasons why hands-on access to networking equipment has not been provided in the past (Dixon, McGill, & Karlsson, 1997). Computer networking courses may need to favour the hands-on, lab-based approach because of the practical nature of the subject (Heuring & Jordan, 1997; Hyde, 2000; Mueller, 2002). This approach is currently very common in teaching of computer networking topics (Yurcik, 2001). Murphy notes:

> That practical hands-on skills and knowledge is fundamental to conceptual understanding. That knowledge is more transferable to different situations when acquired by a gradual process of conceptual understanding (Murphy, 1998, p. 246).

Furthermore Mandla, the national president of the Australian Computer Society notes:

> In our experience, those who have undertaken an appropriate vendor certification to complement their tertiary degree have found a warm reception from employers and recruiters. We are also seeing a growing number of universities offering vendor certification as part of their degree courses (Mandla, 2004, p.14).

Computer networking today is complex (Jeenings, Landweber, Fuchs, Farber & Adrion, 1986; Quarterman & Hoskins, 1986). Regardless of which of the above three different approaches can be used in teaching computer and network technology, most networking courses start by giving an overview of the OSI model (Comer, 1999; Smith & Francia, 2000). Within computer networking, data communication, Information Systems (IS), and management course units based on the OSI model are very common (Gutierrez, 1998; Smith, 2001). Typically, the functionality of each layer of the OSI model is explained and then examples of this functionality in a specific network protocol are presented. Marti et al notes:

> The key to understanding packet based data communication is the abstract concept of a layered architecture. In a layered architecture, functionality is divided into sperate layers, with no sharing of functionality between layers. Each layer uses the services of the layer immediately below it. Layers that are not adjacent do not communicate (Marti, Pooch, & Hamilton, 1996, p.162).

The OSI Model (Figure 3.1) does not provide any new details with respect to protocols and functionality compared with the TCP/IP or Department of Defence Model (Figure 3.2), except that it uses more layers. The traditional seven layer model is now being addressed as a five-layer model (Cassel & Austing, 2000; Comer, 1999; Kurose & Ross, 2000). However, there are few, if any, implemented network protocols in which the actual architectural layers of a real system are strictly aligned with those of the OSI model (Cassel & Austing, 2000). However, the OSI model provides the basis of layering networks and it also helps hardware and software designers to use a common framework for design and development of equipment.

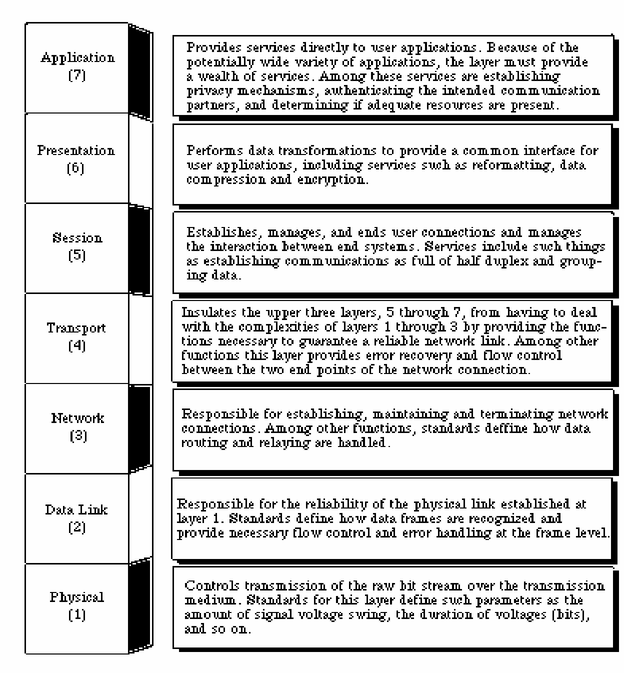


Figure 3.1: OSI Model Reference Model (Coleman, 1995)

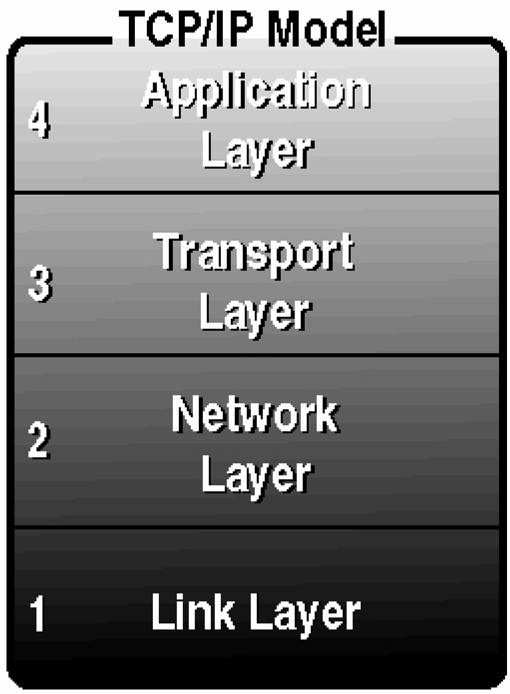


Figure 3.2: TCP Model (Baker, 2002)

3.2 Procedures and Tools used in Teaching Computer Networking

Different pedagogies have been used to aid student understanding of computer network technology. The following section provides an overview of that procedures and tools that are used in the teaching of computer networks.

3.2.1 Packet Tracing Tools

Among the many tools used is the packet trace tool. The packet trace tool employs both the top-down and bottom-up approach to teaching networking. According to Marti and Hamilton:

> First, being able to accurately predict packets shows an understanding of the interaction of the different network layers in a single network device. Second, diagnostic skills are attained when students can understand why observed packets are not the same as the predicated packets. Third, understanding network protocols enables one to debug that application that work over the network by determining exactly what information is exchanged among the application (Marti & Hamilton, 1996, p.163).

Packet tracing tools assume that students have an understanding of the networking system, which may not be the case if they are new to computer networks. Though packet tracing may provide advantages such as improving problem solving skills and troubleshooting there can be problems with the actual packet capture program.

> For traces captured under high-volume traffic conditions, sometimes the packets filters fail to capture all of the packets…there can be some loss of fidelity when data goes through the trace parser. This is in fact general problem for any network-monitoring tool (Pang & Paxson, 2003, p.3).

3.2.2 Virtualization and Simulation

Virtualization and simulation are also commonly used in the teaching of computer networking courses. Virtual Machine technology supports complex guest operating systems and lets them share common hardware installed on the system. This technique tends to favour the top-down approach in which students learn about operating system details and functionality. Its advantages include cost saving, space considerations, and security when other applications are running on the machine

(Kneale, Horta, & Box, 2004). Many courses in network administration are using this approach for teaching. Nakagawa notes:

> VMware is favourable for increasing practical ability and that it is effective for studying administrative aspects. Consequently, it provides a helpful study support and fosters the students own initiatives in regarding administrative aspects (Nakagawa, Suda, Masahiro, & Mida, 2003).

Although Virtual Machine (VMware) technology provides some advantages, there are known problems associated with its use as an aid to teaching. Virtualization may not necessarily provide students with an understanding of the basic concept of client server technologies when both client and server are operating on the same machine. Furthermore, as noted by Neih there can be implementation issues associated with its use:

> The most important feature lacking in VMWare for deploying it in a computer lab setting are easy-to-use system administration options. VMWare does not support any features to let a system administrator limit how a user configures a VM… (Nieh & Leonard, 2000).

Simulation is also widely used in the design and teaching of computer networks. Kheir notes that:

> Basically, simulation is the process by which an understanding of an already existing (or to be constructed) physical system is obtained by observing the behaviour of a model representing the system. Thus simulation is justly considered the art and science of experimenting with models (Kheir, 1996, p.3).

Furthermore, as noted by Breslau, the use of simulation is also common in computer network experimentation:

> The Internet's rapid growth has spurred development of new protocols and algorithms to meet changing operations requirement such as security, multicast transport, mobile networking, policy management, and quality-of-service support, however, building test beds and labs is expensive. Network simulators provide a rich opportunity for efficient experimentation (Breslau et al., 2000, p.59).

Simulation allows students to observe how computer networks work. Barnett (1993) proposed the use of the NetSim simulator to support both major project assignments and more focused homework assignments. Furthermore, Chang (1999) describe the Optimized Network Engineering Tools (OPNET) system which models networks and sub-networks, individual nodes and stations and state transitions, and can be used to teach and research Computer Networking (Chang, 1999; Davis, Ransbotton, & Hamilton, 1998). It should be noted that simulation and modelling are often used in an interchangeable way within the field of computer and network technology. Pooch and Wall note that:

> A model may be described as the body of information about a system gathered for the purpose of studying the system. It is not only an orderly collection of information, but is an orderly representation or structuring of the information. The characteristics should be representative of the characteristics of the real system (Pooch & Wall, 1993, pp.14-16).

Whatever the simulation tool, a prudent technique is to incorporate the tool into supervised laboratory/project assignments, individual homework assignments, and classroom demonstrations (Barnett, 1993).

According to Parker simulation is a good approach but he further notes that students often do not see the ‘big picture’ (Parker & Drexel, 1996). More importantly simulation may not provide students with a suitable pedagogical framework (Coe, Williams, & Ibbett, 1996; Searles, 1993). Although simulated environments have their benefits, providing hands-on exercises gives students a chance to apply the theory they learn from textbooks (Engel & Maj, 1999; Minch & Tabor, 2003). It has been noted that practical experience is necessary to fully understand network management problems, and that practical sessions are desirable to change the level of participation of students. This will make students more responsible for their own learning, and will serve notice of the lecturer’s increased expectation (Lankewicz, 1998; McDonald, Rickman, McDonald, Heeler, & Hawley, 2001). This is important because universities are increasingly required to respond to market forces, such as employer and employee expectations (Fetherston, 2002). One possible solution is to incorporate a vendor-based curriculum as part of university awards.

3.2.3 Hands-on (Vendor Based Approach)

The vendor-based approach to teaching networking is common in Australian Universities (Abelman, 2000; DharuKeshwari & Maj, 2003). Due to the increasing availability of vendor-based courses as components of undergraduate curricula, students can now benefit from a more practical approach and have the opportunity to program networking devices (switches, routers). Vendor specific awards such as Cisco Certified Networking Administration (CCNA) (Cisco, 2004), Cisco Certified Networking Professional (CCNP) (Cisco, 2004), are now readily available as part of the undergraduate and postgraduate courses in Australia (DharuKeshwari & Maj, 2003).

Students studying the Cisco Network Academy Program (CNAP) need to comprehend a large number of new concepts within a short time span. The hands-on approach to network technology education requires an understanding of switch and router operation. However, an extensive analysis of educational materials (Hudson et al., 2003; Lewis, 2003; Waters, Rees, & Coe, 2000) in this area has indicated that these devices are typically treated as 'black boxes'. Such an approach may not be best suited to the promotion of learning, as students are required to construct their own mental model of the internal operation of such devices which may, or may not, be correct and may not be able to encompass future ideas. There is a need to develop a good conceptual model of networking devices.

3.3 A New Model for Teaching

Networking systems are complex and require conceptual understanding. As Tranter states:

> The identification of these essential features often requires considerable engineering judgment and always requires a thorough understanding of the application for which the model is being developed (Tranter, Shanmugan, Rappaport, Kosbar, 2004, p.11).

Students studying computer and networking curricula face the difficult task of understanding complex technology within a limited time frame (Veal, 2004). Furthermore Mudge notes:

> Teaching students architectural details of particular computers or networks can lead to rapid obsolescence of much of the knowledge acquired as the rapid pace change affects factors such as cost compatibility, marketplace, and applications (Mudge, 1996, p. 674).

Traditional approaches to teaching computer and networking technology may not be relevant to modern employment requirements and students learning (Mudge, 1996; Nwana, 1997).

Even though the various pedagogies associated with hands-on workshops are provided, students may find it difficult to get a grasp of the technology they are studying based on a “black box” approach. A teaching approach with too much emphasis on low-level details may no longer be appropriate for many computer science students as Veal notes:

> There is a large conceptual gap in proceeding from the bottom-upwards with traditional syllabi due to the difficulty of reaching the point of understanding of computer and network technology, due to the complexity within modern computer systems and computer networking (Veal, 2004, p.101).

Furthermore, traditional approaches to teaching computer hardware may not be relevant to modern employment requirements (Nwana, 1997).

A bottom-up approach may no longer be appropriate for many computer science students possibly due to the following reasons. For many students, a bottom-up approach may not be appropriate in even traditional syllabi; there is a large gap in proceeding from the traditional bottom-up starting point of digital logic to reaching the point of understanding how a computer works. Modern-day systems and network-centric computing are very complex and foundation of traditional digital logic may be insufficient to permit satisfactory linking and transferring of learning.

For many students, a bottom-up approach may not best suit to introduce concepts from a constructivist perspective.

According to Constructivism, students must actively construct knowledge rather than passively absorbing it via lectures. In this theory, initial learning is based upon the students own cognitive structure. According to Ben-Ari,

> A teacher cannot ignore the student’s existing knowledge; instead he or she must attempt to question the student in order to understand what models the student posses, and only then attempt to guide the student to the ‘correct’ theory (Ben-Ari, 2001, p. 52).

A student’s knowledge construction of systems is based on conceptual models. Such models are constructed by the individual. If a standard model is not provided students will construct their own mental model that will be used to understand, interpret and manipulate the system under investigation. Without a standard model, a lack of understanding by a student may result in them generating a conceptual model that is flawed and which hinders future learning. However, model generation is iterative. Misconceptions of the system may be rectified based on experience and as more advanced topics cause the individual to continually revise and modify their model. However, such misconceptions may be persistent. Experts usually have tried and tested mental models. The need to provide students with a conceptual model are well documented (Fetherston, 1999). According to von Glasersfeld:

> Knowledge cannot simply be transferred by means of words. Verbally explaining a problem does not lead to understanding, unless the concepts the listener has associated with the linguistic components of the explanation are compatible with those the explainer has in mind. Hence it is essential that the teacher have an adequate model of the conceptual network within which the student assimilates what he or she is being told. Without such a model as a basis, teaching is likely to remain a hit-or-miss affair (von Glasersfeld, 1992, p.136).

A conceptual model must not only be technically correct but also valid for different levels of complexity. This will allow the model to support not only introductory but also more advanced concepts.

3.4 Summary

There is a wide range of approaches to teaching of computer network technology. The quantitative approach involves the use of complex mathematics, which may not be suitable or relevant to an employer’s expectation for many computer-networking students. However, its advantages include powerful mathematical tools, which may be helpful in the design and implementation of hardware and software. Simulation is also widely used in the design and teaching of computer networks. Although simulated environments have their benefits, providing hands-on exercises gives students a chance to apply the theory they learn from textbooks.

Many institutions within Australia and internationally have adopted the Cisco Networking Academy Program (CNAP). The CNAP provides on-line curriculum, preparation for internationally recognized vendor based certification Cisco Certified Networking Associate (CCNA) and Cisco Certified Networking Professional (CCNP). The CNAP program also provides low cost equipment such as switches and routers to participating institutions. However, an extensive analysis of educational material in this field suggests that such devices (switches, routers) are effectively treated as black boxes. This approach is contrary to prevailing educational theory, which supports the need for a conceptual model to aid student understanding.

4. EDUCATIONAL PERSPECTIVES

Education uses theories from many diverse fields and there are many different theories applied to education. Theories are mainly concerned with teaching and learning. Smith and Ragan define a theory as: "*an organised set of statements that allow us to explain, predict, or control events*" (Smith & Ragan, 1999, p.18). Gagne define learning as: *"A change in human disposition or capability that persists over a period of time and is not a simply ascribable to processes and growth"* (Gagne, 1985, p. 2). Mayer 1982 defines learning as: "*Learning is the relatively permanent change in a person's knowledge or behaviour due to experience*" (Mayer, 1982, p. 1040).

Learning can be thought of as a process that occurs as a result of intrinsic motivation and reflection on personal experience. There are four main principles that underpin educational theory. Behavioural psychology (Skinner, 1968) focuses only on objectively observable behaviours and discounts mental activities. A cognitive model (Merrill, Li, & Jones, 1991) suggests that knowledge is constructed and is not simply a learned response to stimuli. The third domain of constructivist approaches (Jonassen, 1991) considers the nature of knowledge, the mental activities of learners and how knowledge develops in learning. Finally humanist principles are based on human nature (Merriam & Caffarella, 1991, 1998).

Those four main views can be incorporated into a traditional model (Figure 4.1) and can be applied to teaching and learning theories. The traditional model consists of four major domains of human nature and how these domains interface with each other. The four domains are the physical, the social-emotional, the spiritual and the mental.

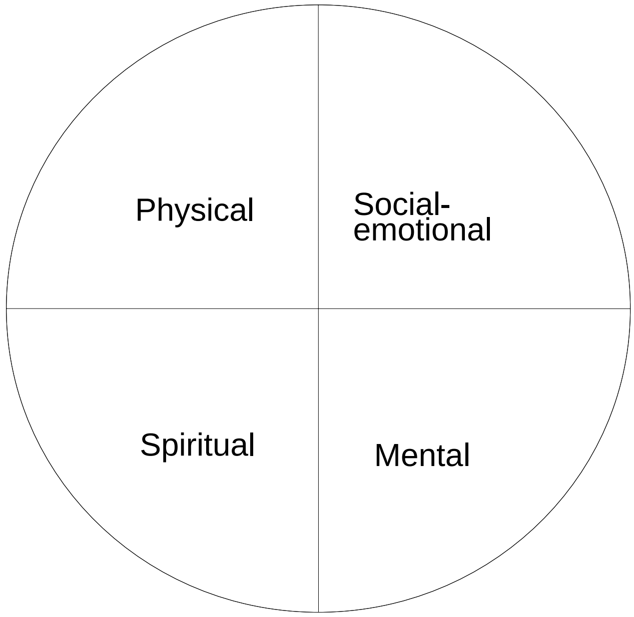


Figure 4.1 The Traditional Model

The Physical domain addresses the aspects of one's physical existence such as aging, maturation, development and the resulting behaviour.

The Social-Emotional Domain addresses the individual's interactions with the self and others. With respect to social interaction Bandura notes: Social learning takes place through observation and modelling of the behaviours of others (Bandura, 1986).

The Spiritual Domain addresses the spiritual aspects of human development. (Coles, 1992).

Finally, the Mental domain primarily addresses the mental functions, cognitive growth, and cognitive development within the individual. As an example of a theory based within the cognitive domain, (Piaget, 1970) suggests that there are four main development cognitive stages:

1. Sensorimotor;
2. Pre-operations;
3. Concrete operations; and
4. Formal operations.

Although recognising the validity of the other domains, the focus of this research is on the development of students' mental models i.e. the mental domain. But it is recognised that the other domains are valid.

4.1 Computer and Network Education

The computer and network technology age is rapidly changing, which puts considerable demand on universities and other higher degree institutions to provide skills relevant to present and future industry needs. One goal of a higher educational institute is not only to train students with relevant skills, but also to provide them with a good theoretical foundation for skills relevant to the present job market, and which will enable them to continue learning. According to Chen:

> The rapidly changing, increasingly complex business world requires college graduates to use multiple, complex skills to solve business problems. Conventional teaching strategies that mainly consist of lectures may not be effective to prepare these students for employment … construct knowledge and emphasizes presenting learning activity in a meaningful context, provides an alternative theoretical foundation for rethinking and redesigning teaching practices (Chen, 2003, p.17).

Conventional teaching methods in computer and networking may not provide students with good conceptual understanding that students can apply after finishing the course and which enable future learning.

> Many of the learning outcomes within the curriculum are non-traditional: that is, they focus on students' developing skills related to doing scientific inquiry and understandings about such inquiry rather than only on students' acquisition of "scientific facts." Such learning goals require teachers to create classrooms in which students regularly interact with one another to share and critique ideas -- and to define a set of expected behaviors ("norms") that will lead to constructive student discussions (Stewart, 2001, para. 4).

Students' understanding at present in many university courses is more related to memory than rich understanding (Bjork, 1978; Hiebert & Carpenter, 1992). There is a need for students to develop a rich mental model, which can provide a context for the technology being studied. Such models may help in developing rich understanding. According to Reeder:

> The problem is, many teachers mistake signs of apparent understanding for true understanding. For example, students using the right words and definitions, manipulating formulas correctly, or answering questions with borrowed options give the impression that they understand. And in fact they may, since someone with understanding can do those things, but it is also possible to do them without understanding. Therefore, it is simply not safe for a teacher to infer understanding from those types of responses. As it turns out, the degree to which students' grasp a concept can be reliably inferred only when they can somehow apply the concept in an authentic context (Reeder, 2002, para. 6).

The ability to recall and explain a concept does not necessarily reflect understanding, nor does it guarantee that students can apply and use the concept in a meaningful way (Julyan & Duckworth, 1996). One main goal of teaching in computer science is to enable students to create a valid mental model of new concepts, consistent with formal Computer and Networking definitions. The student's mental model should be built on a set of correct pieces of knowledge that are related to the concepts taught. Models can be employed as an aid to conceptual understanding. According to Gilbert:

> A model is a simplified representation of a system, which concentrates attention on specific aspects of the system, Moreover, models enable aspects of the system, i.e. objects, events, or ideas which are either complex, or on a different scale to that which is normally perceived, or abstract to be rendered either visible or more readily visible (Gilbert, 1995, p.891).

Furthermore Thomas notes with respect to student difficulty in understanding concepts that:

> The use of diagrammatic representation provides an alternative to just offering more words, which may only compound their difficulties (Thomas, 2000, p. 2).

Models have been used in science education and the use of models has been well documented in science education research (Gilbert & Boulter, 1998; Linn & Muilenberg, 1996). It is reasonable to expect that research from science education can be applied to the teaching of computer networking technology.

4.2 Conceptual Models

Good mental models that allow the construction of richer ideas are also important as they can assist conceptual change. According to the ACM,

> Concepts explain the how and the why of information technology and they give insight into its opportunities and limitations. Concepts are the raw material for understanding new information technology as it evolves (ACM, 2001, p. 70).

This is important because, according to research conducted in science education over at least the last 25 years it has been demonstrated across many contexts that students' existing ideas are often firmly held and unfortunately often wrong. Driver and Erikson (1984) suggested that students' existing incorrect ideas often interfere with instruction and their alternative conceptualisations can hinder further learning.

Researchers at Cornell University developed a theory of conceptual change in the early 1980s (Posner et al., 1982). Posner et al. applied ideas about scientific revolutions (Kuhn, 1970) to individual learning, and derived the following conditions for bringing about conceptual change:

1. There must be dissatisfaction with a currently held conception;
2. The alternative conception must be intelligible;
3. The alternative conception must appear plausible; and
4. The alternative conception must appear fruitful.

Posner et al. later reviewed Kuhn's ideas about conceptual change in 1992 but in both versions the importance of good mental models was apparent. The conditions above give clues as to what constitute a good (alternative to student's existing) mental model – it must be intelligible, plausible and potentially fruitful. A conceptual model describes essential features and identifies the principal processes taking place in that process (Zamg, 2004).

Many authors suggest that providing an individual with a conceptual model of a system before instruction enhances user learning (Bayman & Mayer, 1984; Carroll & Mack, 1992; Moran, 1981). Conceptual models may support superior learning over giving only procedural instruction. This has also been noted by Davis & Bostrom (1993). Furthermore Carroll and Mack notes:

> If a system has been built to conform to a consistent model or well-formed set of methods, training may simply involve presenting the user with the model or methods. Several researchers have been concerned with developing techniques for providing users with appropriate conceptual models, something that even state-of-the-art instructional materials for software fails to do (Carroll & Mack, 1992, p.44).

Conceptual models have the potential to provide students with more effective learning. These models may be analogical or abstract in form, but can assist students to superior learning.

4.3 Mental Models

Mental models provide a powerful mechanism for storing knowledge within the human mind (Norman, 1983). Such structures have a significant impact on virtually all forms of human activity (Barker & van Schaik, 1999) because of the ways in which they can influence behaviour. Mental models need to utilize abstraction. According to D’Andrade:

> Schemas are conceptual structures and processes which enable human beings to store perceptual and conceptual information about the world and make interpretations of events through abstraction (D'Andrade, 1992).

The mental image of the world around us that we carry in our heads is a model (von Glasersfeld, 1992). According to Forrester:

> One does not have a city or a government, or a country in his head. One has only selected concepts and relationships, which one uses to represent the real system (Forrester, 1971, p. 53).

He further states that:

> The mental model is fuzzy. It is incomplete. It is imprecisely stated. Furthermore, within one individual, a mental model changes with time and even during the flow of a single conversation (Forrester, 1994, p. 213).

Mental model knowledge is used to support thinking about the students' prior learning (Rouse & Morris, 1986). Prior learning consists of two types: single, isolated ideas and connected, interrelated ideas. The connected, interrelated ideas are often referred to as schema, or mental models. Some educational researchers think that novices in a field have unorganized and disjointed understandings, while experts have more or less complete and organized sets of understandings (Pitts, 1995).

Furthermore, Woolfolk notes:

> Experts in particular field have a wealth of domain-specific knowledge, that is, knowledge that applies specifically to their area or domain. They not only have declarative knowledge (facts and verbal information), but they also have at their command considerable procedural knowledge, an understanding of how to perform various cognitive activities. And they know when and why to apply their understanding; that is, they have conditional knowledge, so they can manipulate their declarative and procedural knowledge to solve problems (Woolfolk, 1998, p.274).

Most learning theories suggest that knowledge and truth exist outside the mind of the individual and are, therefore, objective (Moallem, 2001). Applying constructivist ideas to teaching computer networking would suggest starting from the concepts held by the student and then possibly utilising commonly held concepts. Knowledge and truth are thereby constructed by the students and do no exist outside their mind (Duffy & Jonassen, 1992).

4.4 Constructivism

Constructivism is a learning model that is popular in educational theory. Constructivist theory is based on active learning and is not only about the discovery of facts. This theory has been extensively tested in the field of science and mathematics education (Cobern, 1991; Confrey, 1991; Driver & Bell, 1985; Wheatley, 1991). According to Ben-Ari:

> Constructivism is a theory of learning which claims that students construct knowledge rather than merely receive and store knowledge transmitted by the teacher. Constructivism has been extremely influential in science and mathematics education, but not in computer education (Ben-Ari, 1998, p.257).

Constructivism is also a foundation of many modern teaching practices. Constructivism is about helping the learner in the construction of their own mental model. Changing a student's mental model requires conceptual reorganization, also known as conceptual change (Chi & Roscoe, 2002). The process of making an existing mental model explicit is a crucial precursor to the cognitive restructuring process (Wichmann, Gottdenker, & Jonassen, 2003). According to Phye:

> Constructivism is a movement that combines cognition from a development perspective with other important issues, such as motivation, self-directed learning, and a focus on the social context of learning (Phye, 1997, p.48).

Students studying computer networking are dealing with abstract and difficult topics that are not always clear or easy to understand. Setting the stage for conceptual change and learning by modelling is in itself a meaningful activity for gaining and refining understanding in complex scientific domains (Milrad, Spector, & Davidsen, 2002). Conceptual change through building models has been advanced as a tool to help student learning in science (Edelson, Gordin, & Pea, 1999), because educators have recognized that model-based reasoning can facilitate the development of mathematical-scientific understanding of the natural world (Lehrer, Horwath, & Schuable, 1994).

An important aspect of conceptual reorganization is abstraction. Abstraction has been a common theme in the context of aiding conceptual understanding (Ramsden, 1992; Staggers & Forrcio, 1993). It should be noted that different subjects have their own appropriate models at differing levels of abstraction dependant upon the area of use. Norman states that:

> A good representation captures the essential elements of the event, deliberately leaving out the rest. The critical trick is to get the abstraction right, to represent the important aspects and the unimportant. This allows everyone to concentrate upon the essentials without distraction from irrelevancies. … Get the relevant aspects rights, enhance people's ability to reason and think; get them wrong, and the representation is misleading, causing people to ignore critical aspects of the event or perhaps from misguided conclusions (Norman, 1993, p.92).

The theory of constructivism proposes that learners construct their own idiosyncratic understanding of concepts (Trowbridge & Wandersee, 1994). Concept maps are effective tools for making the structure of their knowledge explicit (Edmondson, 1993).

4.5 Concept Mapping

A student's mental model can be inferred from a concept map provided by them. Concept mapping helps in thinking, fitting ideas together, as well as mapping new ideas together. The idea of a concept map was proposed by Novak.

> A mind map consists of a central word or concept, around the central word you draw the 5 to 10 main ideas that relate to that word. You then take each of those child words and again draw the 5 to 10 main ideas that relate to each of those words (Novak, 1991, 1993).

Human beings understand the world by constructing models of it in their mind (Johnson-Laird, 1983). Therefore, the analysis of students' mental models provides insights into the content and structure of the knowledge individuals undertaking a course of study have constructed. Matching learning tasks within students' Zone of Proximal Development (ZPD) is a vitally important component of the teaching process (Vygotsky, 1978). von Glasersfeld and Steffe note that:

> Because there is no way of transferring meaning, i.e. concepts and conceptual structures, from one students head to another, teachers, who have the goal of changing something in student's heads must have some notion of what goes on in these heads. Hence it would seem necessary for a teacher to build up a model of the student's conceptual world (von Glasersfeld & Steffe, 1991, p.96).

In learning to construct a concept map, it is important for the students to begin with a domain of knowledge that is very familiar to them. Since concept map structures are dependent on the context in which they will be used, students studying computer and networking units should be provided with schema of networking devices which they can use to learn about how internetworking devices operate. The schemas provided may help students to create a mental schema of the technology they are learning. One can create a context that will help progressive learning. This approach is in line with the hierarchical structure of the concept map. It is always essential to start with a basic concept and then progressively build up the complexity. For example, when teaching communication between two nodes one should start with baseline of two PCs connected by a cross over cable, then introduce a layer 2 switches, followed by a layer 3 routers.

One needs to use abstract models in order to show the relation between different internetworking devices and its operation.

4.6 Summary

Of all the education theory analysed, it appears that knowledge construction is based on conceptual models. Such models are constructed by the individual. If a conceptual model is not provided students will construct their own mental model that will be used to understand, interpret and manipulate the topics that they studying. A conceptual gap arises because of some difference between the user's mental model of the application and how the application actually works. Without a proper conceptual model, students may lack proper understanding. Furthermore, this may result in a student's generating a conceptual model that is flawed. However, model generation is iterative. Misconceptions of the system can be corrected based on experience and more advanced topics, causing the individual to continually revise and modify their model. However such misconceptions may be persistent. Experts have tried and tested mental models. The need to provide students with a conceptual model is well documented. A conceptual model must not only be technically correct but also valid for different levels of complexity thereby supporting not only introductory but also more advanced concepts.

5. PUBLICATIONS CHART

The following section provides an overview of published papers, publication metrics and contribution of each paper towards the final thesis. All these publications make use of the concept of abstraction. The following five publications are based on modelling IT systems using B-Nodes and an e-commerce education case study:

1. E-commerce Curriculum – Is it Secure?
2. B-Nodes: A Proposed New Technique for Database Design and Implementation.
3. Modelling Global IT Structures using B-Nodes
4. B-Nodes: A Bridge Between the Business Model, Information Model and Infrastructure Model
5. Modelling Web Site Infrastructure using B-Nodes Theory

These publications attempt to provide an answer to the first research question: "*Can B-Nodes theory be applied to model hardware infrastructure supported by computer and network technology?*"

The order in which publication appear shows the logical progression of the research. Each publication addresses key problems in modelling, education and design of IT systems. The use of B-Nodes is suggested as a possible aid to model IT systems.

The remaining five publications addresses the second research question outlined earlier as part of this thesis *"Can state model developed from B-Node theory be applied to the teaching of internetworking and hence promote student learning?"* The publications are:

1. An Examination of Vendor-Based Curricula in Higher and Further education in Western Australia
2. Abstraction in Computer Network Education: A Model Based Approach
3. State Models for Internetworking Technologies
4. A New State Model for Internetworking Technology
5. A Pedagogical Evaluation of New State Model Diagrams for Teaching Internetworking Technologies

The use of an abstraction and modelling approach is common to the papers that are outlined in Figure 5.2., whereas research papers concerned principally with educational issues and curriculum development are outlined in Figure 5.3. The arrow upward represents the logic follow and the development of the research. The side arrow represents convergence of research from educational theory into the use of new state models used as pedagogical tool in the teaching of Internetworking Technologies.

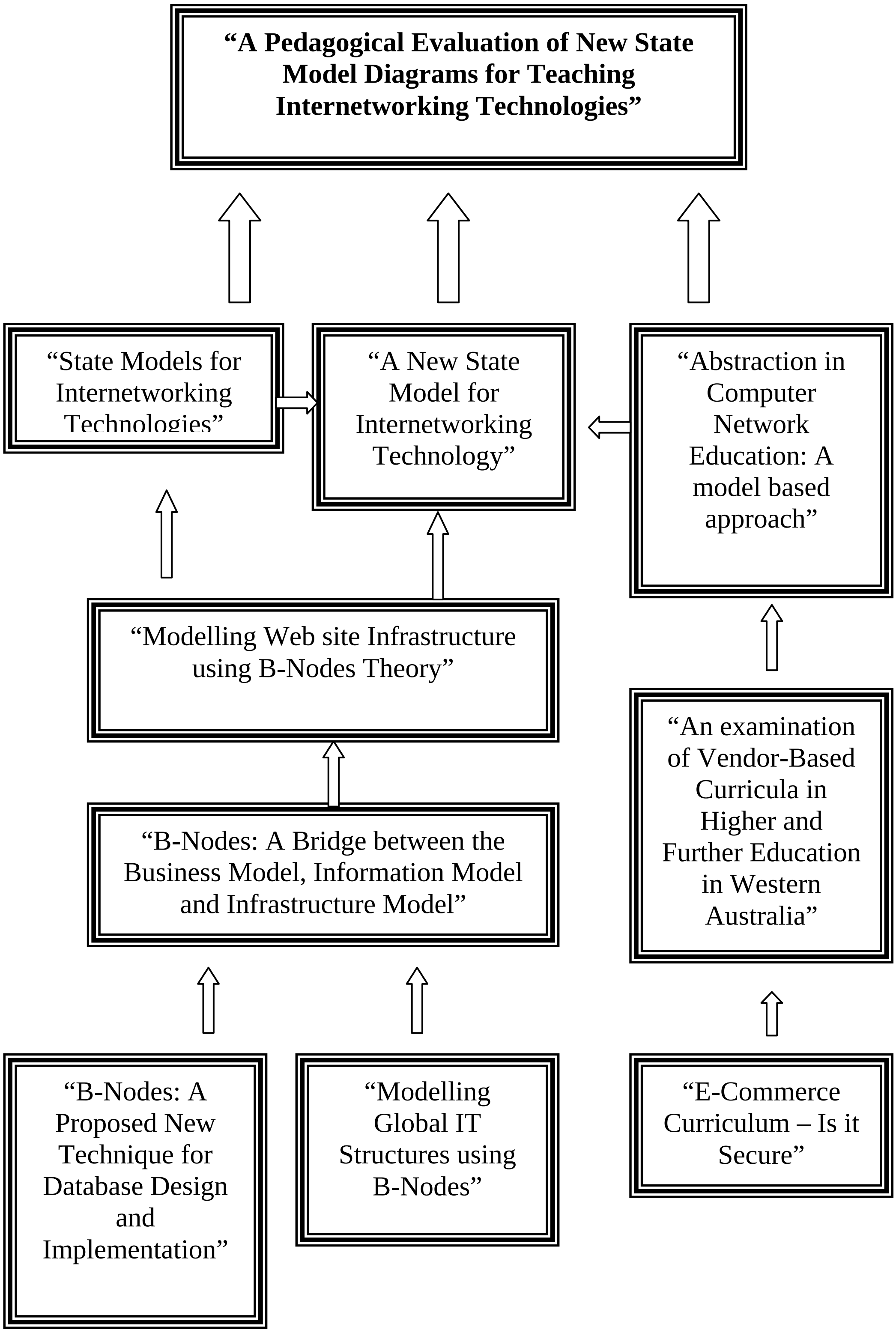


Figure 5.1 Publications Chart

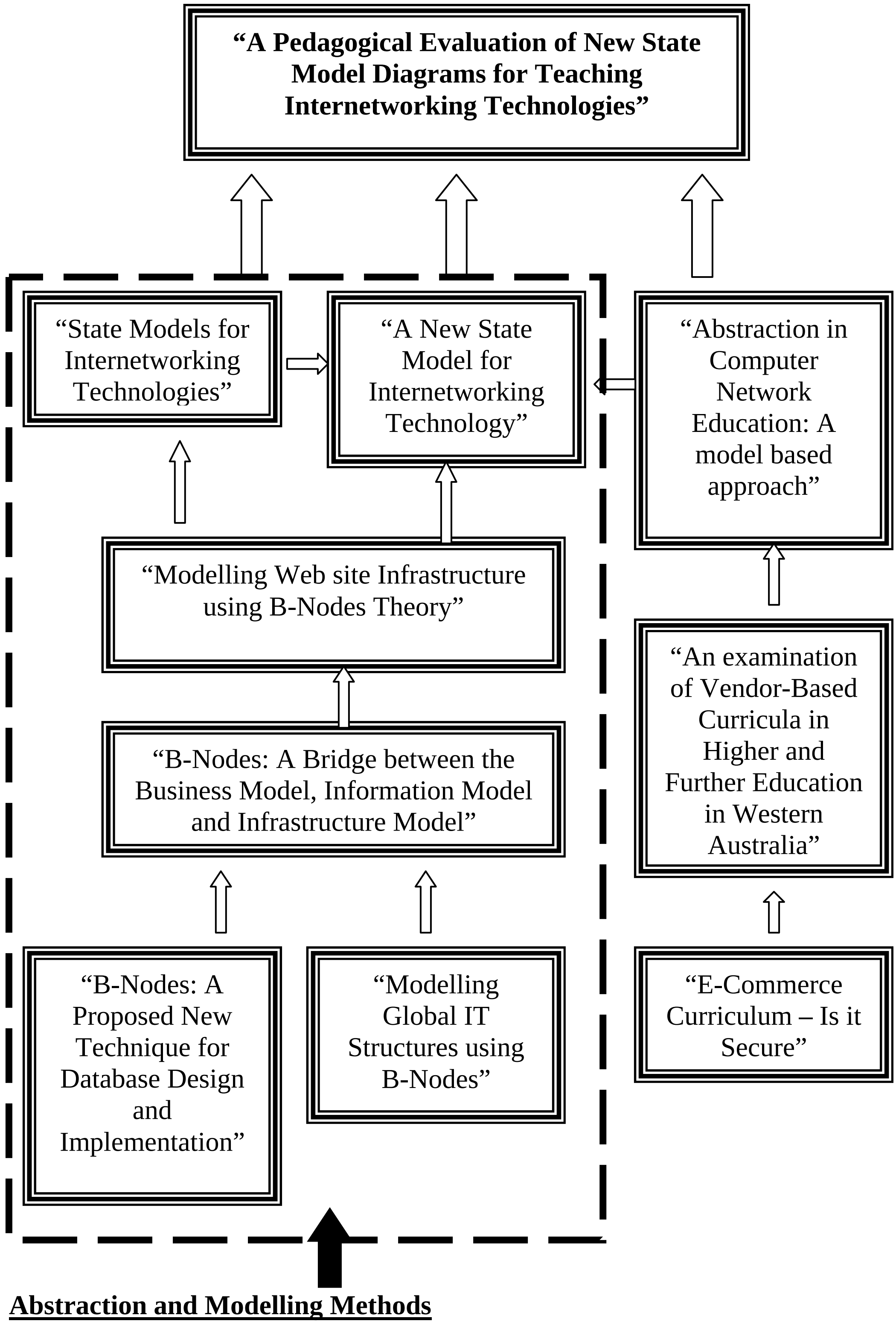


Figure 5.2 Core Theme: Abstraction and Modelling Methods

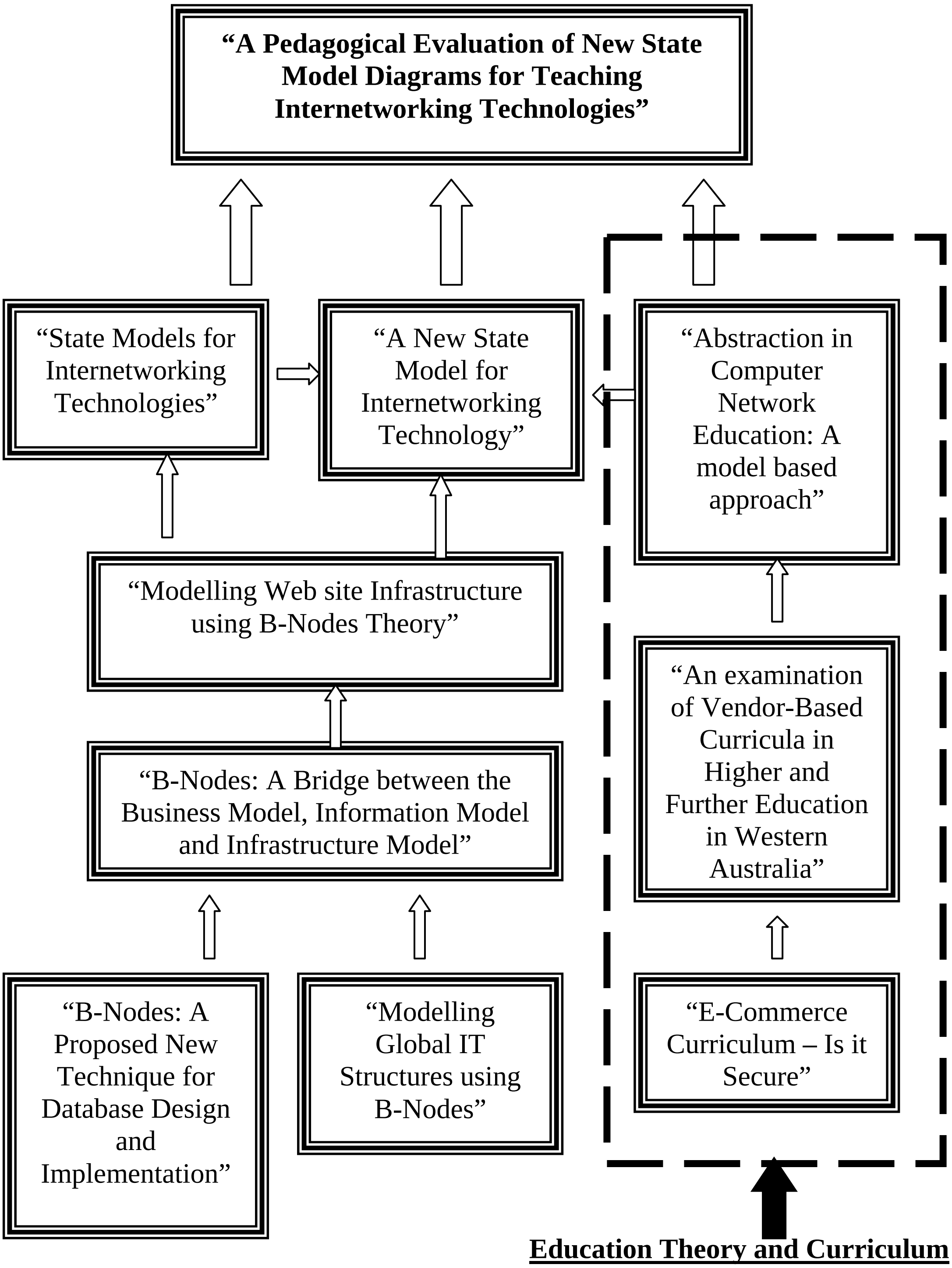


Figure 5.3 Supporting Theme: Education Theory and Curriculum

5.1 E-commerce Curriculum- Is it Secure

Title of the paper : E-commerce Curriculum- Is it Secure

Authors: J Yee, S P Maj, G Kohli

Conference: 4th Baltic Seminar on Engineering Education

Conference Location: Copenhagen, Denmark

Conference date: 2nd -7th July, 2000

This is an annual international conference. The major focus of this conference was to explore issues related to engineering and computer science education. The conference accepts papers and articles that explore new issues and directions in engineering education. The conference accepts papers on technical merit and had acceptance rate of 70 - 75 %. The papers in this conference present global research and development activities. In total, 18 lead papers, and over 70 regular papers have been contributed by authors from 29 countries worldwide. All papers are blind reviewed by two independent referees.

This paper provides an overview of current e-commerce curricula in Australia. The paper is concerned with how universities are responding to challenges due to the exponential growth of e-commerce. Analysis of e-commerce curricula within Australia resulted in the following findings:

1. Twelve universities offer undergraduate curricula in e-commerce
2. Three of these universities do not specifically address e-commerce security
3. Eight universities address security, but do not offer specific units in security
4. One university offered both e-commerce curricula and included specialised units in e-commerce security.

5. None of the universities provided a dedicated laboratory for e-commerce curriculum.

A case is presented for universities to provide curriculum that directly enhances the employment prospects of students. The paper concludes by recommending e-commerce curricula to provide students with practical units with the essential dedicated laboratory, and that this should be a prerequisite for any non-issues based e-commerce security curriculum. Furthermore, Competency Based Assessment should be used to measure skills effectively and efficiently, relevant to employer expectations.

E-COMMERCE CURRICULUM – IS IT SECURE

J Yee, S P Maj, G Kohli
Dept of Computer Science, Edith Cowan University
Perth, Western Australia

ABSTRACT

Today, with the increase in popularity of electronic commerce (e-commerce), security on the Internet for e-commerce transactions is an important issue for both the vendors and customers. In an international context employers are seeking IT professionals with both Internet and e-commerce security skills. In Australia many universities now offer undergraduate e-commerce curriculum. Typically, computer security is addressed however only by means of a general introduction. Previous investigations both within Australia and internationally clearly indicated that both students and employers demand not only a good theoretical foundation but also practical skills. These practical skills can only be gained by extensive workshop exercises in a laboratory equipped with contemporary commercial hardware and software. Such a laboratory has been designed and commissioned at Edith Cowan University. This paper outlines an undergraduate curriculum in e-commerce security. This curriculum is designed to provide not only conceptual knowledge but also essential procedural skills directly relevant to employer expectations.

INTRODUCTION

The exponential growth of the Internet is changing the way organizations and people conduct business. The term Electronic Commerce (e-commerce) is the term used to define this new method of conducting business. Definitions vary but e-commerce is generally considered to be sharing business information, maintaining business relationships and conducting business transactions by means of telecommunications networks. As such e-commerce is taken to include electronic messaging, electronic trading, electronic data interchange (EDI), electronic payroll etc. All these are now an essential part of trade and industry (Phillip, 1997). A review of e-commerce in Australia by the Anderson Consulting Group indicated that about 80% of Australian business leaders are of the opinion that e-commerce will revolutionize the way business will be conducted in the next five years (Consulting, 1999). However, according to (Bhimani, 1996),

> *More sophisticated attacks can take advantage of basic flaws in the Internet's infrastructure. For example, the TCP/IP protocol suite used by all computers connected to the Internet is fundamentally lacking in security services.*

Furthermore, he makes the point that, by almost all accounts, security is the leading barrier to widespread commerce on the Internet. A direct consequence of these market demands is the number and type of jobs progressively required by industry and commerce in this field. This paper is concerned with how universities are responding to these challenges. According to the 1991 ACM/IEEE-CS report, the outcome expected for students should drive the curriculum planning.

Furthermore, Campus Leaders (Review, 1996) notes that, the predominant reason why they (students) have gone to university was to get skills, knowledge and a qualification that would assist them in either gaining employment or enhancing their prospects for promotion or a more rewarding job. (Review, 1996)

It can be argued that universities should provide curriculum to directly enhance the employment prospects of students. However to do this universities must respond to the market place.

INTERNET SECURITY

The original Internet was designed as a research tool, not as a commercial environment. However, according to Bhimani:

A robust security solution for transaction processing satisfies the following fundamental security requirements:

- *Confidentiality*
- *Authentication*
- *Data Integrity*
- *Non-repudiation*
- *Selective application of services*

To meet these criteria a number of protocols have been defined that include network-layer, session-layer and application-layer security. However, one payment system is emerging as market leader – Secure Electronic Transactions (SET) supported by Visa and MasterCard. Furthermore, it should be noted that this is a very dynamic environment and subject to many rapid and unanticipated technological developments. Additionally, there are considerable, on-going developments in 'micro-payments' (Rivest et.al., 1996). A micro-payment is defined as a transaction in which the value of the data being purchased is less than the cost of the transaction. Again this field is subject to development. This is the environment that universities must prepare students for.

INDUSTRY ANALYSIS

A preliminary analysis of '*The Australian*', an Australian national newspaper indicates that there are now jobs that simply did not exist two years ago. Job titles include: Internet Service Provider (ISP) Security Specialist, Internet Security Specialist, E-Business Security etc. An on-going analysis indicates that the number of this type of position is also increasing. Many of the advertisements analyzed request that prospective candidates should posses skills which include setting up and maintaining firewalls, Point to Point Protocol (PPP), File Transfer Protocol (FTP), serial line internet protocol (SLIP), routing, security technologies, network operating systems (predominantly Windows NT and Unix). These positions require not only knowledge of the field but also require practical skills directly relevant to this industry. We are currently conducting a detailed international job market analysis to complement the Australia survey. The question now is how universities are responding to this dynamic job market.

E-COMMERCE CURRICULUM IN AUSTRALIA

Within Australia there are forty universities, however given the size of Australia some universities have distinct and semi-autonomous campuses e.g. University of Western Sydney – Nepean, Hawkesbury and McAuthur. All the universities were analyzed to determine what e-commerce curriculum was provided. The analysis was conducted by means of visiting the web pages of each university. A more detailed investigation is currently being undertaken.

Twelve universities offer undergraduate curriculum in e-commerce. Three of these universities do not specifically address e-commerce security. Eight of these universities address e-commerce security as part of the e-commerce curriculum but do not offer specific units in security. Only one university, Curtin University of Technology, offers both e-commerce units, which include one specialized unit in e-commerce security.

Nine universities offer postgraduate curriculum in e-commerce from which seven specifically address e-commerce security issues. Two universities offer postgraduate awards in e-commerce. They also include specialized units in e-commerce security.

This analysis indicates that universities are responding to the demands for curriculum in e-commerce. However, only one university, Curtin University of Technology, provides both undergraduate and post-graduate students with specialized units in e-commerce curriculum. We could not find any university that provided a dedicated laboratory for e-commerce curriculum. Without such a dedicated laboratory it is simply not possible to provide many of the essential practical skills and knowledge needed in this field. At Edith Cowan University two new units, Computer Installation & Maintenance (CIM) and Network Installation & Maintenance (NIM) were introduced. Accordingly, a new, dedicated laboratory was designed and constructed at an initial cost of AUS$35,000. The extensive use of decommissioned equipment significantly helped to reduce the cost. With this budget it was possible to provide each of the 10 workplaces within the laboratory with three different types of PC of increasing cost and complexity. For the NIM unit each workplace consists of a client (Intel 486), a server (Intel 486) and a ten-port hub i.e. a self-contained network. The cabling in the laboratory (Category 5, Unshielded Twisted Pair) allows all the hubs to be connected together to a switch or hub, located in a patch panel, and hence connected to the worldwide web with protection by means of a firewall. However, the problems associated with managing this type of laboratory are non-trivial. According to Kohli (Kohli, Maj, & Veal, 2000)

At the conclusion of each two-hour workshop all the equipment must be checked and where necessary reconfigured. Experience dictates that two hours of support is needed between successive CIM workshops. For the NIM unit only one hour of technical support is needed between successive workshops. For all other units at ECU only the first hour of each workshop is supervised, for the second hour students are guaranteed access to the laboratory – but no supervision. During the semester they have unrestricted access to any vacant laboratories 24hours a day, 7 days a week. Given the nature of the CIM, NIM units workshop access is restricted to the allocated workshop periods, all of which are fully supervised. Prior to the start of *each workshop each pair of students must complete and sign a checklist of equipment. Students are not permitted to leave the workshop until the equipment is returned.*

The two units CIM and NIM are complemented by two further units – Computer Systems Management (CSM) and also Network Design and Management (NDM). The two units CSM and NDM have a dedicated laboratory established at the initial cost of AUS$60,000. This laboratory has a variety of equipment that includes: Bay Stack switches, Pentium III PCs, NT Server, a direct Internet connection. According to Kohli (Kohli et al., 2000) these units provide students with the opportunity to practically implement network security issues such as: user security, web security etc.

All these units provide a practical, inter-disciplinary, problem oriented approach to computer technology. The objectives of the units are that completion of these units students will be able to, *'Install, maintain and manage a population of networked PCs to a professional standard using safe, systematic, generic and vendor independent skills'*.

Curriculum without such a dedicated laboratory must, by definition, tend to be 'issues' based. Valuable though this approach may be it fails to provide the appropriate procedural knowledge required by industry. According to Havard (Havard, Hughes, & Clarke, 1998):

The world of work requires people who 'know how and not just know what'. Graduates invariably fall into an 'information gap' of having knowledge about a specific subject area but do not necessarily know how to operate in the working environment.

It should be noted that the authors are currently conducting an in-depth analysis of the job requirements in both commerce and industry.

PROCEDURAL AND CONCEPTUAL KNOWLEDGE

Technical expertise is an enhancement of theoretical knowledge. According to Cervero (Cervero, 1992),

the popular wisdom among practicing professionals is that the knowledge they acquire from practice is far more useful than what they acquire from more formal types of education.

Cervero further argues that the '*goal of professional practice is wise action'* and that '*knowledge acquired from practice is necessary to achieve this goal'*. Both declarative and procedural knowledge are necessary for professional practice. Declarative knowledge is knowledge *that* something is the case; procedural knowledge is knowledge *how* to do something. Cervero clearly makes the point that,

A major difference between experts and non-experts in any field is that experts have far more procedural knowledge. That is, they know how to perform their craft.

According to Goldsworthy (Goldsworthy, 1993), skill

refers to a person's ability to do something well. It relates to expertness, a practiced ability, a dexterity in performing a task. It is an outcome that flows from knowledge, practice, inherent abilities and an understanding of the task to be performed.

In order to provide procedural knowledge students must be given the opportunity to develop these skills in the context of 'real world' problems. This approach was successfully taken by Maj (Maj, Robbins, Shaw, & Duley, 1996) who attempted to teach computer technology '*by use of a practical, inter-disciplinary, problem oriented approach*.' Rather then lowering academic standards Professor Lowe (Armitage, 1995) argues, the complexity of the real world is more intellectually taxing than living in imaginary worlds of friction-less planes, perfectly free markets or rational policy analysis.

The practical approach adopted by Maj has met with considerable success. According to Maj:

The student demand has consistently exceeded possible place. The initial quota of 100 students for the unit CIM was exceeded with 118 students enrolling and even then demand exceeding possible places. The student attrition rate was 8.5% with a subsequent unit failure rate of less than 10%. An independent unit review of the unit found : 80% would recommend this unit; 75% found the practical sessions useful; 70% found the unit relevant to their needs and 55% think this should be a compulsory unit (Maj, Fetherston, Charlesworth, & Robbins, 1998).

COMPETENCY

Havard, Hughes and Clarke have noted the general lack of information given to students regarding employer expectations (Havard et al., 1998),

On completion of their academic programme the graduate was given no way of knowing how their skills compared to the requirements of industry.

Different forms of assessments have both advantages and disadvantages. Written examinations can enable students to demonstrate a greater depth of knowledge while at the same time they may also test a limited breadth of their understanding. Whilst written assignments can give students an opportunity to search literature and show understanding. Multi-choice questions provide the means to test students with minimum effort across a large portion of the syllabus, but may not allow students to show their in depth knowledge. It is possible however, for a student to successfully pass units without ever attending any workshop. However both declarative and procedural knowledge must be assessed. Given the need for a practical approach to curriculum Competency Based Assessment (CBA) offers a possible method of testing specific practical skills. Competency in an Australian context is discussed by Karmel (Karmel, 1995) who states that:

The competency movement that has developed in Australian education over the last five or six years has been driven largely by a desire to fashion and improve education/training outcomes in relation to the world of work from entry to professional level. The origins of competency based training, standards and assessment for specific skilled vocations are also to be found in moves to reform industrial awards.

The relevance of CBA in the university sector is questioned by Hamly cited by Wilson (Wilson, 1992),

Frank Hamly, the executive director of the Australian Vice-Chancellor's Committee, said in a graduation address, that he believed that most of the higher order intellectual skills which universities impart are not capable of measurement as competencies and that universities must therefore resist their implementation.

The competency debate still continues in Australia. However, in a very practical sense questions regarding the suitability of such assessment techniques have been addressed by Veal (Veal, Maj, Fetherston, & Kohli, 1999),

*The students were assessed during a standard two-hour workshop. The work was conducted in a normal manner with the assessor independently evaluating students. At no point was the workshop interrupted, or extra time taken. Ten students were evaluated. Despite the fact that considerable emphasis is placed on ensuring that students are provided with the highest possible standard of good workshop practices to underpin Health & Safety the results clearly indicated that some students did not demonstrate a satisfactory competency in basic safety practices. It would not have been possible to determine this result without a CB*A.

It can be concluded that CBAs may be used to test practical skills not tested via other forms of assessment. Furthermore, that no single assessment technique is sufficient if both procedural and conceptual knowledge are to be assessed.

CURRICULUM DESIGN

The major pedagogical problem is to provide students with sufficient skills and knowledge in e-commerce security. A purely 'issues' based approach to this problem is perhaps best suited to management students. This type of student is primarily concerned with resource management rather than the practical implementation of security technologies. The job market analysis clearly indicates the need for students with both a good theoretical understanding of e-commerce security underpinned by associated practical skills. Accordingly any e-commerce security curriculum must encompass a wide range of inter-related subjects. By example, client security includes problems such as ActiveX containers and scripting, JAVA security, desktop integration problems etc. Web server security covers database technology, server-side scripting, CGI scripts, security of third vender software, etc. Data transaction security includes secure channels like using Secure Socket Layer (SSL), web spoofing, store-account payment system (eg, cybercash), stored-value payment system (eg. storing value on smart cards) etc. Operating system security includes the choice of operating system, and understanding operating system holes, firewall security, etc. In order to both meaningfully understand these subjects and also implement them students must have the appropriate relevant foundations.

We recommend that the units CIM & NIM, or similar practical units with the essential dedicated laboratory, should be pre-requisites for any non-issues based e-commerce security curriculum. It is only with this relevant background knowledge that students would be to meet the needs of industry.

CONCLUSIONS

An initial survey has clearly demonstrated that new categories of job vacancies in e-commerce now exist. Furthermore it is anticipated that the number of such positions is likely to significantly increase. In particular there appears to be a growing demand for positions in e-commerce security. A preliminary analysis of e-commerce curriculum within Australia indicates that universities are responding by providing e-commerce curriculum. However there appears to be a very strong prevalence to an 'issues' curriculum. Important though this is a good case can be made for providing students with the opportunity to develop not only declarative but also procedural knowledge. Such procedural knowledge can only be acquired by means of workshop exercises in dedicated laboratories. It is important to be able to measure such skills, not only as useful feedback to students but also as an additional means of assessment. Competency Based Assessments provide a structured environment with which to provide this type of assessment. Competency is an on-going debate within Australia however it has been demonstrated that simple but meaningful CBA's can be conducted in a non-intrusive manner during a normal workshop.

REFERENCES

1. Phillip, M., *Successful e-Commerce*. 1997, Melbourne, Australia: Bookman.
2. Consulting, A., *A Review of eCommerce in Australia*, . 1999, Anderson Consulting.
3. Bhimani, A., *Securing the Commercial Internet.* Communications of the ACM, 1996. **39**(6).
4. Workforce, E.t., *Educating the workforce for the new millenium*, in *Campus Review*. 1996.
5. Rivest, R. and P. Shamir. *PayWord and MicroMint: Two simple micropayment schemes*. in *Proceedings of the 1996 RSA Data Security Conference*. 1996. San Francisco.
6. Veal, D. and S.P. Maj. *Computer and Network Technology Education at Maximum Value and Minimum Cost*. in *ASEE 2000*. 2000. St Louis, Missouri.
7. Kohli, G., S.P. Maj, and D. Veal. *MultiMedia Technology - an Opportunity for Modern Engineering Education*. in *American Society for Engineering Education 2000*. 2000. Missouri, USA.
8. Havard, M., M. Hughes, and J. Clarke, *The Introduction and Evaluation of Key Skills in Undergraduate Courses.* Journal of Further and Higher Education, 1998. **1**: p. 61-68.

9. Cervero, R.M., *Professional practice, learning and continuing education: an integrated perspective.* International Journal of Lifelong Education, 1992. **11**(2): p. 91-101.
10. Goldsworthy, A.W., *IT and the Competency Debate - Skill vs Knowledge A Major Issue.* The Australian Computer Journal, 1993. **25**(3): p. 113-122.
11. Maj, S.P., *et al.*, *Computer and Network Installation, Maintenance and Management - A Proposed New Curriculum for Undergraduates and Postgraduates.* The Australian Computer Journal, 1996. **30**(3): p. 111-119.
12. Armitage, C., *'Irrelevant Degree Factories' must change with times*, in *The Australian*. 1995. p. 23.
13. Maj, S.P., *et al. Computer & Network Infrastructure Design, Installation, Maintenance and Management - a proposed new competency based curriculum*. in *The Proceedings of the Third Australasian Conference on Computer Science Education*. 1998. The University of Queensland, Brisbane, Australia.
14. Karmel, P., *Education and the Economic Paradigm in Confusion Worse Confounded - Australian Education in the 1990s.* Academy of Social Sciences in Australia - Occasional Paper Series, 1995. **1**.
15. Wilson, B.G. *Higher Education*. in *Higher Education and the Competency Movement*. 1992. ACT, Australia.
16. Veal, D., *et al. Competency Based Assesment Techniques for use in a Computer Installation & Maintenance Unit*. in *3rd Baltic Region Seminar on Engineering Education*. 1999. Goteborg, Sweden.

5.2 B-Nodes: A Proposed New Technique for Database Design and Implementation.

Title of the paper :	B-Nodes: A proposed New Technique for Database Design and Implementation
Authors	S P Maj, G Kohli
Conference:	The Twelfth Australasian Conference on Information Systems.
Conference Location:	Coffs Harbour, NSW, Australia
Conference date:	4^{th}-7^{th} December, 2001

This is an annual conference held in the Asia-Pacific region. The main aim of this conference is to publish work on Information System Design and Models, Implementation, and Information System Education. The overall acceptance rate of this conference is around 60 - 65 %.

This paper provides an overview of different modelling techniques used in the analysis and design of IT systems. Of all the techniques analysed none provided a technique or any tools that could be used for hardware selection for running database applications. Database applications are critical for the success of e-commerce applications.

Furthermore, the Oracle case study gave little guidance in providing techniques or tools that can be used for hardware selection. However, Oracle provided different techniques for tuning database applications.

B-Nodes were proposed as a modelling method. This method utilizes a common fundamental unit for measuring the bandwidth of each module, Furthermore; B-Nodes provide recursive top-down decomposition, which allows all underlying detailed to be controlled. Finally, a case was made from the results to date to use B-Nodes as a tool for use in systems analysis and design methods.

B-Nodes: A Proposed New Technique for Database Design and Implementation.

S P Maj, G Kohli
Department of Computer Science,
Edith Cowan University,
Western Australia
p.maj@cowan.edu.au

Abstract

There exist a wide range of methods that can be used for the analysis and design of IT systems. However, a survey of a wide range of methods and a detailed analysis of one structured method indicated the lack of a simple method for modelling hardware. The ORACLE database provides detailed guidelines regarding the minimum platform to run the database and how to derive table spaces (system, user, applications, rollback etc) size of shared pool buffer, Redo log buffer pool etc. that can be used to define hard disc capacity. The system can then be optimised. However little guidance is given regarding the performance of other devices (microprocessor, RAM, bus structures etc). This paper evaluates the new B-Node modelling technique as a possible standard technique in structured systems analysis and design for evaluating hardware performance.

Keywords: Systems Analysis and Design, modelling, B-Nodes

Introduction

There are a wider range of systems analysis methods that employ a variety of different modelling tools and techniques. A method consists of phases or stages that in themselves may consist of sub-phases. Systems analysis and design methods include: ad hoc (Jones 1990), waterfall (Royce 1970), participative (Mumford and Wier 1979), soft systems (Checkland 1981), prototyping (Nauumann and Jenkins 1982), incremental (Gibb 1988), spiral (Boehm 1984), reuse (Matsumoto and Ohno 1989), formal (Andrews and Ince 1991), rapid application development (Martin 1991), object oriented (Coad and Yourdon 1991) and software capability (Humphrey 1990). Regardless of the underlying theme of each information system all methods must provide techniques for modelling data, processes and system functions. However, there appears to be no simple technique that will model the digital infrastructure (hardware and software) to determine if it will perform to an acceptable standard required by the analysis and design specifications. The Structure Systems Analysis and Design Method (SSADM) was evaluated in-depth as a method for developing an information system. SSADM is mandatory for UK central government software development projects. This method is sponsored by the Central Computer and Telecommunications Agency (CCTA) and the National Computing Center (NCC) thereby further ensuring its importance within the software industry within the UK. SSADM provides tools that allow the estimation of storage requirements. From the Composite Logical Data Design and Logical Design Volumes, detailed information about the data volumes may be extracted. It is possible to obtain detailed information about: data space for each data group, volumes of each data group, volumes of relationships, variance of volumes over time etc. However, according to Ashworth,

> '*The prediction of overall system performance is a difficult area and there are several simulation programs and experts system programs available which attempt to predict and improve system performance either generally or for specific hardware and software*.' (Ashworth and Goodland 1990)

Other than this there are no simple tools or techniques that can be used for the selection of hardware. Furthermore, SSADM employ a range of different, heterogeneous performance metrics that include: MPS, CPU time, disk access time, number of instructions per database call etc. Such benchmark metrics are in themselves problematic as it is not possible to directly relate the technical specification to metrics typically used in the Service Level agreement e.g. Transactions/s, Throughput etc.

Database design using ORACLE

Oracle provides technical specifications for the minimum platform to run the database (e.g. Uni-processor Pentium 300MHz, 128Mbytes RAM and 1.2 Gbytes Hard Disc space for the database installation). Other architectural solutions include: Symmetric Multiprocessor Systems (SMP), Clusters, Massively Parallel Processing Systems (MPP) and Non-uniform Memory Access Computers (NUMA) Table 1 (Greenwald 1999).However, beyond this, little guidance is provided for the selection of hardware.

Ranking	Scalability	Manageability	Availability	Price
Best	MPP	Uni-processor	Cluster	Uni-processor
	Cluster	SMP	MPP	SMP
	NUMA	NUMA	NUMA	Cluster
	SMP	MPP	SMP	NUMA
Worst	Uni-processor	Cluster	Uniprocessor	MPP

Table 1: Processor Architectures

Standard Entity Relationship Diagrams (ERD's) are used to determine the number of tables needed from which it is possible to calculate the upper and lower table space limits for the user, applications, roll backs etc. Hence the size of the hard disc can be calculated. The main emphasis here is database sizing for current and future needs. It is then possible to tune the Oracle database using performance views such as V$SYSTEM_EVENT and V$SESSION_WAIT. Solutions to I/O bottlenecks include: disk striping technologies to spread I/O across spindles (e.g. Redundant Array of Independent Disks (RAID) technology), use of table-space to segregate and target different types of I/O, and distribute system overheads evenly across spindles.

Following this it is possible to optimize the database using either a cost-based or rule-based criteria. Cost based optimizer depends upon Logical reads, CPU, and

Network calls. The cost base optimiser has access to statistics relating to the tables and indexes, such as the size of the table, the min and max value in indexed columns. The cost based optimiser uses the INIT.ORA parameter DB_FILE_MULTI_BLOCK_READ to estimate the number of I/O's required to perform a full table scan.

Bandwidth Nodes

A PC is a collection of heterogeneous technologies (microprocessor, electronic memory, hard disc drive etc). Comparison of the relative performance of these different technologies is difficult due to the use of heterogeneous units that include: MHz, nanoseconds, seek times in milliseconds etc. However a PC and its associated sub-modules (microprocessor, electronic memory, hard disc drive etc) may be modeled using B-Nodes (Maj, Veal et al. 2000). Each B-Node can be treated as a data source/sink capable of, to various degrees, data storage, processing and transmission. The performance of each B-Node may be calculated, to a first approximation, by Bandwidth = Clock Speed x Data Path Width (B = C x D) with units in either MBYTES/S. The use of a common fundamental unit (MBYTES/S) allows the relative performance of these heterogeneous technologies to be directly compared. The use of B-Nodes using the performance metrics of MBYTES/S and Images/s has been confirmed experimentally (Maj, Veal et al. 2000). Furthermore, the use of a simple, fundamental unit allows other derived metrics to be used. By example, the units Images/s may be more meaningful to a typical user because it relates directly to their perception of performance. An image is defined as 1024x1024 pixels with a colour depth of 3 bytes per pixel i.e. 3MBytes. To a first approximation, smooth animation requires approximately 30 Images/s (90MBYTES/S). The performance of each B-Node may be calculated using this metric. We have therefore a common unit of measurement, relevant to common human perception, with decimal based units, that can be applied to different nodes and identify performance bottlenecks. Similarly other metrics may be derived.

Device	Clock Speed (MHz)	Data Width (Bytes)	Bandwidth (MBYTES/S) B = C x D	Bandwidth (Images/s)
Processor	400	8	3200	1066
DRAM	16 (60ns)	8	128	42
Hard Disc	60rps	90Kb	5.4	1.8
CROM	(30 speed)	(150Kbytes/s)	4.6	1.5
ISA Bus	8	2	16	5.3
Ethernet	100	1/8	12.5	4.1

Table2: Bandwidth

The B-Node model has been successfully applied to a wide range of PC architectures allowing a direct comparison not only between different B-Nodes within a given PC but also comparisons between different PC's. Using B-Nodes it was possible to analyze PC's with different Intel microprocessors (8088/6, 286, 386, 486 etc.) and various associated bus structures (Micro Channel Architecture, Extended Industry Standard Architecture, Video Electronic Standards (VESA) Local Bus).

B-Nodes typically operate sub-optimally due to their operational limitations and also the interaction between other slower nodes. For example, a microprocessor may need two or more clock cycles to execute an instruction. Similarly a data bus may need multiple clock cycles to transfer a single data word. The simple bandwidth equation can be modified to take this into account i.e. Bandwidth = Clock x Data Path Width x Efficiency (B = C x D x E). The early Intel 8088/86 required a memory cycle time of 4 clocks cycles (Efficiency = ¼) however, for the Intel 80x86 series, including the Pentium, the memory cycle time consists of only 2 clocks (Efficiency = ½) for external DRAM. Efficiencies can be calculated for each device and the performance calculated accordingly.

B-Nodes and E-Business Architecture

Using a structured systems analysis and design method, such as SSADM, it is possible to design a database for an e-business system. For such a system the Customer Model can be used to describe the user behaviour patterns in which the number of clients, type of resources requested, pattern of usage etc is all used to determine the workload characteristics. Workload characteristics, in conjunction with the resource infrastructure model will determine site performance and whether or not the Service Level Agreements can be met. Customer Behavior Modelling methods have been successfully used to determine aggregate metrics for E-Commerce web sites (Menasce, Almeida et al. 1999). Using these various models it is possible to obtain a wide variety of different performance metrics that include: Hits/s, Page Views/Day, Unique Visitors etc. Furthermore, the SSADM method employs a range of different, heterogeneous performance metrics that include: MPS, CPU time, disk access time, number of instructions per database call etc. Given this wide range of different units performance evaluation is problematic. By example how can the units of Hit/s be used to select the bus structure of a PC? However, if a web server is modeled as a B-Node then the performance metric is bandwidth with units of MBYTES/s. The sub-modules of a server (microprocessor, hard disc, electronic memory etc) and also be modeled as B-Nodes, again using the same performance metric. The use of fundamental units (MBYTES/s) allow other units to be derived and used e.g. transactions per second (tps). Assuming the messages in a client/server interaction are 10kbytes each, the performance of each B-Node can be evaluated using the units of transactions/s (Table 3)

Device	Clock Speed (MHz)	Data Width (Bytes)	Efficiency	Bandwidth (MBYTES/S) B = C x D x E	Bandwidth (Tps)	Load (Tps)	Utilization
Processor	400	8	0.5	1600	160k	250	<1%
DRAM	16 (60ns)	8	0.5	64	6.4k	250	4%
Hard Disc	60rps	90Kb	0.5	2.7	270	250	93%
CROM	(30 speed)	(150kBytes/s)	0.5	2.3	230	250	>100%
ISA Bus	8	2	0.25	4	400	250	63%
Ethernet	100	1/8	0.9	11.25	1.1k	250	23%

Table 3: Bandwidth (Transactions/s)

If the demand on this server is 250 Transactions/s it is a simple matter to determine both performance bottlenecks and also the expected performance of the equipment upgrades. From table 2 it is possible to determine that for this web server, the hard disc drive, CDROM and ISA bus are inadequate. The metric of transactions/s can easily be converted to the fundamental unit of MBYTES/S, which can then be used to determine the required performance specification of alternative bus structures, CDROM devices and hard discs. A PCI (32 bit) bus structure is capable of 44MBYTES/S. A 40-speed CDROM device has a bandwidth of approximately 6MBYTES/S. Similarly replacing the single hard disc drive by one with a higher performance specification (rpm and higher track capacity) results is a new server capable of meeting the required workload.

Conclusions

A survey of a wide range of methods and a detailed analysis of one structured method (SSADM) indicated the lack of a simple method for modelling hardware. Similarly some software vendors provide only limited guidance for the selection of hardware. A possible modelling method is B-Nodes. B-Nodes are easy to use, scalable and hence can be used for PC modules (microprocessor, hard disc etc). The use of recursive decomposition allows detail to be controlled. An e-commerce server infrastructure may be modeled as a B-Node or collection of B-Nodes (microprocessor, hard disc etc). B-Nodes use a fundamental performance metric (MBYTES/S) from which other, more meaningful metrics may be derived. As B-Nodes use abstraction they are independent of underlying technologies and are applicable not only for old and current technologies but may well be of value for some time to come.

References

Andrews, D. and D. Ince (1991). Practical Formal Methods with VDM. New York, McGraw Hill.

Ashworth, C. and M. Goodland (1990). SSADM: A Practical Approach. London, McGraw-Hill.

Boehm, B. W. (1984). “A software development environment for improving productivity.” Computer **17**(6): 30-42.

Checkland, P. B. (1981). Systems Thinking, Systems Practice. Chichester, John Wiley.

Coad, P. and E. Yourdon (1991). Object-oriented Analysis. Englewood Cliffs, NJ, Prentice-Hall.

Gibb, T. (1988). Principles of Software Engineering Management. Reading, MA, Wesley.

Humphrey, W. S. (1990). Managing the Software process. Reading, MA, Addison-Wesley.

Jones, G. W. (1990). Software Engineering. New York, Wiley.

Maj, S. P., D. Veal, et al. (2000). Is Computer Technology Taught Upside Down? 5th Annual SIGCSE/SIGCUE Conference on Innovation and Technology in Computer Science Education, Helskinki, Finland, ACM.

Martin, J. (1991). Rapid Application Development. New York, Macmillan.

Matsumoto, Y. E. and Y. E. Ohno (1989). Japanese Perspectives in Software Engineering Practice. Reading, MA, Addison-Wesley.

Menasce, D. A., V. A. F. Almeida, et al. (1999). A Methodology for Workload Characterization for E-Commerce Servers. 1999 ACM Conference in Electronic Commerce, Denver, CO, ACM.

Mumford, E. and M. Wier (1979). Computer Systems in Work Design - the ETHICS Method. London, Associated Business Press.

Nauumann, J. D. and A. M. Jenkins (1982). Prototyping: the new paradigm for systems development. MIS Quarterly.

Royce, W. W. (1970). Managing the development of large software systems: concepts and techniques. WESCON.

R Greenwald, R. S., J Stern (1999). Oracle Essentials, O'Reilly.

5.3 Modelling Global IT Structures using B-Nodes

Title of the paper : Modelling Global IT structures using B-Nodes

Authors S P Maj, G Kohli

Conference: 3rd Annual Global Information Technology Management (GITM) World Conference

Conference Location: New York, USA

Conference date: 20th- 25th June , 2002

This conference is held annually in the USA. The conference was jointly sponsored by Global Information Technology Management Association (GITMA), IFIP 8.7: Informatics in Multinational Enterprises, IFIP 9.4: Social Implications of Computers in Developing Countries. The conference provides an opportunity to Information Technology practitioners and researchers from many countries to get together for the purpose of exchanging and enriching their ideas and experience. The conference publishes papers on a range of issues facing Information Technology Management. In addition, the conference publishes work on how information technology can be applied to the current business environment. The acceptance rate of this conference was 65 - 70%. The conference mainly publishes papers on management issues. It was noted by one of the reviewer that the work presented in this paper provides “a useful technique to management when dealing with Information Technology hardware selection”.

This paper provides an overview of current vendor-based practices used in industry for the selection of hardware and software required to support IT infrastructure. Previous research by the authors had resulted in the following conclusions:

1. Oracle, provides the technical specification for the minimum platform to run a database.
2. Oracle provides techniques for estimating the optimal size of database for current and future needs;
3. Oracle provides techniques that help in optimizing and tuning e-commerce applications, but fails to provide any estimation of the required web site capacity.

In order to investigate these issues further, a questionnaire was designed to obtain a measure of the current practices used by Small to Medium Enterprises (SMEs) in Information Technology hardware selection. This questionnaire was sent to SMEs within Western Australia. The following conclusions were drawn from the data analysis conducted.

1. No company interviewed used a standard metric when planning the IT infrastructure
2. Most server hardware requirements were planned in conjunction with a vendor-based solution and companies budgeted to purchase the fastest server available in the market.
3. Previous experience of staff was the major factor in planning for IT infrastructure.
4. Finally, it was found that none of the companies interviewed had a standard method for undertaking capacity planning.

The B-Node method was proposed as an aid to measure the required capacity of the web site with reference to hardware provisioning. From the previous work, it was found that B-Nodes could be used to calculate the utilisation of various components such as processors, memory and buses. This information can be used for capacity planning, which is identified as the process of predicting future workloads and determining the most cost-effective way of postponing system saturation. A single web server in most cases is not able to handle e-business transactions. Typical e-business architecture consists of a front-end web server, a secure server, payment server, and a database server. Using B-Nodes it is possible to evaluate the load on individual servers based upon the relative traffic frequency.

From this information, it is then possible to identify system bottlenecks so that hardware upgrades can decided based on hard information presented by B-nodes.

B-Nodes provide a simple technique to model the hardware running on a web site. Based upon the work to date, it can be concluded that B-Nodes can be used to model IT infrastructure for running e-business applications.

MODELLING GLOBAL IT STRUCTURES USING B-NODES

S P Maj, G Kohli,
School of Computer & information science
Edith Cowan University
{s.maj, g.kohli} @ecu.edu.au
Tel: 61-8-93706084

Abstract

There exists a wide range of system analysis methods. The majority of the structured methods have well-established tools and techniques for modelling processes (e.g. Data Flow Diagrams) and data (Entity Modelling). However, none of the structured methods analyzed had any simple technique for deriving the hardware specifications in order to meet the service level agreements. A preliminary analysis of several companies in Western Australia indicated that infrastructure requirements were typically based upon purchasing the highest performance equipment within budget constraints and on past experience. Alternatively companies outsourced the problem to vendors. Through this paper we will address how B-Nodes may be used to control the complexity of I.T infrastructure especially e-commerce.

Introduction

There is a wide range of system analysis methods. A method consists of phases or stages that in themselves may consist of sub-phases. Systems analysis and design methods include: waterfall [1], prototyping [2], incremental[3], spiral[4]. Regardless of the underlying theme of each information system all methods must provide techniques for modelling data, processes and system functions. The majority of the structured methods have well-established tools and techniques for modelling processes (e.g. Data Flow Diagrams) and data (Entity Modelling). However none of the structured methods analyzed had any simple technique for deriving the hardware specifications in order to meet service level agreements. The Structure Systems Analysis and Design Method (SSADM) [5]was evaluated in-depth as a method for developing an information system. SSADM is mandatory for UK central government software development projects. SSADM provides tools that allow the estimation of storage requirements. From the Composite Logical Data Design and Logical Design Volumes, detailed information about the data volumes may be extracted. It is possible to obtain detailed information about: data space for each data group, volumes of each data group, volumes of relationships, variance of volumes over time etc. Furthermore, SSADM employs a range of different, heterogeneous performance metrics that includes: MPS, CPU time, disk access time, number of instructions per database call etc. Other than this there are no simple tools or techniques that can be used for hardware selection. This led the authors to investigate what information software vendors provide in regards to capacity planning.

Database design

An analysis of vendor-based solutions was conducted. It was found that Oracle, for example, provides technical specifications for the minimum platform to run the database (e.g. Uni-processor Pentium 300MHz, 128Mbytes RAM and 2 Gbytes Hard Disk space for the database installation). Other architectural solutions include: Symmetric Multiprocessor Systems (SMP), Clusters, Massively Parallel Processing

Systems (MPP) and Non-uniform Memory Access Computers (NUMA). However, investigations to date found that, little guidance provided for the selection of hardware. Standard Entity Relationship Diagrams (ERD's) are used to determine the number of tables needed from which it is possible to calculate the upper and lower table space limits for the user, applications, roll backs etc. Hence the size of the hard disk can be calculated. The main emphasis here is database sizing for current and future needs. Oracle provides proven techniques that help you in optimizing and tuning your application but the question still remains; How do companies estimate the capacity of their sites? In order to answer this question an initial investigation was carried out in Western Australia.

Case Studies

Interviews were undertaken with IT managers and server operators of 3 local companies and 1 university within Western Australia. Questions asked included: Organization structure? Number of people working in the organization? Number of end users? How many servers do they have in their organizations? How often they upgrade their hardware and software? What standard methods do they follow when buying hardware for new server? Hits/day on their web site? Frequency of infrastructure planning? etc. To preserve anonymity these three companies investigated are referred to here as 'A', 'B' and 'C'.

Company A had a portal site, which dealt with virtual eco tourism. The virtual community consisted of managers, tour planners and tourist operators. The eco tourism portal site enabled the creation of community owned projects concerning planning and sustainable development. The eco tourism system provided customized views of information sources (projects) owned or subscribed to by registered users. Uploading of projects was via HTML forms. The site consisted of a three-tier model, which included a front-end web server, application server and database server. The maximum hits per day was 1000 and the average session time per was 30-45 minutes during a given session. Companies B and C were small size online businesses, both had the same three-tier architecture as company A but they dealt with online ticketing of local events. Lastly the university IT infrastructure was carefully analyzed, and it was found that 5 people were acting in the role of system managers with a single IT manager supporting 42 servers during the semester. Recently one of the departments in the University made its course available online. This further increased the administrative load due to the large traffic generated on the web server. Most of the requests were made from within the University's LAN. The above interviews resulted in the following findings:

1. All Companies interviewed do not use a standard metric when planning the IT infrastructure.
2. The ratio of server administrator to server was 1:8 in case of the University; this resulted in a high level of administrative overhead per administrator.
3. Most of the servers were planned in conjunction with a vendor based solution and companies budgets to purchase the fastest server available in the market.
4. Past experience was the major factor in planning for IT infrastructure.
5. All the companies interviewed had no standard method for undertaking capacity planning.

Bandwidth Nodes

Computer and network systems consist of a wide range of heterogeneous technologies (Hard disk drives, electronic memory, etc) and associated heterogeneous units (MHz, milliseconds etc). Comparison of the relative performance of each of these different technologies is therefore difficult. The principles of measurement science include: fundamental unit/s (Mass, Length and Time), derived units (e.g. velocity), decimal scaling system and the use of meaningful units relevant to human dimensions and perceptions (e.g. Km/hr). In this context a PC may be described as a Multiple Instruction, Multiple Data (MIMD) architecture of B-Nodes [6]. B-Nodes represent a 'stateless', higher level of abstraction than the Turing machine. In contrast to finite automata, which have a finite number of states that may be described using state transition graphs B-Nodes are a quantifiable data source/sink capable of, to various degrees, data storage, processing and transmission. This approach allows the performance of B-Node to be assessed by a simple, common measurement – bandwidth. Where Bandwidth = Clock Speed x Data Path Width (B = C x D) with the common units of Mbytes/s. This unit is defined as the fundamental unit. In effect the use of a common fundamental unit allows heterogeneous technologies (microprocessor, hard disk drive etc), with heterogeneous units (MHz, rpm etc) to be directly compared using B-Nodes (Table1). Though useful it may not be the best initial, user oriented unit. For general acceptance a unit of measurement must be easy to understand and should therefore be based on user perception of performance. The performance of a PC and associated nodes can still be evaluated using the measurement of bandwidth but with the units of standard Image/s (Table1). The fundamental units of Mbytes/s can easily be converted to other derived units more relevant to different users and applications: text, facsimile, audio data, video and multi-media applications. In medical applications radiology images are often stored digitally. A single ultrasound image represents approximately 0.26Mbytes of data [7], which at 30 Images per second is 7.8Mbytes/s. B-Nodes may be classified as volatile or non-volatile data stores also using fundamental or derived units. B-Nodes typically operate sub-optimally due to their operational limitations and also the interaction between other slower nodes. For example, a microprocessor may need two or more clock cycles to execute an instruction. Similarly a data bus may need multiple clock cycles to transfer a single data word. The simple bandwidth equation can be modified to take this into account i.e. Bandwidth = Clock x Data Path Width x Efficiency (B = C x D x E) (Table1). The use of B-Nodes has been confirmed experimentally; however, other factors such as compression, operating system overheads etc are currently being examined.

Device	Clock Speed (MHz)	Data Width (Bytes)	E	Bandwidth (MBytes/s) = CxDxE	Bandwidth (Image/s)
Processor	400	8	0.5	1600	533
DRAM	16 (60ns)	8	0.5	64	21
Hard Disk	60rps	90Kb	0.5	2.7	0.9
ISA Bus	8	2	0.2	4	1.3
Ethernet	100	1/8	0.9	11.25	3.8

Table1: Bandwidth Nodes

Even though technical detail is lost, this model is conceptually simple and controls detail by abstraction. In effect the bit stream is converted to an information stream. As a scalable method B-Nodes can also be used to model micro-architectures[8]. A hard disk drive can be considered as two B-Nodes, one representing the electro-mechanical components and the other the control circuits. An RAM circuit can also be modeled using B-Nodes. According to Golshani, ‘*Therefore it is timely to view information as an end product and look at its creation/generation process from an engineering standpoint.’[9]*

E-Commerce Modelling

It is possible to use B-Nodes as a tool in systems analysis and design to model large (global) e-commerce architectures. Service Level Agreements in conjunction with cost constraints and the technologies used are the primary determinants in performance and capacity planning. In this context a variety of models are used. The Business model defines the purpose of the organization; the Functional Model defines the navigational structures and the Customer Model is used to describe the user behaviour patterns. Using the customer model, the number of clients, type of resources requested, pattern of usage etc are all used to determine the workload characteristics. Workload characteristics, in conjunction with the resource infrastructure model will determine site performance and whether or not the Service Level Agreements can be met. Customer Behaviour Modelling methods have been successfully used to determine aggregate metrics for E-Commerce web sites (Menasce & Almendia, 1999). Using these various models it is possible to obtain a wide variety of different performance metrics that include: Hits/s, Page Views/Day, Unique Visitors etc. However there are problems in using these metrics to define the characteristics of the required infrastructure. There appears to be no simple method to convert these various metrics to units that can be directly used to evaluate hardware performance. If a web server is modeled as a B-Node then the performance metric is bandwidth with units of Mbytes/s(S.P. Maj, D. Veal, & A. Boyanich, 2001). The sub-modules of a server (microprocessor, hard disk, electronic memory etc) and also be modeled as B-Nodes, again using the same performance metric. The use of fundamental units (Mbytes/s) allow other units to be derived and used e.g. transactions per second (T/s). Assuming the messages in a client/server interaction are 10kbytes each, the performance of each Information-Node can be evaluated using the units of transactions/s (T/s) (Table 2).

Device	Bandwidth (MBytes/s) = C*D*E	Bandwidth (T/s)	Load (T/s)	U
Processor	1600	160k	250	<1%
DRAM	64	6.4k	250	4%
Hard Disk	2.7	270	250	93%
ISA Bus	4	400	250	63%
Ethernet	11.25	1.1k	250	23%

Table2: Utilization

If the demand on this server is 250 Transactions/s it is a simple matter to determine both performance bottlenecks and also the expected performance of the equipment upgrades. From table2 it is possible to determine that the hard disk drive, CDROM and ISA bus are inadequate – the utilization (U) is too high. The metric of transactions/s can easily be converted to the fundamental unit of Mbytes/s, which can then be used to determine the required performance specification of alternative bus structures, CDROM devices and hard disks. By example, replace the single hard disk drive by one with a higher performance specification (rpm and higher track capacity). Furthermore, from the derived unit of Transactions/s other units such as Response time may also be derived. Response time is simply the reciprocal of Transactions/s.

Saturation – Transient and Sustained

Capacity planning is the process of predicting future workloads and determining the most cost-effective way of postponing system saturation (transient and sustained). Assuming that the web traffic is anticipated to rise to 550 transactions/s – the current single server solution will be inadequate. If this traffic is transient in nature a simple buffer can be used. The size of the buffer can be simply determined by the difference in input/output transactions for the saturation period. If saturation is sustained a new architecture is needed e.g. a front-end Web server, a Secure Web server, a Payments server and a Database server. Assuming each server is a separate device connected by a 100Mbps Ethernet link it is possible to model this configuration using B-Nodes. Each server represents a B-Node. The communication link may be represented as a directed arc (arrow) annotated by its bandwidth performance (MBytes/s or Transactions/s). Using Customer Behaviour Model Graphs (CBMG's) it is possible to evaluate the relative frequency that each dedicated server is used. Assuming probability based on relative traffic frequency (F) for each server the results can be tabulated (table 3).

Servers	Network Bandwidth (Mbytes/s)	F	Actual Server load (Mbytes/s)	Actual server load (T/s)
Web	11.25	0.5	5.625	0.55k
Secure	11.25	0.5	0.5625	0.055k
Payment	11.25	0.5	0.5625	0.055k
Database	11.25	0.2	2.25	0.22k

Table 3: E-commerce Servers

The load data obtained from table 3 can then be used to evaluate the performance of the individual components within each server. In the case of the Web server the actual load is 5.625Mbyte/s (0.55kTransactions/s) from which the utilization of each module can be evaluated. The CMBG's clearly indicate that the majority of the traffic is to the Web server. Assuming that it is necessary to plan for an expected load of 1,000 transactions/s. From table 3 it is evident that the web server would not perform satisfactorily. Traffic characterization may be used. Assuming that such an analysis indicates that 60% of the traffic is for static JPEG images and 40% for dynamic HTML pages. A possible solution may be to use a caching proxy to serve the static web pages. Assuming that the traffic analysis further finds that this brings down the average message size to the server from 10Kbytes to 5kBytes – the web server can easily be modeled using B-Nodes. For the web server 40% of 1,000 transactions/s is 400 transactions/s, and each transaction is on average 5kBytes. This results in a different, and much lower, utilization for the Web server.

Conclusions

An initial analysis of the companies in Western Australia indicated that their infrastructure requirement procedures were typically based on purchasing the highest performance equipment within the company's budget constraints and past experience. Alternatively companies outsourced the problem to vendors. This paper proposes B-Nodes, whose performance is rated by bandwidth (MBytes/s), as a method for modelling computer and network systems. B-Nodes represent a new, higher level of abstraction that allows technical detail to be controlled using top-down recursive decomposition. As this model is based upon high level abstraction it is independent of underlying architectural detail and can therefore accommodate rapid changes in technology. It is valid for all generations of digital PC technology to date and could therefore continue to be useful for some years to come. The use of fundamental units (Mbytes/s) allows other, more useful, units to be derived (Transactions/s) and also allows the performance of heterogeneous devices to be directly compared and evaluated. The authors propose B-Nodes as a possible aid for modelling IT infrastructure. However this needs to be extensively tested.

References:

1. W.W.Royce, *Managing the development of large software system: Concepts and Techniques*. 1970: Wescon.
2. J.D.Naumann, A.M.J., *Prototyping: the new paradigm for system development*. 1982: MIS Quarterly.
3. T.Gibb, *Principles of Software Engineering Management*. 1988, MA: Wesley, Reading.
4. B.W.Boehm, *A software development environment for improving productivity*. 1984. 30-42.
5. C.Ashworth, M.G., *SSADM: A practical Approach*. 1990, London: McGraw-Hill.
6. S.P. Maj, D.V., *Is Computer Technology Taught Upside Down?* 5th Annual SIGCSE/SIGCUE conference on Innovation and Technology in computer Science Education Helskinki, Finland, ACM, 2000.
7. S.Dwyer, *Teleradiology Using Switched Dialup Networks.* IEEE Journal on Selected Areas in Communication, 1992.
8. S.P. Maj, D.V., A.Boyanich. *A New Abstraction Model for Engineering Students*. in *4th UICEE Annual Conference on Engineering Education*. 2001. Bangkok, Thailand: UNESCO International Centre for Engineering Education (UICEE).
9. F.Golshani, S.P., O.Friesen, Y.C.Park, J.J.Song. *A comprehensive Curriculum for IT Education and Workforce Development: An Engineering Approach*. in *32nd Technical Symposium on Computer Science Education, ACM*. 2001. Charlotte, North Carolina.
10. D.A Menasce, V.A.F.A., *A metholody for workload characterization for e-commerce server.* ACM conference in Electronic Commerce, Denver, CO, ACM, 1999.
11. S.P. Maj, D.V., A.Boyanich. *A New Method for controlling Information Technology Hardware Complexity*. in *American Conference on Information Systems*. 2001. Boston, USA.

5.4 B-Nodes: A Bridge between the Business Model, Information Model and Infrastructure Model.

Title of the paper :	B-Nodes: A Bridge Between The Business Model, Information Model and Infrastructure Model
Authors	S P Maj , G Kohli
Conference:	7th International Workshop on evaluation of Modelling methods in systems analysis and design
Conference Location:	Toronto, Canada
Conference date:	25th- 31st, 2002

This workshop provides a forum for researchers and practitioners interested in modelling methods in systems analysis and design to meet and exchange research ideas and results. The Conference on Advanced Information Systems Engineering (CaiSE) and the International Federation jointly sponsored the workshop for Information Processing Working Group 8.1 (IFIP8.1).

The workshop allows researchers to share information on new modelling methods used in system analysis and design. Out of the 45 papers submitted, only 20 (44.5 %) papers were included in the workshop program after review. Furthermore, the reviewers of the workshop were activity involved in the development of systems analysis and design models used within industry for designing information systems.

Building an effective information system demands good modelling techniques. This paper examines different modelling methods used in design of Information Systems.

From the previous work (Maj and Kohli 2001; Maj and Kohli 2002) it has been found that commercial practice is based upon past experience and purchasing the highest performance equipment within budget constraints.

Various systems analysis and design methods were analysed, but none provided any simple techniques for modelling the IT infrastructure that could relate the overall performance of the system to the performance of the individual components within that system.

The performance analysis techniques used are either benchmark or analytical modelling. However, a benchmark suite can be “a one-size fits all solution”, which is not necessarily the case. Furthermore, benchmarks age with time due to changes in the underlying technology and need either to be replaced or radically updated. Analytical models are required for large systems and organisations but may not be appropriate for SMEs.

Menasce et al developed the Customer Behaviour Modelling Graphs (CMBG). This model was also analysed, but it failed to provide any technique for hardware selection.

Detailed experiments were carried out based upon the CMBG to provide a meaningful analysis of the relative load on the actual device handling the e-commerce infrastructure. As part of the experiment a client – server network was established using a Windows 2000 server with Internet Information Server 5 and a Windows 2000 Professional client. Using Microsoft’s network monitoring tool data transfer traffic for variety of commands was captured. More details about the experiments are published in the paper.

The paper proposed B-Nodes as a possible aid for modelling performance, for hardware selection and for providing a simple and potentially useful approach to model IT infrastructure. Furthermore, the B-node model can be used in conjunction with CMBG to provide more meaningful metrics for site design and hardware selection.

B-Nodes: A Bridge between the Business Model, Information Model and Infrastructure Model.

S. P. Maj, G.Kohli
School of Computer and Information Science
Edith Cowan University
Perth, Western Australia
G.Kohli, S.Maj @ {ecu.edu.au}

Abstract

Building an information system demands good modelling techniques. In structured systems analysis and design there are standard techniques for modelling data and processes. Typically data is modeled by Entity Relation Diagrams and processes are modeled by Data Flow Diagrams. A preliminary analysis of several companies in Western Australia indicated that infrastructure requirements were typically based on past experience and purchasing the highest performance equipment within budget constraints. Alternatively companies outsourced the problem to vendors. These approaches are arguably entirely unsatisfactory as they relegate IT systems analysis to conventional 'wisdom' and mythology. None of the companies analyzed employed any techniques for modelling infrastructure performance. A wide variety of structured systems analysis and design methods were therefore analyzed. None provided any simple modelling technique that could be used for the IT infrastructure. Performance analysis techniques (e.g. operational and stochastic analysis) exist and are essential for large, expensive installations. However their considerable mathematical rigor does not lend them to general use and for small to medium enterprises may not represent simple cost effective solutions. Furthermore hardware is chosen using technical specifications. Whilst performance analysis techniques employ abstraction (i.e. queues and service centers) there appear to be no guidelines for relating these abstractions to technical specifications. The use of benchmarks remains controversial. Techniques to capture customer navigational patterns using Customer Model Behavior Graphs (CMBG's) and hence obtain quantitative data about workloads have been successfully used and related to the new Transaction Processing Council Web (TPC-W) benchmark. However they employ a wide range of metrics that cannot simply be related to technical specifications, which are the basis of equipment selection. To address these problems a new modelling technique has been developed, Bandwidth Nodes. This paper is an evaluation of this technique.

Introduction

An Information Technology (IT) system is defined as all the necessary hardware and software to meet a particular business requirement. An IT system consists of different devices or sub systems, for example; microprocessor, hard disc drive, servers, and Local Area Network (LAN). Therefore, it should be noted that an IT system is a complex heterogeneous system consisting of many different technologies each with their own performance metrics, for example, revolutions per minute (rpm), megahertz (MHz), and nanoseconds (ns). The entire system's performance depends upon the relative performance of each sub-system or device in use. However, regardless of the size or complexity of the IT system, it must respond to end user demands. Using an e-Commerce web site as an example, an end user (client or customer) may visit (logon to) the web-site, that may be remotely located (on a server) and from which the end user may demand a range of different services. Such services may include browsing, searching, the purchase of goods, and then logging off. In some cases, a company's entire business (e-Businesses) may depend upon the behavior of their web site. A poorly performing web site may well result in loss of business. Surveys clearly demonstrate that web sites are sensitive to slow response times (GVU). Poor performance may well result in a number of problems, including lost revenue, inferior service delivery, customer attrition, and poor company perception. Clearly it is crucial that IT systems must be designed, built and maintained to meet an agreed Quality Of Service (QoS) which defines the levels of security, availability, and performance. A Service Level Agreement (SLA) typically specifies the performance of an IT system under normal operating conditions. SLA performance metrics include transactions per second, page views per day, hits per second. The number of clients attempting to logon may abruptly increase for a short time which may lead to temporary saturation of the IT system, giving rise to poor performance. Alternatively there may be a gradual but continuous increase in the number of users over time, which will lead to the sustained saturation of the IT system. Capacity planning is the process of both predicting when the increasing number of users will cause saturation, and determining the most cost-effective way of preventing saturation. In order to design an IT system and maintain it according to a specific SLA, it is essential to relate the overall performance of the IT system (in transactions per second or hits per second) to the performance of heterogeneous sub-systems (for example, hard disc drives, network interface cards, LAN), each with their own performance metrics (for example, rpm, MHz, nanoseconds). The problem associated with infrastructure design is non-trivial According to Fenik, '*Being able to manage hit storms on commerce sites requires more then just buying more plumbing*' (Fenik 1998).

Commercial Practices

A preliminary analysis of several companies in Western Australia indicated that infrastructure requirements were typically based on past experience and purchasing the highest performance equipment within budget constraints (Maj 2002). Alternatively companies outsourced the problem to vendors. These approaches are arguably entirely unsatisfactory as they relegate IT systems analysis to conventional 'wisdom' and mythology. None of the companies analyzed employed any techniques for modelling infrastructure performance. This scope of this survey is currently being extended both within Australia and internationally. Based on the results from this survey systems analysis and design techniques were investigated.

Systems Analysis and Design

A wide range of structured systems analysis and design methods were analysed for a simple modelling technique that could be used for evaluating and predicting the performance of equipment. Methods analysed included: ad hoc (Jones 1990), waterfall (Royce 1970), participative (Mumford and Wier 1979), soft systems (Checkland 1981), prototyping (Nauumann and Jenkins 1982), incremental (Gibb 1988), spiral (Boehm 1984), reuse (Matsumoto and Ohno 1989), rapid application development (Martin 1991), object oriented (Coad and Yourdon 1991) and software capability (Humphrey 1990). All the structured methods analysed typically provide standard techniques for modelling data (Entity Relation Diagram for example) and processes (Data Flow Diagrams for example). DFD's are not only simple, but also graphical; hence serve not only as documentation but also as a communication tool. It is recognized that communication with end users is especially important as this helps to validate for correctness.

> *The major problems associated with systems analysis arise in the understanding of the user's environment and the subsequent specification of the user's requirements. Both of these problem areas arise from problems in communication between the user and the systems analyst.* (Cutts 1988)

However none of the methods analyzed provided any comparable simple techniques for modelling the IT infrastructure that can relate the overall performance of the IT system to the performance of the individual components within that system (Maj, Veal et al. 2001). According to Shklar, '*Sites have been concentrating on the right content. Now, more of them specially e-commerce sites realize that performance is crucial in attracting and retaining online customers*.'(Shklar 1998) Apart from vendor out sourcing and prior experience, which is unsatisfactory, there are three basic techniques: Technical specifications; Performance analysis techniques and the Customer and Resource models.

Technical specifications

Regardless of the size of an IT system all sub-systems must work together, for example: a microprocessor, hard disc drive, motherboard, and LAN. It is possible to obtain detailed technical specification for all of these devices. However, the different devices employ different technologies (for example, electronic and electromechanical), so a wide range of performance metrics exists. In effect, we have heterogeneous technologies with heterogeneous metrics. Consequently, it is very difficult to evaluate the relative performance of all these devices as, unlike in science, there are no common fundamental units from which other more useful units may be derived. According to Lilja:

> *Most fields of science and engineering have well-defined tools and techniques for measuring and comparing phenomena of interest and precisely communicating results. In the field of computer science and engineering, however there is surprisingly little agreement on how to measure something as fundamental as the performance of a computer system. For example, the speed of an automobile can be readily measured in some standard units, such as meters travelled per second. The use of these standard units then allows the direct comparison of the speed of the automobile with that of an airplane, for instance. Comparing the performance of different computer systems has proven to be not so straightforward however. The problems begin with a lack of agreement in the field on even the seemingly simplest of ideas, such as the most appropriate metric to use to measure performance.* (Lilja 2000)

There does not appear to be a simple modelling technique that can be used to define a common fundamental unit for heterogeneous technologies, thereby allowing the relative performance of these devices to be evaluated, and other more useful units to be derived. This can be contrasted to science and engineering, which are founded on the principles of measurement science, hence employ internationally defined units and standards for measurement i.e. three completely independent quantities (mass, length and time) from which other units may be derived, for example, velocity and acceleration.

Performance analysis techniques

The two main techniques for performance analysis are benchmark measurement and analytical modelling.

Benchmark Measurement

Benchmark programs are used for evaluating systems. Different end user applications have very different execution characteristics; hence there exist a wide range of benchmark programs. The four main categories are: science and engineering (examples include; Whetstones, Dhrystones, Livermore loops, NAS kernels, LINPACK, PERFECT club, SPEC cpu), Transaction Processing (for example; TPC-A, TCP-B, TPC-C), server and networks (examples include; SPS/LASSIS, SPEC web) and general use (examples include; AIM Suite III, SYSmark, Ziff-Davis PC Benchmark). However, many of those currently in use (including though not limited to clock rate, Millions of Instructions executed Per Second (MIPS), and Millions of Floating Point Operations per Second (MFLOPS)) fail to meet the good performance characteristics, and others may be incorrectly interpreted (Lilja 2000), (Hennessy and Patterson 1996). The only available benchmark for e-commerce is TPC-W, designed by the Transaction Processing Council (**www.tpc.org**). The TPC-W primary metrics are Web Interactions Per Second (WIPS). Significantly, Web-server performance metrics include HTTP bytes per second, which may be used to determine device utilization. However, equipment selection is ultimately based upon technical specifications. The TPC-W standard provides no guidance on how HTTP bytes per second may be related to the many different metrics of the heterogenous devices in an e-commerce infrastructure.

Analytical modelling (queuing analysis)

Analytical modelling describes IT systems and sub-systems as resources called servers, and the load on each resource as a series of jobs. Jobs not currently being serviced wait in a queue until the current job is completed. The two principal techniques used are operational and stochastic analysis. In operational analysis there are no assumptions about the distribution of the times required to service jobs and the distribution of job arrival time. Using operational analysis it is possible to calculate utilization, traffic intensity, and average number of jobs in a queue (for example Little’s Law, Quantitative System Performance). Operational analysis can be applied to any system or part of a system and provides a high-level metric of job flow. Stochastic analysis, however, assumes probabilistic (stochastic) distribution of job arrival and service times. Using stochastic analysis it is possible to write flow equations for single queue, single server and more complex single queue, multiple server systems (Kleinrock 1975). Analytical modelling has strong mathematical foundations that are actively being researched with an associated large body of literature. However, it is only a model whose accuracy is dependent upon assumptions made. In Lilja's terms,

> *It is important to bear in mind, however, that the conclusions drawn from a queuing analysis must eventually be applied to a real system if they are to be of any use. The results of this analysis will be accurate only insofar as the assumptions made in the analysis match the system's characteristic. This means that, if the characteristics of the actual system only approximate some of the assumptions made in the queuing analysis, then the conclusions drawn from the analysis will (at best) only approximate what will happen in the actual system.* (Lilja 2000)

For very large systems, queuing analysis is an important technique where the effort required in measuring performance is proportional to the cost of making the wrong decision. However, such considerable mathematical rigour does not lend itself to general use, and may not result in simple cost-effective solutions for small to medium-sized enterprises. A preliminary survey found that none of the IT organisations analysed employed these techniques (Maj 2002). The application of a specific technique depends upon the problem domain, that is, the cost of getting it wrong. Furthermore, all of these techniques model job flow and hence use performance metrics that include jobs per second, response time, service time. However, hardware is chosen using technical specifications. While performance analysis techniques employ abstraction (i.e. queues and service centres), there appears to be no guidelines for relating these abstractions to technical specifications. It remains problematic to relate these different units to the performance of an IT system that consists of a heterogeneous collection of different technologies each with their own performance metrics. In effect it is relevant to ask, 'How can job flow be directly related to technical specifications?' This is important, because it is on this basis that equipment is actually selected.

Customer and Resource models

The performance of an IT system depends also upon the behaviour of end users. Four different reference models for commerce web sites 'business', 'functional', 'customer' and 'resource' have been defined (Menasce, Almeida et al. 1999). The Customer Model Behaviour Graph (CMBG) and Client/Server Interaction Diagrams (CSIDs) are techniques that can be used to capture the navigational patterns of customers during site visits and hence obtain quantitative information about workloads. Though these techniques are useful, a wide range of different performance metrics are still used. For example hits per second, page views per day, click-throughs, transactions per second, CPU time, processor demand (milliseconds). Furthermore there is no simple technique proposed for resource modelling.

Model requirements

Models are used not only as a means of communication and controlling detail but may also forming the basis of a conceptual understanding of a system. According to Cooling there are two main types of diagram: high level and low level. High-level diagrams are task oriented and show the overall system structure with its major sub-units. Such diagrams describe the overall function of the design and interactions between both the sub-systems and the environment. The main emphasis is ‘what does the system do’ and the resultant design is therefore task oriented. According to Cooling, ‘Good high-level diagrams are simple and clear, bringing out the essential major features of a system’ (Cooling 1991). By contrast, low-level diagrams are solution oriented and must be able to handle considerable detail. The main emphasis is ‘how does the system work’. However, all models should have the following characteristics: diagrammatic, self-documenting, easy to use, control detail and allow hierarchical top-down decomposition. By example, the Data Flow Diagram (DFD) model enables a complex system to be partitioned (or structured) into independent units of an amenable size so that the entire system can be more easily understood. It is possible, therefore, to examine a system in overview and with increasing levels of detail, whilst maintaining links and interfaces between the different levels. DFD’s are not only simple, but also graphical; hence serve not only as documentation but also as a communication tool. DFD’s are therefore a top-down diagrammatic representation of information flow within a system, and are a means of defining the boundaries and scope of the system being represented, checking the completeness of the analysis and providing the basis for program specifications. This technique is relatively simple to use, yet powerful enough to control complexity during the analysis and design of both small and large systems. It is recognized that communication with end users is especially important as this helps to validate for correctness.

Bandwidth Nodes

Work to date on the Bandwidth Nodes (B-Nodes) suggests that it meets the requirements of both modelling theory and measurement science (Maj and Veal 2000). Measurement science defines the use of common fundamental units (mass, length and time) from which other, more useful units may be derived, for example, velocity. In keeping with one of the standard principles of measurement science, the B-Node technique proposes bandwidth (Mbytes per second) as a single, common performance metric where **Bandwidth = Clock frequency (MHz) x Data Path Width (Bytes) x Efficiency**. From bandwidth, other more meaningful units (for example images per second, transactions per second) can then be derived. This has the added advantage that the relative performance of heterogeneous technologies, for example, hard disc drive, microprocessor, can then be directly compared using a single, derived performance metric that is directly relevant to users, for example transactions per second. Using the B-Node model it is possible to convert any heterogeneous unit (for example MHz, rpm) to any derived units (for example, text, voice, image, video, medical image) and back again. This technique can be used to not only communicate with the users but also relate the overall performance of the IT system to the performance of the individual components within that system. In effect, it is possible to communicate with the end user in terminology that is directly relevant to them, in keeping with the principles of measurement science.

Experimental Work

A client/server network was established using a Windows 2000 Server with Internet Information Server 5 and a Windows 2000 Professional client. Using Microsoft’s Network and Monitoring Tool http traffic for a variety of commands was captured and analysed (table 1). By example a request for a URL resulted in a total traffic of three packets each 546 bytes long. This is consistent with standard http protocols (Wong 2000).

HTTP Transaction	Client Packets send	Server Packets send	Total Packets	Total data Bytes	Total data Kbytes
Logon	2	3	5	1541	1.5
Search	2	3	5	1477	1.44
Browse (simple text)	1	2	3	519	0.5
Browse (images)	1	4	5	1048576	1024

Table 1: Web Traffic

It is therefore possible to use http transaction as derived units in the B-Node model.

E-Commerce Infrastructure Modelling using B-Nodes

Work to date suggests that this technique is scalable and hence valid for both small such as LANs and also global e-Commerce systems (Maj, Veal et al. 2001). Assuming the HTTP browse transaction in a client/server interaction then this requires 1024 Kbytes for each transaction. The client and server are connected over a 100 MHz network line, which is 11.25 Mbytes/s. The performance of each B-Node can then be evaluated (Table 2).

If the demand on this server is 10 users, it is a simple matter to determine both performance bottlenecks and also the expected performance of the equipment. From table 1 we have calculated the number of Kbytes required for each HTTP transaction. The metric of transactions/s can easily be converted to the fundamental unit of Mbytes/s, which are then used to determine the required performance specifications of the Processor, DRAM and the Hard disk etc.

Device	Clock Speed (MHz)	Data Width	Efficiency	Bandwidth (MBytes/s) B = C x D x E	Number of browse transaction	Demand	U (%)
P II	400	8	0.5	1600	3200	10	0.3
DRAM	16	8	0.5	64	136	10	7.3
Hard D	100rps	250K	0.5	12.5	25	10	40
ISA Bus	8	2	0.5	8	16	10	63
Ethernet	100	1/8	0.9	11.25	23	10	44

Table 2: Server Hardware Specification for Browse (simple text)

Device	Clock Speed (MHz)	Data Width	Efficiency	Bandwidth (MBytes/s) B = C x D x E	Number of browse transaction	Demand	U(%)
P II	400	8	0.5	1600	1563	10	0.5
DRAM	16	8	0.5	64	63	10	14.7
Hard D	100rps	250K	0.5	12.5	13	10	76.9
ISA Bus	8	2	0.5	8	8	10	125
Ethernet	100	1/8	0.9	11.25	11	10	91

Table 3: Server Hardware Specification for Browse (images)

From Table 3 we can see that the ISA bus is more then 100 % utilized for browse transaction of 1 Mbytes page. Thus the bottleneck in the system can be identified. Capacity planning is the process of predicting future workloads and determining the most cost-effective way of postponing system overload and saturation. Assuming that the web traffic is anticipated to rise to 550 transactions/s – the current single server solution will be inadequate. To accommodate much higher web traffic a typical e-business configuration may consist of a front-end Web server, a Secure Web server, a Payments server, an Application server and a Database server. Assuming each server is a separate device connected by a 100Mbps Ethernet link it is possible to model this configuration using B-Nodes. Each server can be represented by a B-Node. The communication link may be represented as a directed arc (arrow) annotated by its bandwidth performance (units MBytes/s or Transactions/s). Using Customer Behaviour Model Graphs (CBMG's) it is possible to evaluate the relative frequency that each dedicated server is used. Assuming probability based on relative frequency our performance equation is now Bandwidth = Clock Speed x Data Path Width x Efficiency x Frequency (B = C x D x E x F).

Device	Network Bandwidth (Mbytes/s)	Network Bandwidth (Transactions/s)	Relative traffic frequency (f)	Actual server load (Mbytes/s)	Actual server load (Transactions/s)
Web server	11.25	1.1k	0.5	5.625	0.55k
Secure server	11.25	1.1k	0.05	0.5625	0.055k
Payment server	11.25	1.1k	0.05	0.5625	0.055k
Database server	11.25	1.1k	0.2	2.25	0.22k
Application server	11.25	1.1k	0.1	1.125	0.11k

Table: 4 E-commerce servers

Conclusions

B-Nodes are:

- Simple, easy to use;
- Diagrammatic;
- Employ abstraction;
- Scalable;
- Use a common fundamental performance metric (Mbytes per second);
- Employ a decimal scaling system;
- Employ derivable units directly relevant to end-users; and

The use of fundamental units allows other, more useful, units to be derived (Transactions/s). Moreover using fundamental units (Mbytes/s) enables the performance of heterogeneous devices to be directly compared and evaluated. Certain implicit assumptions have been made of necessity to simplify the calculations this results in figures that the system must meet to be able to give the required performance. Our above assumptions included no operating system overhead or slowdown, no wait times or collision on the network etc. The authors propose B-Nodes as a possible aid for modelling IT infrastructure. However this needs to be extensively tested.

References

Boehm, B. W. (1984). “A software development environment for improving productivity.” Computer **17**(6): 30-42.

Checkland, P. B. (1981). Systems Thinking, Systems Practice. Chichester, John Wiley.

Coad, P. and E. Yourdon (1991). Object-oriented Analysis. Englewood Cliffs, NJ, Prentice-Hall.

Cooling, J. E. (1991). Software Design for Real-Time Systems. Padstow, Cornwall, Chapman and Hall.

Cutts, G. (1988). Structured Systems Analysis and Design Methodology. London, Paradigm.

Gibb, T. (1988). Principles of Software Engineering Management. Reading, MA, Wesley.

GVU GVU's WWW User Surveys.

Hennessy, J. L. and D. A. Patterson (1996). Computer Architecture A Quantitative Approach. San Francisco, Morgan Kaufmann.

Fenik, H., *Zona Research*, in *Lan Times*. 1998

Shklar, G., in *The New York Times*. 1998: New York

Humphrey, W. S. (1990). Managing the Software process. Reading, MA, Addison-Wesley.

Jones, G. W. (1990). Software Engineering. New York, Wiley.

Kleinrock, L. (1975). Queueing Systems, Volume 2: Computer Applications. New York, John Wiley.

Lilja, D. J. (2000). Measuring Computer Performance - A Practitioner's Guide. Cambridge, Cambridge University Presss.

Maj, S. P. and D. Veal (2000). “Computer Technology Curriculum - A New Paradigm for a New Century.” Journal of Research and Practice in Information Technology **32**(August/September): 200-214.

Maj, S. P., D. Veal, et al. (2001). A New Method for Controlling Information Technology Hardware Complexity. Americas Conference on Information Systems, Boston, USA.

Martin, J. (1991). Rapid Application Development. New York, Macmillan.

Matsumoto, Y. E. and Y. E. Ohno (1989). Japanese Perspectives in Software Engineering Practice. Reading, MA, Addison-Wesley.

S.P. Maj, G. Kohli. (2002, June). Modelling Global IT Structure Using B-Nodes. Paper to be presented Third Annual Global Information Technology Management (GITM) World Conference, New York.

Menasce, D. A., V. A. F. Almeida, et al. (1999). A Methodology for Workload Characterization for E-Commerce Servers. 1999 ACM Conference in Electronic Commerce, Denver, CO, ACM.

Mumford, E. and M. Wier (1979). Computer Systems in Work Design - the ETHICS Method. London, Associated Business Press.

Nauumann, J. D. and A. M. Jenkins (1982). Prototyping: the new paradigm for systems development. MIS Quarterly.

Royce, W. W. (1970). Managing the development of large software systems: concepts and techniques. WESCON.

5.5 Modelling Web site Infrastructure using B-Nodes Theory

Title of the paper : Modelling Web site Infrastructure using B-Nodes Theory

Authors G Kohli, D Veal, S P Maj

Conference: 9^{th} ANZSYS conference

Conference Location: Melbourne, Australia

Conference date: 18^{th}–20^{th} November, 2003

This conference is held annually in the Asia-pacific region. The primary aim of the conference is to address research in the field of information technology, system design and modelling methods. The acceptance rate at this conference was 60 - 65 %.

Models are used not only as a means of communication and to control detail, but may also form the basis of a conceptual understanding of a system. Among the various models analysed for web performance none proved relatively simple to use, yet powerful enough to control complexity during the analysis and design of WWW infrastructure. Previous work undertaken by Maj on B-nodes was concerned with PCs and components inside PCs. Undertaking detailed experiments with PCs and operating system had proved to be problematic. In order to address this concern it was decided to quantify the effects of various communication devices e.g. switches and routes on the bandwidth. B-node theory is based upon bandwidth; in order to measure network bandwidth it is necessary to consider the effects of networking devices upon the transfer time.

The authors undertook a range of experiments starting with the base-line experiment of transferring data between two PCs connected using a crossover cable. The authors then decided to add layer 2 switches, followed by layer 3 routers to investigate the effects of these devices upon the actual performance of the equipment. B-Node theory was used to measure the relative performance and utilization of these networking devices.

The performance of network applications affects productivity across many areas. A model-based approach provides a good foundation for developing solutions to these problems. B-Node models were proposed as an aid to model network infrastructure from the bottom-up and to enable a top-down conceptual understanding of workload characteristics. Providing good conceptual models assists in better understanding of complex systems by people who have limited experience within the field of IT systems.

Modelling Website Infrastructure using B-Node Theory

G. kohli, D. Veal & S.P. Maj
School of Computer and Information Science
Edith Cowan University
Perth, Western Australia
{g.kohli,d.veal,p.maj}@ecu.edu.au

Introduction

A large volume of business is conducted via the Internet (Schneider & Perry, 2000). However, this has resulted in increased transaction delay times as systems and computer networks become overloaded (Devlin et al., 1999). Surveys and studies indicate that slow downloading time is the most often cited reason that an online customer leaves a site and searches for another vendor's site (Bakos, 1998). According to Shklar: *"Sites have been concentrating on the right content". Now, more of them specially e-commerce sites realize that performance is crucial in attracting and retaining online customers*."(Shklar, 1998). The performance of an Internet site is dependant not only upon the behavior of end users using that site but also the performance of the technologies employed. Currently there are a number of different models for defining e-business web sites performance, such as the 'business', 'functional', 'customer', and 'resource' models (Menasce et al., 2000). The Customer Behavior Modelling Graph (Inverardi & Wolf, 1995) (Union, 1996) and Client /Server Interaction Diagrams (CSIDs) (Stohr & Kim, 1998b) are techniques that can be used to capture the navigation patterns of customers during site visits and hence obtain quantitative information on workloads. However, although these models attempt to predict user behavior, they do not provide information about the actual load on the systems running on such sites. In the final analysis user load must be translated to hardware requirements thereby allowing performance bottlenecks to be identified. However, such problems associated with infrastructure design are non-trivial. According to Fenik *"Being able to manage hit storms on commerce sites requires more then just buying more plumbing."(Fenik, 1998)*. In order to predict the workload characteristics of e-commerce sites, effective modelling needs to be undertaken to determine key bottlenecks within the system. It is therefore necessary to investigate different types of models which could be used to model the infrastructure of e-commerce web sites.

Model Requirements:

Models are used not only as a means of communication and controlling detail but may also form the basis of a conceptual understanding of a system. According to Cooling there are two main types of diagram: high level and low level (Grady Booch, James Rumbaugh, & Jacobson, 1999; Cooling, 1991). High-level diagrams are task oriented and display overall system structure and major sub-units. Such diagrams describe the overall function of both the design and interactions between both the sub-systems and the environment. The main focus is upon finding answers to the question what does the system do? According to Cooling, "*Good high-level diagrams are simple and clear, bringing out the essential major features of a system*" (Cooling, 1991). By contrast, low-level diagrams are solution oriented and must be able to handle considerable detail. The main emphasis is 'how does the system work'. However, all models should have the following characteristics: diagrammatic, self-documenting, easy to use, control detail and allow hierarchical top-down decomposition. For example, the Data Flow Diagram (DFD) (Hawryszkiewycz, 2001) model enables a complex system to be partitioned (or structured) into independent units of an amenable size so that the entire system can be more easily understood. It is possible, therefore, to examine a system in overview and with increasing levels of detail, whilst maintaining links and interfaces between the different levels. DFD's are not only simple, but also graphical; hence they serve not only as documentation but also as a communication tool (R. S. Pressman, 1992). DFD's are therefore a top-down diagrammatic representation of information flow within a system, and are a means of defining the boundaries and scope of the system being represented, checking the completeness of the analysis and providing the basis for program specifications. This technique is relatively simple to use, yet powerful enough to control complexity during the analysis and design of both small and large systems. It is recognized that communication with end users is especially important as this helps to validate a model for correctness. There are various high level models that are used to evaluate web site performance.

Web Performance

There are various well established methods for evaluating Internet site performance (Menasce, Almendia, & L.Dowly, 1994). The Customer Behaviour Modelling Graph (CBMG) can be used to measure aggregate metrics for web sites (Menasce & Almendia, 1999). Using this modelling technique it is possible to obtain a wide variety of different performance metrics that include: hits/s, unique visitors etc. However, when using this technique it is not possible to relate these metrics to hardware specifications. The Client /Server Interaction Diagrams (CSIDs) (Stohr & Kim, 1998b) can be used to capture the navigation patterns of customers during site visits and hence obtain quantitative information on workloads. However, it does not provide any insight into how the workload will affect the underlying infrastructure.
Furthermore, the World Wide Web (WWW) has some unique characteristics that distinguish it from traditional systems (Almedia et al., 1996; Almeida et al., 1997; Arlitt & Williamson, 1996; Mogul, 1995). Firstly, the number of WWW clients is in the range of tens of millions and rising. Secondly, the randomness associated with the way users visit pages makes the problem of workload forecasting and capacity planning difficult. (Menasce, Almendia, & Dowdy, 1994).
Benchmarks are the standard metrics used in defining the scalability and performance of a given piece of hardware or software. For example the Adaptive Computing System (Sanjaya et al., 2000) is a collection of benchmarks that focus upon specific characteristics from the start of a computation until its completion. Benchmarks evaluate the ability of a configurable computing infrastructure to perform a variety of different functions. SPECWEB and TPC-C (W. D. Smith, 2000) are notable benchmarks in the e-business environment**.** These benchmarks come close to representing the complex environment of an e-business workload. Benchmark programs are used for evaluating computer systems. Different end user applications have very different execution characteristics; hence there exists a wide range of benchmark programs. The four main categories are: science and engineering (examples include; Whetstones, Dhrystones, Livermore loops, NAS kernels, LINPACK, PERFECT club, SPEC CPU), Transaction Processing (for example; TPC-A, TCP-B, TPC-C), server and networks (examples include;

SPS/LASSIS, SPEC web) and general use (examples include; AIM Suite III, SYSmark, Ziff-Davis PC Benchmark). However none of these benchmarks are directly relevant to E-commerce web transactions. The Transaction Processing Council introduced the TPC-W that simulates the workload activities of a retail store Web sites (W. D. Smith, 2000). In the TPC-W standard the products are books and the user is emulated via a remote Browse that simulates the same HTTP traffic as would be seen by a real customer using the browser. E-business sites have transient saturation so it is hard to use these benchmarks to get the correct idea about the actual load generated on the web servers. Benchmarks currently in use fail to measure the web performance characteristics, whilst others may be incorrectly interpreted (Humphrey, 1990; Lilja, 2000). According to Skadron, "*Research cannot pursue futuristic investigation when they are limited to systems for which no benchmark programs are available. The current short coming in computer systems evaluation could ultimately even obstruct the innovation that is driving the information technology revolution*" (Skadron et al., 2003). Furthermore the basic problem still remains. Using benchmarks it is not possible directly relate the technical specification to the metrics used in the service level agreements. The difficulties of developing effective models for large networks are becoming greater as noted by Clark, "*As networks grow to connect millions of nodes, and as these nodes all communicate in unpredictable patterns, the resulting behaviour becomes very difficult to model or predicts*" (Clarke & Pasquale, 1996). The question was then asked, what methods do IT web site managers use to design and manage the performance of an Internet web site?

Commercial Practices:

A questionnaire was distributed to several small to medium size companies in Western Australia. The results indicated that infrastructure requirements were typically based upon past experience and also purchasing the highest performance equipment within budget constraints (Maj & Kohli, 2002). Alternatively companies outsourced this problem to vendors. These approaches are arguably entirely unsatisfactory as they relegate IT systems analysis to conventional 'wisdom' and mythology (Maj & Kohli, 2002). None of the companies analysed employed any techniques for modelling infrastructure performance. The scope of the above survey is currently being extended both within Australia and internationally. From the data gathered to date it can further be concluded that most companies where not aware of an effective model that could applied to effectively model e-commerce website workloads. Hence there is current need for such a new model in this area.

Modelling infrastructure using B-Nodes:

Computer and network equipment is complex. Furthermore they use a wide range of heterogeneous technologies with different performance metrics. By example hard disc drive performance is often quoted in rpm; electronic memory performance is quoted in nanoseconds; microprocessor performance is quite in MHz etc. This results in two problems. Firstly the performance of a web site (server with switches, hubs etc) depends upon the speed of the slowest device. It is not possible, using these metrics, to easily determine the relative performance of each device. Is 10ns electronic memory faster or slower than a 1GHz microprocessor? Secondly it is difficult to relate the technical performance metrics to user requirements defined in the Service Level Agreement. Can a hard disc drive operating at 5,000 rpm deliver 100 web pages per minute? The B-Node model was proposed to address these problems (Maj & Veal, 2000). The B-Node model:

- Can be used to model a wide range of computer and network technology equipment
- Is diagrammatic, self documenting and easy to use
- Uses recursive decomposition, hence can be used to model both small systems (e.g. a server) or a larger system (e.g. an Intranet)
- Uses a common performance metric (Mbytes/s). Hence the performance of heterogeneous technologies can easily be compared
- Uses a common fundamental unit (Mbytes/s) allowing other units to be derived. Hence it is possible to define the performance of a wide range of different technologies using, for example, a common, derived metric such as web pages per second.

The B-Node model has been used to model an E-commerce server and hence identify hardware bottlenecks. It has also been used to evaluate the performance of different E-commerce serves (Web server, payment server etc) (Maj et al., 2001).
However, the use of bandwidth as a sole indicator of performance may be problematic. According to McComas notes there are problems due to bandwidth and latency (McComas, 2001), as does Buzen and Shum, (Buzen & Shum, 1996). This point is also made with respect to network technology by Openhiemer

> *"It is possible to improve throughput such that more data per second is transmitted, but not increase goodput, because the extra data transmitted is overhead or retransmissions …more data is transmitted per time, but the user sees worse performance. …most end users are concerned about throughput rate for applications. Marketing materials from some networking vendors refer to application-layer throughput as 'goodput'. Calling it goodput sheds light on the fact that it is a measurement of good and relevant application-layer data transmitted per unit time"* (Oppenheimer, 2001).

In effect, it is possible to have higher bandwidth but it is not being used effectively to transfer data. Hennessy also notes the: "*… pitfall of using bandwidth as the only measure of network performance. … this may be true for some applications such as video, where there is little interaction between the sender and the receiver, but for many applications such as NFS, are of a request-response nature, and so for every large massage there must be one or more small messages … latency is as important as bandwidth*" (Hennessy & Patterson, 1996). In spite of this the B-Node model has many potential advantages and it may be possible to address the latency issue.

Using B-Nodes to measure Network Technology performance:

A wide range of different files was transferred between two PCs using a simple cross over cable using FTP. This represented the base line performance. Then a range of different networking technologies were introduced and the performance measured. In order to address the problem of latency the authors have subsumed the effects of latency under a definition of bandwidth. Namely bandwidth = the size of the file in Mbytes / total time to send that file:

$$\mathbf{B = L_1 / T_T}$$

Where $T_T = t_1 + t_L$
B = (bits passed)/ (time taken to pass those bits)
L_1 = Length of the files in Mbytes.
t_1 = Time required to transfer the file
t_L = Latency measured in (msec)

Table 1 shows a summary of various devices with respect to bandwidth using File Transfer Protocol (FTP):

Technology	Bandwidth (Mbytes/s)
PC crossover cable PC	11.5
PC switch PC	11.5
PC router PC	7.5

Table 1 B-Node performance figures (Mbytes per second)

The crossover cable between two PCs can be modelled as a B-Node with a performance of 11.5Mbytes/sec. A switch can be modelled as a B-Node with a performance also of 11.5Mbytes/sec. In effect a switch works at 'wire speed' and has no measurable affect on performance. A router modelled as a B-Node gives a performance of 7.5 Mbytes/sec. The use of common fundamental units allows two different technologies (layer 2 switches and layer 3 routers) to be compared. Furthermore, common derived units can be used. Assuming the messages in a web transaction are 10 Kbytes each and the load is 1000 per second. It can then we concluded from Table 2 by introducing a Router will create a bottleneck in the system as the utilisation is more then 100%. Additionally it is possible to identify, using meaningful metrics the relative performance of each technology which can then help network designer to better design web sites infrastructure.

Technology	Bandwidth (Mbytes/s)	Transaction size (Kbytes)	Load	Utilization
PC crossover cable PC	11.5	10	1000	86%
PC switch PC	11.5	10	1000	86%
PC router PC	7.5	10	1000	133%

Table 2 B-Node performance figures (Transactions per second)

The authors are further developing the experiment by taking into account different protocol like HTTP and HTTPS and the use of Access control list (ACL).

Conclusion:

The performance of Network application affects the productivity in many areas: e-commerce, a model base approach provides a good foundation for developing solutions to these problems. The B-Node model is simple, diagrammatic and self-documenting modelling technique, it use common fundamental units, which can help the network designer to identify the key bottlenecks within the system. The B-Node model is undergoing development and testing in an attempt to model infrastructure from the bottom-up to enable top-down conceptual understanding of workload characteristics.

References:

Almedia, V., A. Bestravos, et al. (1996). Characterization reference locality in the WWW. IEEE-ACM PDIS.

Almeida, J. M., Virgilio Almeida, et al. (1997). "WebMonitor: a Tool for Measuring World-Wide Web Server Performance."

Arlitt, M. and C. Williamson (1996). Web server Workload Characterization. ACM SIGMETRICS Conference of Measurement of Computer Systems, Philadelphia, PA.

Bakos, Y. (1998). "The Emerging Role of Electronic Market Places on the Internet." ACM **41**(8).

Booch, G., James Rumbaugh, et al. (1999). The Unified Modelling Langauge User Guide, Addision Wesley Longman, Inc.

Buzen, J. P. and A. N. Shum (1996). Beyond Bandwidth: Mainframe Style Capacity Planning for Networks and Windows NT. Proceedings of the 1996 Computer Measurement Group Conference., Olando Florida USA.

Clarke, D. and J. Pasquale (1996). "Strategic Directions in Networks and Telecommunications." Computing Surveys **28**(4): 679 - 690.

Cooling, J. E. (1991). Software Design for Real-Time Systems. Cornwall, Chapman and Hall.

Devlin, B., Gray. J, et al. (1999). Scalability Terminology: Farms, Clones, Partitions and Packs: RACS and RAPS,, Microsoft Research.

Fenik, H. (1998). Zona Research. Lan Times.

Hawryszkiewycz, I. (2001). Introduction to Systems Analysis & Design. Malaysia, Prentice Hall.

Hennessy, J. L. and D. A. Patterson (1996). Computer Architecture A Quantitative Approach. San Francisco, Morgan Kaufmann.

Humphrey, W. S. (1990). Managing the software process. MA, Addison-Wesley, Reading.

Lilja, D. J. (2000). Measuring Computer Performance - A Practitioner's Guide. Cambridge, Cambridge University Press.

Maj, S. P., D.Veal, et al. (2001). A New Method for controlling Information Technology Hardware Complexity. American Conference on Information Systems, Boston, USA.

Maj, S. P. and G. Kohli (2002). B-Node: A bridge between the business model, information model and infrastructure model. Seventh Caise/IFIP-WG8.1 Internation Workshop on Evaluation of Modelling Methods in System Analysis and Design, Toronto.

Maj, S. P. and D. Veal (2000). "Computer Technology Curriculum - A New Paradigm for a New Century." Journal of Research and Practice in Information Technology **32**(August/September): 200-214.

McComas, B. (2001). Can the Pentium 4 Recover?, InQuest Market Research. **2001**.

Menasce, D. A. and V. A. F. Almendia (1999). "A metholody for workload characterization for e-commerce server." ACM conference in Electronic Commerce, Denver, CO, ACM.

Menasce, D. A., V. A. F. Almendia, et al. (1994). Capacity Planning and Performance Modelling: From Mainframes to Client-Server Systems. Upper Saddle Rive, NJ, Prentice Hall.

Menasce, D. A., V. A. F. Almendia, et al. (1994). Capacity Planning and Performance Modelling. N J, Prentice Hall.
Menasce, D. A., A. F. Virgilio, et al. (2000). Scaling for E-Business: Technologies, Models, Performance, and Capacity Planning.
Mogul, J. C. (1995). Operating system supportfor busy Internet servers. Proceedings of the fifth workshop on Hot Topics in Operating Systems.
Oppenheimer, P. (2001). Top-down Network Design: A systems analysis approach to enterprise network design. Indianapolis IN, Cisco Press.
Pressman, R. S. (1992). Software Engineering A practitionar's Approach. Singapore, McGRAW-HILL International Editions.
Sanjaya, K., N. Chirag, et al. (2000). "A Benchmarking suite for evaluating configurable computing systems- Status, Reflections, and Future Directions." Eighth International Symposium on Field- Programmable Gate Arrays, Monterey, California.
Schneider, G. P. and J. T. Perry (2000). Electronic Commerce, Course Technology.
Shklar, G. (1998). The New York Times. New York.
Skadron, K., M. Martonosi, et al. (2003). Challenges in Computer Architecture Evaluation. Computer. **36:** 30-36.
Smith, W. D. (2000). TPC-W: Benchmarking An Ecommerce Solution, TPC.
Stohr, E. A. and Y. Kim (1998). A Model for Performance Evaluation of Interactive Systems. 31st. Annual Hawaii Internternational Conference on System Sciences.

5.6 An examination of Vendor Based Curricula in higher and further education in Western Australia

Title of the paper :	An examination of Vendor Based Curricula in higher and further education in Western Australia
Authors	G Murphy, G Kohli, D Veal, S P Maj
Conference:	ASEE 2004 conference
Conference Location:	Salt Lake City, Utah
Conference date:	20th – 23rd June 2004

This paper was presented at American Society for Engineering Education. This conference is held annually in the USA. The aim of this conference is to address key issues in engineering education. The conference has over 1000 delegates from various fields of engineering education e.g. civil engineering, mechanical engineering, and software engineering. The acceptance rate varies depending upon the division under which the paper is submitted. The overall acceptance rate was around 70 %

Vendor-Based Curricula (VBC) is becoming increasingly prevalent in two-year college (Technical and Further Education (TAFE)) courses and in University programs in Western Australia. This paper examines the strength and weaknesses of offering curriculum over which universities and college have no control of content and standards.

Traditionally the main task of universities, and their major area of expertise, is the provision of education and research. Relevant workplace experience may be difficult for students to obtain and its provision is likely to become a contributing factor in their selection of courses. Computer science graduates currently demand employment related education as mentioned earlier in this thesis. Some universities in Australia have gradually incorporated vendor-based programmes as a part of their courses. Some of the advantages of vendor-based programmes are:

- The material is supplied free of charge to participating educational institutions worldwide.
- The material is continually updated to reflect the rapid technological changes
- The course is provided on-line and in several major languages
- A major component of the program is hands-on training

At the same time there are number of potential problems associated with the provision of vendor-based curricula. These include:

- The accusation may be levelled that vendor-based curriculum units emphasise vendor specific solutions at the expense of a solid grounding in underlying theory.
- There is large amount of material for students to learn, a lot of which is knowledge recall.
- Resistance to the inclusion of what many educationalists classify as essentially training rather then education. This ongoing debate is unlikely to be resolved in the near future.

In order to discover students' views on vendor-based curricula a questionnaire was given to different student groups from which the following conclusions were drawn:

- A majority of students believed that studying vendor-based course improved their understanding of other topics in networking.
- A majority of students were confident that they could use the knowledge and skills learned in a work environment and believed that this added value to their degree or diploma
- Some students were enrolled in VBC to get promoted in their current workforce. This case result was more pronounced in the TAFE students.

An Examination of Vendor-Based Curricula in Higher and Further Education in Western Australia

G. Murphy, G. Kohli, D. Veal and S. P. Maj
Edith Cowan University, Perth, WA, Australia

Abstract

Vendor-based curricula are becoming increasingly prevalent in two-year college (Technical and Further Education (TAFE) courses and in University programs in Western Australia. This reflects a world-wide trend in the provision of such programs; for example, in October 2003 Cisco Systems reported that there were over half a million students enrolled in Cisco Networking Academies in 150 countries around the world. In Western Australia, vendor-based curricula, such as the Cisco Certified Network Associate (CCNA) program, the Cisco Certified Network Professional (CCNP) program and the Microsoft Certified Systems Engineer (MCSE) program are offered for credit in TAFE Engineering and Information Technology (IT) Diplomas and in Bachelor and/or Masters Degrees in three of the five universities based in the State. In this paper we seek to examine the reasons why students enrol in the courses, and what career benefits they believe will accrue as a result of their studies. The paper will conclude with an evaluation of the strengths and weaknesses of offering curriculum over which universities and college have no control of content and standards.

Introduction

According to Nelson and Rice, "…*in today's business world the ideal employee has three critical components: education, certification and experience…*' [17] . Traditionally the main task of universities, and their major area of expertise, is the provision of education. Experience is difficult for students to obtain and its provision is likely to become a contributing factor in their selection of courses [1, 8, 14]. Certification has been addressed by the industry itself. Major IT companies are now endorsing training specifically tailored to the use of their product lines. These include companies such as Cisco, Microsoft and Novell. These companies have implemented [2, 5, 12], or have endorsed their own certification programmes [16] . Such qualifications are known as 'Vendor Certifications' [7, 10, 15]. Units that incorporate Vendor Based Curricula (VBC) are very different in nature to the traditional offerings of the university sector. Hornbaker notes that:

"Over the years, vendors have created their own certification programs because of industry demand. The demand arises when the marketplace needs skilled professionals and an easy way to identify them. Vendors benefit because it promotes people skilled in their product. Professionals benefit because it boosts their careers. Employers benefit because it helps them identify qualified people" [10].

The introduction of these certifications has forced universities and Technical and Further Education (TAFE) colleges within Australia to re-examine the content and methods of delivery of their units. This in turn has led to the adoption by many educational institutions of VBC, where the curricula are under the control of the vendors and not the institutions themselves. It should however be noted that the form of delivery or assessment is not prescribed by the vendors, although the educational institutions must conform to the vendors' quality control measures. A major player in the area of internetworking vendor based education and training is Cisco Systems. The Cisco Networking Academy Program (CNAP) is offered online in 150 countries, in most major languages, around the world. On 18 December 2003, the Academy website reported that there were 454,657 students enrolled in 10,236 academies in 150 countries around the world [4]. The CNAP offers courses in networking, computer hardware and operating systems, Java, Unix, Web design, wireless networks and security. For many of its programs, Cisco has available textbooks that map to and supplement the CNAP Web-based content [19, 20]. Cisco's quality control procedures mandates that all of the users of such curricula, the teachers, lecturers and trainers, are required to attend and pass "train the trainer" sessions prior to delivering the material to students. The adoption of Cisco VBC can result in a number of advantages for participating academies:

- The material is supplied free of charge to participating educational institutions worldwide.
- Course material is provided on-line essentially 24 hours a day for seven days a week, and is supplemented by a large selection of books [12, 13, 9, 18], simulators, examination questions and other learning material.
- The material is continually updated to reflect rapid technological change.
- Laboratory equipment is provided at substantial discount.
- Equipment provided through the program can be used for other, non-vendor based units, and for research purposes.
- Staff are trained and up-skilled through required train the trainer courses.
- The hierarchical structure of participating institutions means that staff training and support can usually be provided locally to any new academies.
- A strong sense of community and cooperation is developed among participating institutions and their staff.
- The globalisation of the program means that students can overcome study interruptions caused by international or inter-state relocation.
- The course is provided on-line and in several major languages.
- A major component of the program is hands-on training.
- The programs reflect the reality of the industry, in that for example internetworking is carried on Cisco equipment, Web design uses Adobe software, and Unix training is undertaken using Sun Solaris [4].

There are of course a number of potential problems associated with the provision of VBC within a traditional education format. These include:

- Resistance to the inclusion of what many educationalists classify as essentially training rather than education. This ongoing debate is unlikely to be resolved in the near future. Denning notes that:

"Learning the professional practices of a speciality of information technology is every bit as important as learning the intellectual core of computing. The mark of a well educated professional will be a balance of the two, and perhaps through partnerships and training companies. The current academic inclination to disdain skills-specific training does not fit a profession" [6].

- Each unit of a VBC generally requires more time than is available under the university course structure as reported. For example, in one university in Western Australia, four hours per week for 12 teaching weeks, for a total of 48 student contact hours, are allocated to deliver a CNAP program that Cisco recommend should take 80 hours.
- Universities and colleges may believe that they will lose control of the material delivered to satisfy a particular unit and hence cannot control quality.
- Although laboratory equipment is provided at significantly discounted rates, the purchase of such equipment can place heavy demands on already stretched educational institutes' budgets.
- There is large amount of material for students to learn, a lot of which is knowledge recall.
- Since only vendor-certified staffs are permitted to deliver the material, schools lose some flexibility with respect to allocation of staff to units. Of course from a quality control perspective, this must be regarded as a benefit to the students.
- The accusation may be levelled that VBC units emphasise vendor specific solutions at the expense of a solid grounding in underlying theory.
- Vendors are able to change curriculum standards without reference to university.

The Questionnaire

Given the above lists of potential advantages and disadvantages, we sought the opinions of students enrolled in units delivered using VBC in colleges and universities where these units are taught. The student survey was administered anonymously, and collected in such a manner that students' anonymity was guaranteed. The authors administered the questionnaire to four groups of students:

- CCNA students at a university in Western Australia (WA). This group complete the four Cisco semesters over two university semesters, for a total of 96 hours contact time (48 hours per semester). There are no entry prerequisites, and this permits cross-faculty and cross-institutional enrolment. This method of delivery requires the students to complete a significant amount of self directed study. For example, students are required to read curriculum in their own time, and use class contact for condensed curriculum-based lectures, practical exercises, and formative on-line quizzes and tests. The unit is assessed by a hands-on, competency-based test, assignment and a written final paper.

- CCNA students at a TAFE (two-year community college) in WA. This group complete the four Cisco semesters over two college semesters, for a total of 160 hours contact time (80 hours per semester). This college also has entry prerequisites in electrical and electronic theory, mathematics and data communications. The units are assessed using the Cisco on-line final tests and a competency-based practical test.
- CCNP students at a university in WA. This group completes one CCNP unit a university semester, for a total of 288 contact hours for the four semesters (48 hours per semester). The entry prerequisite is passes in the CCNA-based units (but not necessarily the certification examination), or a CCNA certification. The units are assessed by a hands-on, competency-based test and a written final paper.
- CCNP students at a TAFE college in WA. This group completes one Cisco semester per TAFE semester, for a total contact of 320 hours (80 hours per semester). The prerequisite is CCNA certification or a pass in the TAFE CCNA course. This course has no college credit, and no internal assessment. These students pay full fees, and their objective is to obtain CCNP certification, although the students complete several case studies. This student population is much smaller (25%) of the populations surveyed in the other three groups.

Student Survey Results

The questions were grouped under the following topics:

1. Reasons for studying the unit:

To Obtain University/College credit (by percentage, rounded)						
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Not Applicable
CCNA University	6	0	13	56	19	6
CCNA TAFE	18	24	12	29	6	12
CCNP University	0	5	10	55	30	0
CCNP TAFE	0	20	0	40	0	40

Table 1

To obtain certification (by percentage, rounded)						
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Not Applicable
CCNA University	13	0	13	44	31	0
CCNA	0	0	6	12	82	0

TAFE						
CCNP University	0	5	20	45	30	0
CCNP TAFE	0	0	0	0	100	0

Table 2

To add value to my degree/diploma (by percentage, rounded)						
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Not Applicable
CCNA University	0	0	0	50	50	0
CCNA TAFE	0	0	24	47	29	0
CCNP University	0	0	5	25	70	0
CCNP TAFE	0	0	0	0	60	40

Table 3

Some interesting points arise from tables 1 to 3:

- Although CCNP TAFE students (fee for service) are primarily interested in certification (100%), and know that there is no college credit for the course per se, the results may reflect awareness of the fact that it is possible to obtain university and college credit when supported by the certification exam.
- Only 36% of TAFE students actively seek college credit, but 100% seek certification. This may reflect the fact that the survey group were an evening class group, and as such are more likely to be selecting classes with a view to furthering their employability.
- All groups believe that this training will add value to degrees/diplomas, although the CCNP TAFE and CCNA TAFE groups have significant percentages that are either neutral or believe the question not applicable. This may again reflect the industry focus of these two groups.

Benefits of studying VBC:

Studying in this course has not improved my understanding of other topics in computing.						
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Not Applicable
CCNA University	13	50	13	19	0	6
CCNA TAFE	12	76	6	6	0	0
CCNP University	15	55	15	10	5	0
CCNP TAFE	20	60	20	0	0	0

Table 4

Two points arise from table 4:

- Most students in all groups believe that studying the material in their program has improved their understanding of other topics in computing. Interestingly, this is true for the students in both CCNP groups, who must have completed significant study to reach that level.
- Some students in the University CCNA group believe that this question does not apply. This may be an indication of the fact that there are no prerequisites for the course, and is offered as an optional unit to all students. There have been cases of performing arts students on other hands-on computing and networking units, and such students might have little or no knowledge of these topics to be improved.

Studying in this course has improved my understanding of other topics in networking.						
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Not Applicable
CCNA University	0	0	6	63	25	6
CCNA TAFE	0	0	0	71	29	0
CCNP University	0	0	5	65	30	0
CCNP TAFE	0	0	0	40	60	0

Table 5

Table 5 shows that the vast majority of students in all groups believe that the CNAP courses improve their understanding of other topics in networking, although again a small percentage of CCNA University students believe that this question does not apply.

I am confident that I could use the knowledge and skills I have learned in this course in a work environment.						
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Not Applicable
CCNA University	0	6	6	50	31	6
CCNA TAFE	0	0	0	65	35	0
CCNP University	0	10	10	40	40	0
CCNP TAFE	0	0	0	20	80	0

Table 6

The authors regard the results of the question in the above table 6 as highly significant. Improving employment prospects is the predominant reason students undergo university education [3] (Campus Review, 1996). Furthermore, students have noted the importance of their ability to apply knowledge in the workplace [11](Kelly and Else, 1996). Furthermore, table 6 shows that at least 80% of students agree or strongly agree that they are confident of their ability to use their learning in a work environment, which indicates student opinion of value of this program.

The course strikes the right balance between the underlying theory and vendor specific material.						
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Not Applicable
CCNA University	0	6	25	50	13	6
CCNA TAFE	0	6	41	53	0	0
CCNP University	0	5	10	70	10	5
CCNP TAFE	0	0	0	100	0	0

Table 7

A major concern of academics has been a perception that VBC emphasises vendor specific skills and knowledge at the expense of underpinning theory. The results from table 7 show that the students believe that this is not the case. It may be argued that students are least able to comment on their lack of knowledge – how can they identify that which they do not know? However, if the students are required to carry out complex internetworking exercises as required in these programs and lack the underpinning knowledge, they cannot complete the task. The results for the question in the above table indicate that across all groups a majority agree that the balance is correct. It should be noted that both CCNP groups, who could be regarded as a more sophisticated population, have significantly higher levels of agreement than the CCNA groups.

3. Students satisfaction with course:

I will not recommend this course to other people						
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Not Applicable
CCNA University	13	69	0	6	6	6
CCNA TAFE	6	94	0	0	0	0
CCNP University	15	75	5	0	0	5
CCNP TAFE	80	20	0	0	0	0

Table 8

The authors used this question to judge students' levels of satisfaction with the courses. Table 8 shows that in all groups an overwhelming percentage indicated that they would recommend the course to others.

4. Comparison of the VBC courses to non-VBC courses at the same level:

The vendor based course needs more of my time than other courses.						
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Not Applicable
CCNA University	6	0	19	44	25	6
CCNA TAFE	0	12	12	47	24	6
CCNP University	0	10	5	55	15	15
CCNP TAFE	0	40	20	0	0	40

Table 9

Table 9 shows that CCNA University students, CCNA TAFE students, and the CCNP university students have provided similar responses to this question. In all three cases, the vast majority (around 70%) believe that the courses require more time than other courses from their respective institutions at the same level. In all three cases, there is a significant percentage to whom the question is not applicable. These are likely to be students enrolled in a single unit. The CCNP TAFE students are unlikely to be enrolled in other courses.

The vendor based course is harder than the other courses.						
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Not Applicable
CCNA University	6	13	44	31	0	6

CCNA TAFE	0	12	29	29	18	12
CCNP University	0	20	25	45	0	10
CCNP TAFE	0	0	40	20	0	40

Table 10

Table 10 responses show a range of opinions, although leaning towards agreement.

Currently the authors are undertaking further research involving staff and faculty teaching VBC and non-VBC networking units.

Conclusions

Although VBC can present potential problems not present in more traditional university and TAFE units. There are many possible benefits that can arise from the incorporation of VBC. Many students believed that VBC based units improved their prospects of gaining employment in the networking industry. Also a majority of students believed that studying on a vendor based course improved their understanding of other topics in networking, were confident that they could use the knowledge and skills learned in a work environment and believed it added value to their degree or diploma. Overall the student response to VBC based curricula was very positive. VBC can involve extra expense for institutions in the form of equipment purchasing requirements and the cost of staff training. There is also an added restriction of the equipment to use trained staff to deliver VBC curricula.

Bibliography

1. Abelman, C. A Parallel Universe. Change, 32(3). 20-29. (2000).
2. Abelman, C. A Parallel Universe, Expanded: Certification in the Information Technology Guild. AAHE. (2000).
3. Campus Review. Educating the Workforce for the New Millennium. Campus Review. 1-7. (1996).
4. Cisco. URL: http://cisco.netacad.net/public/index.html, Cisco Academy Connection. Cisco Systems. (2003).
5. Clarke IV, D.J. Novell's CNA study guide for NetWare 5.1. Provo, UT: Novell Press. (2001).
6. Denning, P.J. Computing the Profession. Educom Review. 1- 20. (1998).
7. Fage, D., Agosta, J., Merchant, D., Foltz, R. and Barnes, J. CCNP Remote access study guide. Berkley, CA: Osborne/McGraw-Hill. (2000).
8. Grubb, W.N. The bandwagon once more: Vocational preparation for hi-tech occupations. Harvard Educational Review. 54 (1). 429-451. (1984).
9. Heap, G. & Maynes, L. CCNA practical studies. Indianapolis, IN: Cisco Press. (2002).
10. Hornbaker, R.D. How to take a Cisco Certification. In CCNP Test Yourself Practice Exams. xxxi-liii. Berkley, CA: Osborne McGraw-Hill. (1999).

11. Kelly, P. & Else, D., Working with Industry. in Technology in Education and training Conference IIR.6, Sydney, NSW, Australia: 25-27. (1996).
12. Lammie, T. CCNA Cisco network associate study guide. San Francisco, CA: Sybex. (2002).
13. Lammie, T., Timm, C. & Odom, S. CCNP/CCIP BSCI study guide. San Francisco, CA: Sybex. (2002).
14. Maj, S. P., Fetherston, T., Charlesworth, P. & Robbins, G., Computer & Network Infrastructure Design, Installation, Maintenance and Management: A proposed new competency based curriculum. In The Third Australasian Conference on Computer Science Education. The University of Queensland. Brisbane, QLD, Australia: (1998).
15. Meyers, M. A+ All in One Certification Exam Guide. Berkeley, CA: Osborne/McGraw-Hill. (2001).
16. Microsoft. Frequently asked questions about adaptive exams. (2000).
17. Nelson, L.M. &Rice, D. Integrating third party certification with traditional computer education. Journal of Computing in Small Colleges, 17 (2). 280-287.
18. Odom, W. Cisco CCNA Exam 640-607 Certification Guide. Indianapolis. IN: Cisco Press. (2002).
19. Cisco. Cisco Networking Academy Program CCNA 1 and 2 Companion Guide. Indianapolis, IN: Cisco Press. (2004).
20. Cisco. Cisco Networking Academy Program CCNA 3 and 4 Companion Guide. Indianapolis, IN: Cisco Press. (2004).

5.7 Abstraction in Computer Network Education: A model based approach

Title of the paper :	Abstraction in Computer Network Education: A model based approach
Authors	G Kohli, S P Maj, G Murphy, D Veal
Conference:	ASEE 2004
Conference Location:	Salt Lake City, Utah
Conference date:	20th- 23rd June 2004

This paper was presented at American Society for Engineering Education. This conference is held annually in the USA. The aim of this conference is to address key issues in engineering education. The conference had over 1000 delegates from various fields of engineering education e.g. civil engineering, mechanical engineering, and software engineering. The acceptance rate varies depending upon the division under which the paper is submitted. The overall acceptance rate is around 70 %.

Rapid developments in network technology have resulted in the inclusions of ACM/IEEE recommendations for Net-Centric computing as part of the computer science undergraduate body of knowledge. Traditionally, computer network courses have not provided students with hands-on access to networking equipment and software. However, due to increasing popularity of vendor-based courses, students now have opportunity to study a more practical approach and hence program networking devices e.g. switches and routers.

This paper demonstrates that certain vendor specific courses fail to provide a good conceptual model of device operation, which is contrary to the education theory of constructivism. According to this theory, students construct new ideas based upon their current/ past knowledge. A student's mental model may or may not be correct and may require reorganisation to align with accepted current practices and theory.

The authors proposed two simple conceptual models of switch and router operations. The conceptual model assists student understanding, and may be useful in students learning. Experiments were carried out to measure the effect the use of these models had upon students' learning. It was concluded from the experiments that students who were taught using these models demonstrated conceptual understanding of router and switch operations. This understanding was more in accordance with that of an expert in the field, than those students who undertook the Cisco Network Academy Program (CNAP) and who were not taught using this model.

Abstraction in Computer Network Education: A model based approach

G. Kohli, S. P. Maj, G. Murphy & D. Veal
Edith Cowan University, Perth, WA, Australia

Abstract

Rapid developments in network technology have resulted in the inclusion of ACM/IEEE recommendations for Net-Centric computing as a part of the Computer Science Undergraduate Body of Knowledge. Accordingly ACM/IEEE networking curriculum now represents an increasingly significant component of Computer Science curriculum. Furthermore, relatively inexpensive equipment, such as switches and routers, and associated on-line vendor based curricula, such as CCNA, CCNP, are now readily available. This approach to network technology education requires an understanding of switch and router operation. However, an extensive analysis of educational materials in this area has indicated that these devices are typically treated as 'black boxes'. Such an approach may not be best suited to the promotion of learning as students are required to construct their own mental model of the internal operation of such devices and which may, or may not, be correct. To address this problem a state model has been designed for both switches and routers which allows complexity to be controlled and hence can be used as a basis for teaching both introductory and advanced courses. These models have been used as the pedagogical foundation for both undergraduate and postgraduate curricula in network technology and the results evaluated. Work to date suggests that these models strongly support student learning at all levels. A wide range of students were analysed and significantly students who had studied a number of networking units but had not been taught via these models scored lower than novice students taught using such models.

Introduction

Rapid developments in network technology have resulted in the inclusion of ACM/IEEE recommendations for Net-Centric computing within the Computer Science Undergraduate Body of Knowledge. Accordingly ACM/IEEE networking curriculum now represents an increasingly significant component of Computer Science curriculum. A central issue within computer network education is the hands-on laboratory-based approach versus the traditional in-call lecture-based approach. Traditionally, computer networks courses have not provided students with hands-on access to networking equipment and software. However, due to increasing popularity of vendor-based courses as components of undergraduate curricula, students now have the opportunity to study a more practical approach and hence program networking devices (switches, routers). Furthermore, many students are studying networking and internetworking from a non-computing science and even a non-technical background. Such students need to comprehend a large number of new concepts within a short time span. In order to find potential solutions to these problems the authors have investigated modern educational theory regarding student understanding and conceptual changes. Diagrams can be employed as an aid to conceptual understanding. Thomas notes in respect to student difficulty in understanding concepts that: *"The use of diagrammatic representation provides an alternative to just offering more words which may only compound their difficulties*" [1]

Concepts and constructivism

Constructivism is a dominant theory in education. According to this theory students construct new ideas based upon their current/past knowledge. The students select and transform information, and make decisions that are dependent upon their present schema or mental models. It is important that educators are aware of student's mental models. According to von Glasersfeld:

"*Because there is no way of transferring meaning, i.e. concepts and conceptual structures, from one students head to another, teachers, who have the goal of changing something in students heads must have some notion of what goes on in these heads. Hence it would be seem necessary for a teacher to build up a model of the students conceptual world*" [2].

A student's mental model may or may not be correct and may require reorganisation to accord to accepted current practices and theory.

Changing a student's mental model requires conceptual reorganization, also known as conceptual change [3]. Before a conceptual change can take place, the naïve concepts that student possess need to be trapped. The process of making an existing mental model explicit is a crucial precursor to the cognitive restructuring process [4]. The authors are interested in mental models that map the structure of conceptions and forms of thinking that are used to describe how networking devices operate. Students studying internetworking are dealing with abstract and difficult topics that are not always clear. Setting the stage for conceptual change and learning by modelling is in itself a meaningful activity for gaining and refining understanding in complex scientific domains [5]. Conceptual change through building models have been advanced as tools to help student learning in science [6], because educators have recognized that model-based reasoning can facilitate the development of mathematical-scientific understanding of the natural world [7].

An important aspect of conceptual reorganization is abstraction. ACM states: "*Enduring computing concepts include ideas that transcend any specific vendor, package or skill set... While skills are fleeting, fundamental concepts are enduring and provide long lasting benefits to students, critically important in a rapidly changing discipline*". [8] Abstraction has been a common theme in the context of aiding conceptual understanding [9,10]. It should be noted that different subjects have their own appropriate models at differing levels of abstraction dependant upon the area of use. Norman states that:

"A good representation captures the essential elements of the event, deliberately leaving out the rest. The critical trick is to get the abstractions right, to represent the important aspects and the unimportant. This allows everyone to concentrate upon the essentials without distraction from irrelevancies. Herein lie both the power and the weakness of representation: Get the relevant aspects rights, enhance people's ability to reason and think; get them wrong, and the representation is misleading, causing

people to ignore critical aspects of the event or perhaps form misguided conclusions" [11].

Despite the advantage of abstraction the authors were not able to identify any conceptual models of routers and switches used in networking education.

Accordingly a conceptual model of networking devices is needed that can be used as the pedagogical framework for teaching internetworking technologies to both introductory and advanced students. In addition a physical model is also required to describe the components, both hardware and software, that will be deployed into the target environment Physical models include all forms that have an external manifestation [12]. Physical models can assist students understanding of how networking devices may be connected via physical models, whilst conceptual models assist student understanding of the principles employed to enable devices to communicate with each other. A good model incorporates both approaches.

A Switch and Router Model

Two models have been developed – one for a switch and one for a router. Each model is based on a simple physical model thereby supporting initial student learning. However both models allow for the incorporation of progressively advanced conceptual features without the need for new models. Experience teaching network technology clearly indicates it is important to control complexity. Key concepts must be introduced in a controlled manner. The authors are developing such a model which has the requirements that it must not only be technically correct but also valid for different levels of complexity thereby supporting not only introductory but also more advanced concepts. This model also hides the complexity via the use of abstraction.

A Switch Model

In the first instance a switch is represented as a simple box with ports (interfaces). Each physical port is represented on the switch model (e.g. Fastethernet 0/1 or Fa0/1). At the simplest level connectivity can be represented by internal connections between the ports within the switch. At a more complex level switches perform three main tasks: address learning; address forwarding and filtering; loop avoidance. A simple table can be incorporated into this diagram to teach how a switch learns and hence maps physical MAC addresses to ports i.e. address learning. This table can be used to teach how a switch establishes one to one connectivity (micro-segmentation) and hence performs address forwarding and filtering.

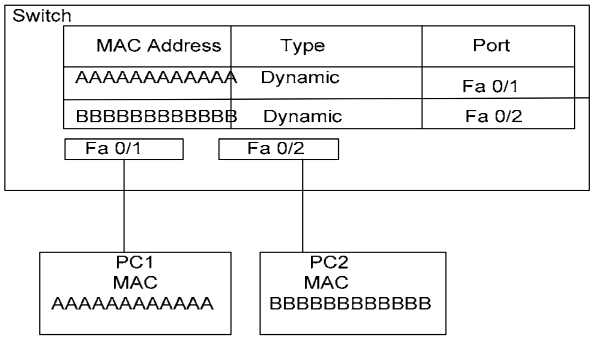


Figure 1: Switch Model

Similar a router model was also developed to represent the working of router and routing principles.

A Router Model

A PC is modeled as a device with a logical (IP) to physical (MAC address) Address Resolution Protocol (ARP) table and a Network Interface Card (NIC) table (IP address, Subnet mask and MAC address). Using this simple model it is possible to demonstrate simple PC to PC connectivity and the principles of ARP using a cross over cable. This exercise can further be enhanced by asking students to complete ARP (request) and ARP (reply) frames with the correct fields. The PC command "IPCONFIG" output directly maps onto these diagrams.

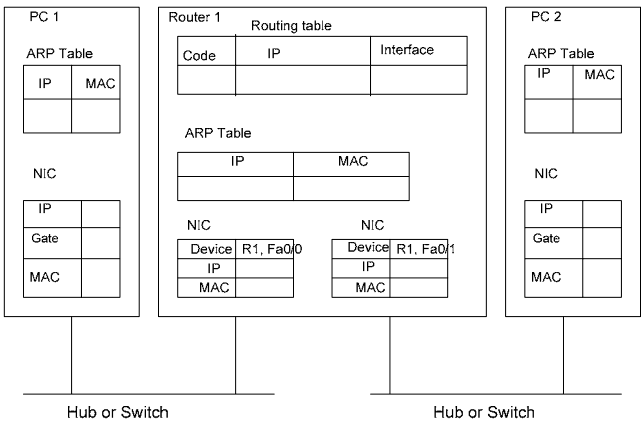


Figure 2: Router Model

A router is then modeled using the ARP and NIC table (as found in the PC) plus a routing table. Hence an incremental learning path is provided. The router commands "show arp" and "show ip route" can be used to in conjunction with the diagrams to show the state changes as networks are connected together. Again these diagrams form an integral part of the lectures. Workshop exercises require students to work together on different networks and complete these diagrams using the output from router commands and "IPCONFIG" on the PCs. Again it is possible to enhance this exercise by requiring students to complete the fields for an IP packet as it traverses the different devices and hence observe how, due to encapsulation and de capsulation, only the frame addresses change.

Analysis and Results

Three separate groups of students participated in this study. One of the group were studying the CISCO Network Academy Program (CNAP) which includes both CISCO
Certified Network associate (CCNA) [13, 14] and CISCO certified Network Professional (CCNP) [15,16,17] at university. A standard university unit consists of 48 hours of staff contact time per 12 week semester. The CCNA curriculum represents the equivalent of two standard university units i.e. a total of 96 hours. Students must successfully complete the CCNA prior to enrolling on the CCNP course. The CCNP course consists of the equivalent of four university units i.e. a total of 192 hours. The second group was studying the CCNA at a local college. At this college the CCNA is taught as four modules each representing 40 hours of tuition i.e. a total of 160 hours. The local college awards Diplomas and Advance Diplomas but not Bachelor awards. The third group were enrolled on Masters Courses which were specifically designed for graduates with a non-IT undergraduate qualification i.e. conversion masters. For these postgraduate students there are two units (106 and 206) as a prerequisite chain. The first unit (106) is an introductory unit to computer and network technology, the second postgraduate unit (206) is dedicated to network technology. Half of 106 unit (24 hours) is allocated to computer technology and the other half (24 hours) is dedicated to network technology. The second postgraduate unit (206) is dedicated entirely to network technology.

Completion of both these units represents therefore a total of 72 hours of teaching in network technology. The vendor based curriculum students (CCNA and CCNP) were taught in the normal prescribed manner. The units for the postgraduate students were based on the models developed for switches and routers. Using these models the postgraduate students were taught the principles of switch and router operation. The models were used both in the lectures and in the workshops. All students were given a questionnaire designed to obtain their understanding of router operation. Hence in addition to standard questions such as, ‘Briefly describe a router’, students were given several details of different router configurations that they had to explain.

The CCNA students at both the local college and university showed a lack of understanding of router operation because of misconceptions as shown in (table 1). Such misconceptions are trapped by comparing the description given by an expert in that field to that given by a student. The CCNP students at both the local college and university performed only marginally better than all the CCNA students (table 1). On the other hand postgraduate students clearly showed a better understanding of router operation. They made fewer misconceptions and were able to correctly use a far wider range of technical terms (table 1). The postgraduate students arguably may have better study skills but had only completed 72 hours of network education. The CCNA students are exposed to 96 hours of staff contact time. It should also be noted at the beginning of the units, all students failed to demonstrate their mental model although they must have possessed such a mental model. Postgraduate students, whose learning was based upon the state models, demonstrated a comprehension of devices which more closely matched those of an expert when compared to students studying Cisco Network Academy Program (CNAP). Postgraduate students are arguably more mature and are likely to have better study skills than undergraduate students. However, one group of postgraduate students had completed only 24 hours of instruction in contrast to CCNP students who had successfully completed the CCNA (96 hours of instruction) and an additional semester of CCNP material (at least 96 hours of instruction).

CCNA University (30)	CCNA TAFE (18)	CCNP University (9)	106 University (30)	206 University (18)
Optimiser (3)	Connects Networks (9)	Connects Networks (2)	Connects Networks (3)	**Connects Network (15)**
Road Map (2)	Layer 3 (4)	Layer 3 (2)	Layer 3 (9)	**Layer 3 (7)**
Connects Networks (3)	Best Path (4)	Broadcast Domain (2)	Best Path (3)	**Best Path (1)**
Layer 3 (5)	Direct Traffic (4)	Control flow (1)	Direct traffic (2)	**Direct traffic (2)**
Best Path (1)	No idea (10)	Collision domain (1)	Broadcast domain (5)	**Routing table (2)**
Direct traffic (4)		Make decision on IP (1)	Control flow (1)	**Make decision on IP address (1)**
No idea (4)		No idea (3)	Routing table (1)	**Bridge gap (1)**
			Packet switching (1)	**Network equipment (1)**
			Collision domain (3)	**Communicate between VLAN (1)**
			Make decision on IP address (2)	**Connects switch (3)**
			Network Equipment (1)	**ARP table (2)**
			Connects switch (1)	
			Arp table (2)	

Table 1: Student Ideas of Router Operation

Conclusions

Constructivism is the dominant theory of conceptual understanding in modern educational theory. Understanding of students' conceptual models is of vital importance to educators because students use these models to facilitate their understanding. The authors have developed a state model as an aid to understanding the operation of routers and switches. Postgraduate students who were taught using these models demonstrated conceptual understanding of router and switch operation that was more in accordance with that of an expert in the field, than did the CNAP students who were not taught using this model. Further work is currently being undertaken to extend the scope of these models. Results to date clearly indicate that these models have had a significant impact upon student learning. The authors are developing this study further to obtain a better understanding of student's mental models.

Bibliography

1. Thomas, R. *Fewer Words, clearer picture: The value of a diagrammatic representation of ideas*. in *Third Biennial Communication skills in University education (CSUE)*. Fremantle, WA, Australia: Teaching and Learning Centre, Murdoch University, Murdoch, WA, Australia. (2000).
2. von Glasersfeld, E. and L.P. Steffe, *Conceptual models in educational research and practice.* Journal of Educational Thought, **25**(2): 91 - 103. (1991).
3. Chi, M.T.H. and R.D. Roscoe, *The processes and challenges of conceptual change.* In M. Limon & L. Mason (Eds.), Reconsidering conceptual change: Issues in theory and practice, 3-27. (2002)
4. Wichmann, A., J. Gottdenker, and D. Jonassen. *Developing a Framework for Conceptual Change within Scientific Inquiry*. In T*he 3rd IEEE International Conference on Advanced Learning Technologies (ICALT'03)*. (2003).
5. Milrad, M., M. Spector, and P. Davidsen, *Model Facilitated Learning*, in *Learning and Teaching with Technology: Principles and Practices*. Kogan Page Publishers: London, UK. (2002)
6. Edelson, D., D. Gordin, and R. Pea, *Addressing the challenges of inquiry-based learning through technology and curriculum design.* The Journal of the Learning Sciences, **8**(3&4): 391-450. (1999).
7. Lehrer, R., J. Horwath, and L. Schuable, *Developing model-based reasoning.* Interactive Learning Environments, **4**(3): 219-231. (1994).
8. ACM. *Computing curricula 2001 ACM/IEEE Joint Task Force Computer Science Final Report*. 2001: ACM Press. (2001)
9. Ramsden, P., *Learning to Teach in Higher Education*. 1992, London: Routledge. (1992).
10. Staggers, N. and A. Forrcio, *Mental models: Concepts for human-computer interaction research.* International Journal of Man-Machine Studies, **38**(4): 587-605. (1993).
11. Norman, D. R., *Things That Make Us Smart*. Peruses Books. (1993).
12. Sparks, G., *An introduction to modelling software systems using the Unified Modelling Language.* 2001.
13. Cisco, *Cisco Networking Academy Program: First Year Companion Guide*. (2 ed.). 2001, Indianapolis, IN USA: Cisco Press. 991. (2001).
14. Cisco, *Cisco Networking Academy Program: Second Year Companion Guide*. (2 ed.). 2002, Indianapolis, IN: Cisco Press. 718. (2002).
15. McGregor, M. and V. Amato, *CCNP Cisco Networking Academy Program: Semester Five Companion Guide Advanced Routing*. 2001, Indianapolis, IN: Cisco Press. (2001)
16. McGregor, M., *CCNP Cisco Networking Academy Program: Remote Access Companion Guide*. 2002, Indianapolis IN: Cisco Press. 492. (2002).
17. Lewis, W., *CCNP Cisco Networking Academy Program: Multilayer Switching Guide*. 2003, Indianapolis, IN: Cisco Press. 636. (2003).

5.8 A New state Model for Internetworks Technology

Title of the paper : A new state model for Internetworks technology

Authors: S P Maj, G Kohli

Journal: Issues in Information Science and Information Technology

Conference Location: Rockhampton, Australia

Conference date: 25th–27th June, 2004

The journal solicits the fines submissions in areas that explore issues in effectively and efficiently informing clients through information technology. As part of the journal, contributions for a special section on IT education were invited. This paper was among the top three papers presented.

There are wide ranges of equally valid approaches to teaching networking. One approach is to teach internetworking technologies (switches, routers). Extensive analysis of educational materials in this area had indicated that these devices are typically treated as black boxes. In order to provide good conceptual models of switches and routers two models were proposed initially. This paper provides an insight into how the switch models can explain the basic functions of switches, which are:

- Address learning
- Address forwarding and filtering
- Loop avoidance

Students, whose learning was based upon the state models, demonstrated a comprehension of devices comparable to a qualified and experienced expert in the field. Furthermore, work to date on switch models shows that these models can be used to capture all the necessary state information of these devices.

A New State Model for Internetworks Technology

S.P.Maj & G.Kohli
Edith Cowan University, Perth, Australia
p.maj@ecu.edu.au g.kohli@ecu.edu.au

Abstract

There are a wide range of equally valid approaches to teaching networking. One approach is to teach internetworking technologies (switches, routers). However, an extensive analysis of educational materials in this area has indicated that these devices are typically treated as 'black boxes'. This is contrary to educational theory that supports the need for a conceptual model. Two state models were designed and used as the pedagogical foundation of network curriculum. These models are valid for different levels of technical complexity and work to date strongly suggests they support student learning. Based on these results the models have been further developed.

Keywords: Abstraction, Conceptual Models, Finite State Machine and State Models

Introduction

The ACM/IEEE Computing Curriculum 2001 included Net-Centric Computing in the Computer Science Undergraduate Body of Knowledge (IEEE/ACM, 2001). There are however a wide range of equally valid approaches to teaching network curriculum ranging from quantitative (engineering) to software/algorithmic (computer science) (J. Kurose et al., 2002). Both within Australia and internationally there is a demand for a practical 'hands on' approach to networking curriculum. Accordingly some universities have adopted the Cisco Network Academy Program (CNAP) and hence obtain access not only to vendor specific curriculum and certification (Cisco Certified Networking Associate (CCNA) and Cisco Certified Networking Professional (CCNP)) but also low cost equipment (hubs, switches and routers). It should be noted that the CCNP is based upon an educational web site that cost US$25 million to develop and an extensive repertoire of textbooks. Both a typical university curriculum in networking and the vendor specific networking curriculum (CCNA and CCNP) were analyzed. Both curricula teach networking fundamentals however the Cisco curriculum also provides an in-depth 'hands on' approach to switch and router configurations. However in both cases the internetworking devices (switches and routers) are considered as 'black boxes'. This is contrary to Constructivism, the dominant educational theory, in which students construct knowledge rather than merely receive and store knowledge transmitted by the teacher (M. Ben-Ari, 2001). von Glasersfeld states, "*… knowledge cannot simply be transferred by means of words. Verbally explaining a problem does not lead to understanding, unless the concepts the listener has associated with the linguistic components of the explanation are compatible with those the explainer has in mind. Hence it is essential that the teacher have an adequate model of the conceptual network within which the student assimilates what he or she is being told. Without such a model as a basis, teaching is likely to remain a hit-or-miss affair.*"
(von Glasersfeld, 1989). A conceptual model of a router and a switch is therefore needed. This model must not only be technically correct but also valid for different

levels of complexity thereby supporting not only introductory but also more advanced concepts.

State Models

Models are a means of controlling detail and communication. Desirable model characteristics include: diagrammatic, self-documenting, easy of use and hierarchical top-down decomposition to control detail. Levelling is the property in which complex systems can be progressively decomposed to provide completeness. According to Cooling there are two main types of diagram: high level and low level (Cooling, 1991). High level diagrams show the overall system structure with its major sub-units. By contrast, low level diagrams are solution oriented and must be able to handle considerable detail. Some systems may be modeled using state diagrams. According to the National Institute of Science and Technology,

> 'A finite state machine is a model of computation consisting of a set of states, a start state, an input alphabet and a transition function that maps input symbols and current states to the next state. Computation begins in the start state with an input string. It changes to new states depending on the transition function.' (Unknown)

At any given moment in time the system exists in a certain state. The set of all states is the state space. Significantly the state diagrams should show only relevant details. Two simple state models have been developed – one for a switch and one for a router. However unlike typical state models these new models allow the introduction of progressively advanced conceptual features hence supporting student learning. According to Von Glasersfeld: "Because there is no way of transferring meaning, i.e. concepts and conceptual structures, from one students head to another, teachers, who have the goal of changing something in students heads must have some notion of what goes on in these heads. Hence it would be seem necessary for a teacher to build up a model of the students conceptual world" (E. von Glasersfeld & Steffe, 1991)

Switch – Simple Model

In the first instance a switch is represented as a simple box with ports/interfaces. Each physical port is represented on the switch model (e.g. Fastethernet 0/1 or Fa0/1). At the simplest level connectivity can be represented by internal connections between the ports within the switch. This simple model does not capture states and hence it is not a state model. At a more complex level switches perform three main tasks: address learning; address forwarding and filtering; loop avoidance. These tasks are associated with state changes within a switch - hence the following state model.

Switch State model - Address learning

The minimum relevant switch states for address learning are: MAC address, MAC address type and port identification. A simple table can be incorporated into the simple switch diagram to capture this information (figure 1) hence establishing a simple state model. In the initial state (S0) this table is empty. Obviously the connecting PCs must be represented as simple state diagrams with their MAC addresses. PC state information can be derived from the command line '*ipconfig*'.

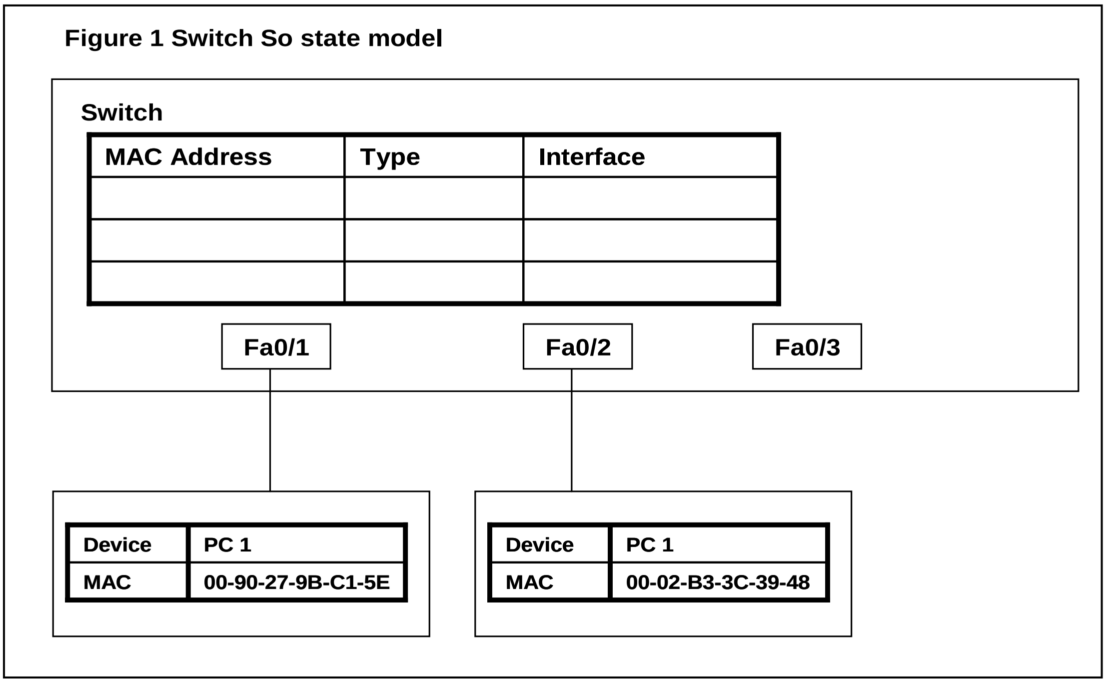


Figure 1: Switch S0 state model

When a PC attempts to send a data frame to another PC the switch learns the MAC address of the transmitting PC and hence enters the next state S1.

Switch State Model - Address forwarding/filtering

The process of address learning continues until the switch learns the MAC addresses of all the connected PCs (figure 2). This information can be derived from the switch command '*Switch#show mac-address-table*'. In this state the switch can forward and filter data frames.

Switch#show mac-address-table

Mac address	*Type*	*Interface*
00-90-27-9B-C1-5E	*dynamic*	*fa0/1*
00-02-B3-3C-39-48	*dynamic*	*fa0/2*

Figure 2 Switch – Sn state model

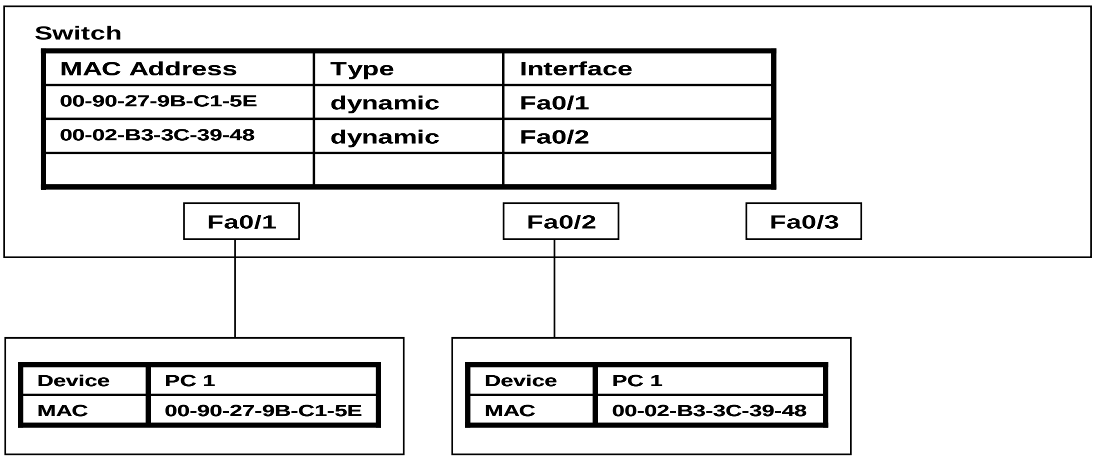


Figure 2 Switch – Sn state model

The state diagrams may be extended to include Virtual LAN (VLAN) state information by including in the switch table a VLAN column (figure 3). Again this state data may be derived from the switch configuration command, '***switch#show vlan'***,

Switch#show mac-address-table

VLAN	***Mac address***	***Type***	***Interface***
1	***00-90-27-9B-C1-5E***	***dynamic***	***fa0/1***
1	***00-02-B3-3C-39-48***	***dynamic***	***fa0/2***

Figure 3 Switch – VLAN state model

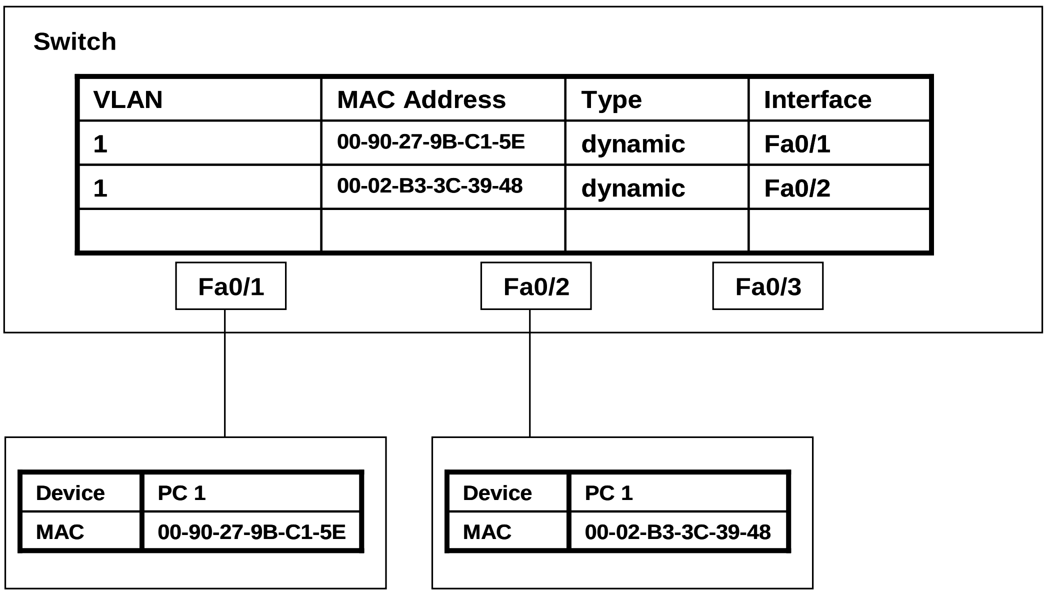


Figure 3 Switch – VLAN state model

Switch State Model - Loop Avoidance

The third function of a switch is loop avoidance i.e. Spanning Tree Protocol (STP). The switch state model may be extended further to capture STP information by including a second switch table incorporating Bridge Identification based on Priority and MAC address (Lewis, 2003). Furthermore port status (Designated, Root, Blocking) may be added to the ports (figure 4).

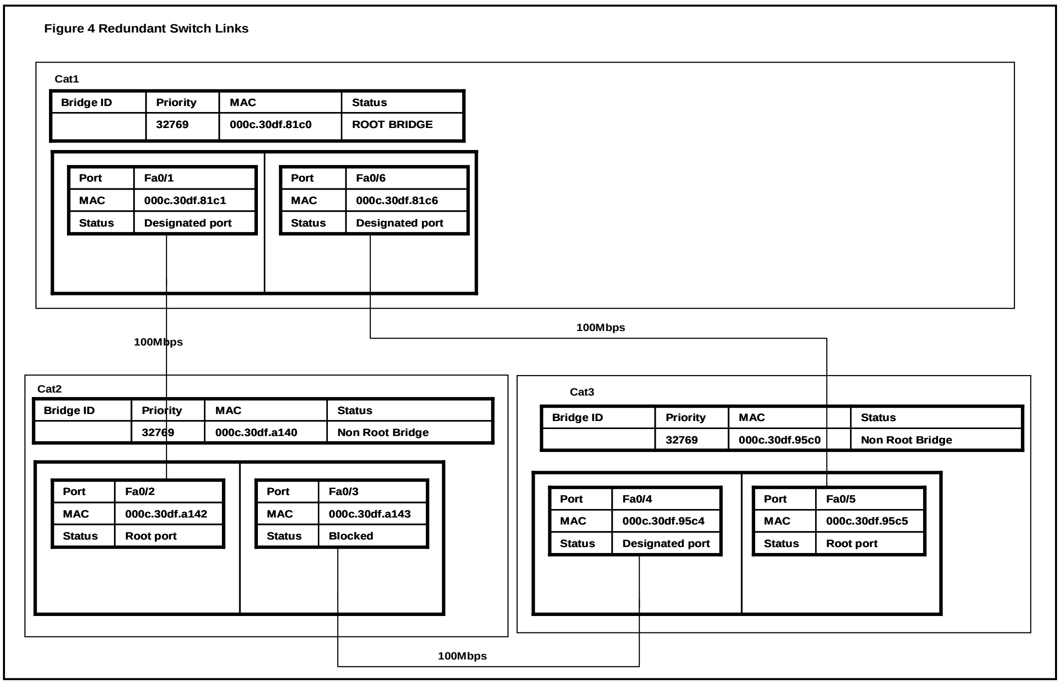


Figure 4: Redundant Switch Links

Router Model

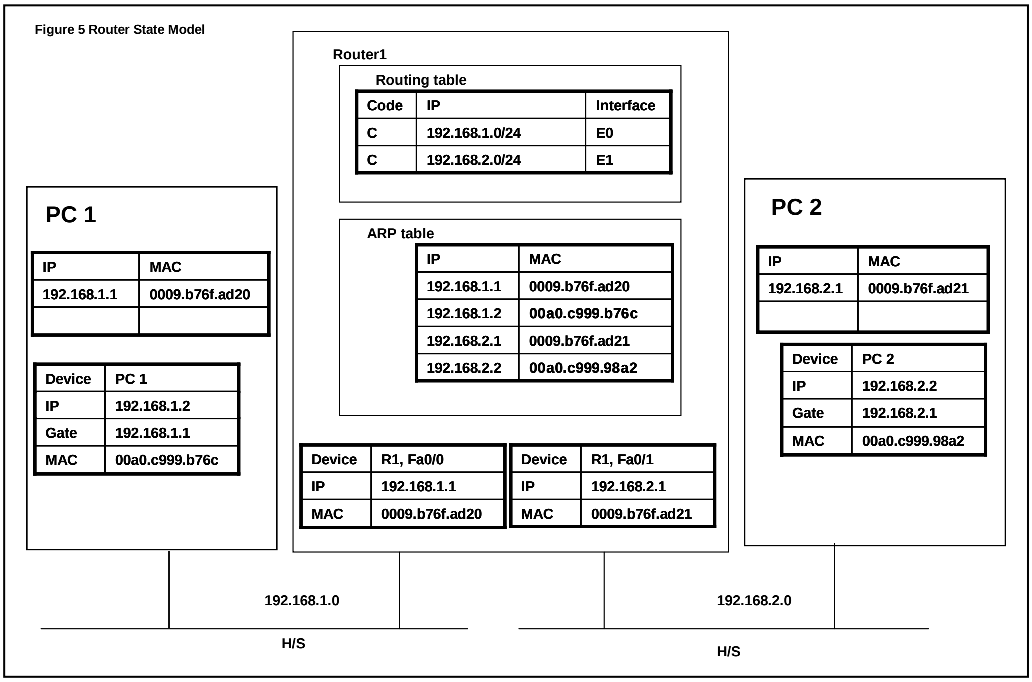


Figure 5: Router State Model

A PC can be modeled as a simple state device with a logical (IP) to physical (MAC address) Address Resolution Protocol (ARP) table and a Network Interface Card (NIC) table (IP address, Subnet mask and MAC address). The PC command “**IPCONFIG**” output directly maps onto these simple PC state diagrams. A router can then be modeled as a state diagram using the ARP and NIC table (as found in the PC) plus a routing table (figure 5). Hence an incremental learning path is provided. The router commands "**show arp**" and "**show ip route**" can be used to in conjunction with the diagrams to show the state changes as networks are connected together.

Evaluation

The switch and router models were evaluated in a teaching environment. Two groups were selected as part of the experiment design. The first group was doing an undergraduate course in networking. At the undergraduate level it is possible to study vendor specific awards (CCNA and CCNP) or more generic network units. A standard university unit consists of a 2 hour lecture and 2 hours of workshop time (fully supervised) every week for a 12 week semester. The CNAP mandates student contact time hence the CCNA curriculum represents the equivalent of two standard university units i.e. a total of 96 hours. Students must successfully complete the CCNA prior to enrolling on the CCNP course. The CCNP course consists of the equivalent of four university units each of which has 48 hours of staff contact time per 12 week semester.

This university offers a number of postgraduate awards in Information Technology. There are Master Courses specifically designed for graduates with a non-IT undergraduate qualification i.e. conversion masters. For these students there are two units (106 and 206) as a prerequisite chain. The first unit (106) is an introductory unit to computer and network technology. Half of this unit (24 hours) is allocated to computer technology and the other half (24 hours) is dedicated to network technology. It should be noted that there are no prerequisites to this unit. The second postgraduate unit (206) is dedicated entirely to network technology. Each postgraduate unit is a standard university unit (48 hours of staff contact time per 12 week semester). The vendor based curriculum students were taught in the normal Cisco prescribed manner using online Cisco material, practical lab exercises (hand on) and case studies presented by Cisco curriculum. The units for the postgraduate students were based on the models developed for switches and routers (Figures 1, 2, 3, 4 and 5).

Students on all the above courses (undergraduate vendor based CCNA/CCNP and postgraduate IT) completed a questionnaire which was designed to determine their understanding of router operation. This questionnaire was distributed at the end of the semester. Questions included both simple definitions (e.g. What is a router?) and questions to determine the depth of understanding of router operation. In particular students were provided with router and PC configurations diagram for a given network and asked to explain its operation. In addition to this the postgraduate students participated in an open forum during which all discussions were recorded. Furthermore the questionnaire was also given to a qualified and experienced network expert who was not involved teaching the postgraduate curriculum.
The responses for all student groups and the expert were analyzed. The expert clearly demonstrated the use of a wide range of technical vocabulary and a clear understanding of router operation (Table 1).

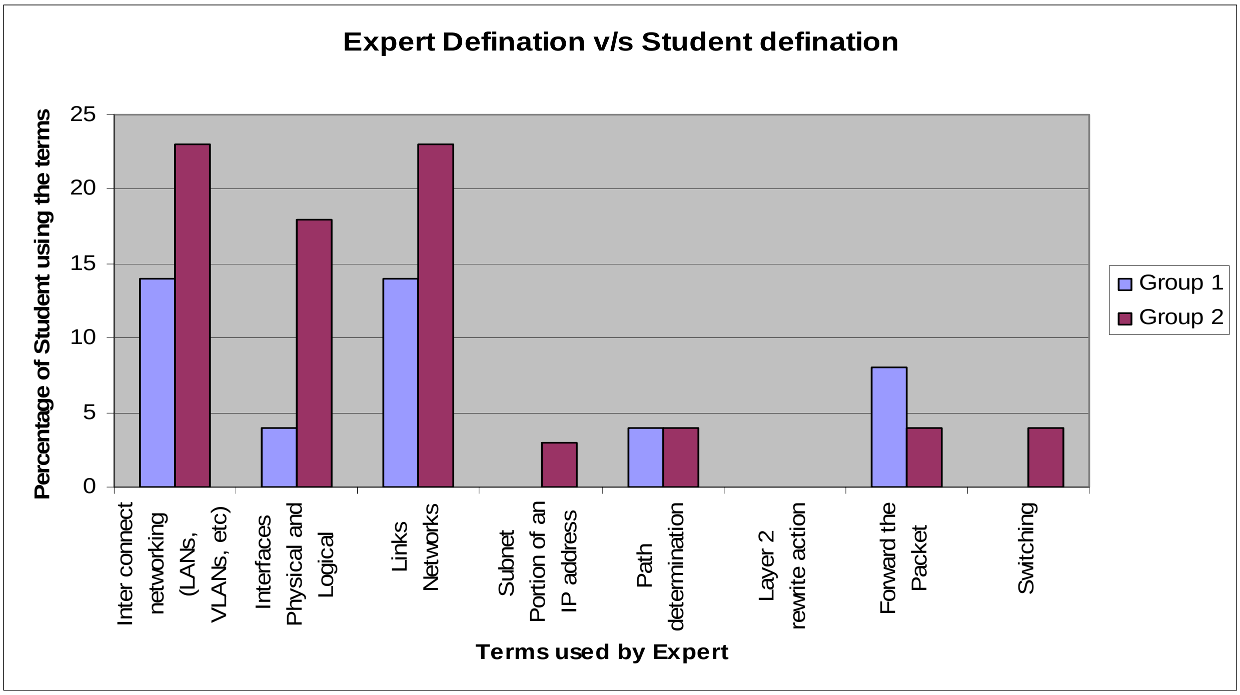
Experts Terms used
Inter connects networking (LANs, VLANs, etc)
Interfaces (Physical and Logical)
Links Networks
Subnet Portion of an IP address
Path determination
Layer 2 rewrite action
Forward the Packet
Switching

Table 1: Vocabulary of terms used by an expert

Group 1 which was doing the Vendor specific course provided standard text book based definitions. However, they demonstrated a lack of depth of understanding of device operation (Graph 1).

Group 2 was studying on the postgraduate university unit 106 and 206 which were taught using these models also provided standard text book based definitions. However they clearly demonstrated a far better understanding of router operation. They had a far more extensive vocabulary of technical terms all of which were in conjunction with expert definition as shown in the Graph 1. The students on the unit 206 performed comparably to those on unit 106.

From the analysis of terms used by the expert and the two groups, it is highly significant that the percentage of students using terms that the expert used was in most cases much higher in group 2 then in group 1 who were taught using the model.



Graph 1: Expert term versus student definitions

Further work

Given the success of these models as an aid to learning they were further developed. Additional port state information includes: Port Number, priority and cost (figure 6). Again the state diagram model corresponds to switch command line output. This state model can be used to show all state transitions occurring during STP operation i.e.

S0 Initial state
S1 Root Bridge elected (i.e. Bridge ID status)
S2 One Root port per Non Root Bridge elected
S3 One Designated Port per segment elected
S4 Non-root and non-designated ports blocked i.e. operational

Furthermore, using these diagrams it is possible also to capture port state transitions: Disabled, Blocking, Listening, Learning and Forwarding.

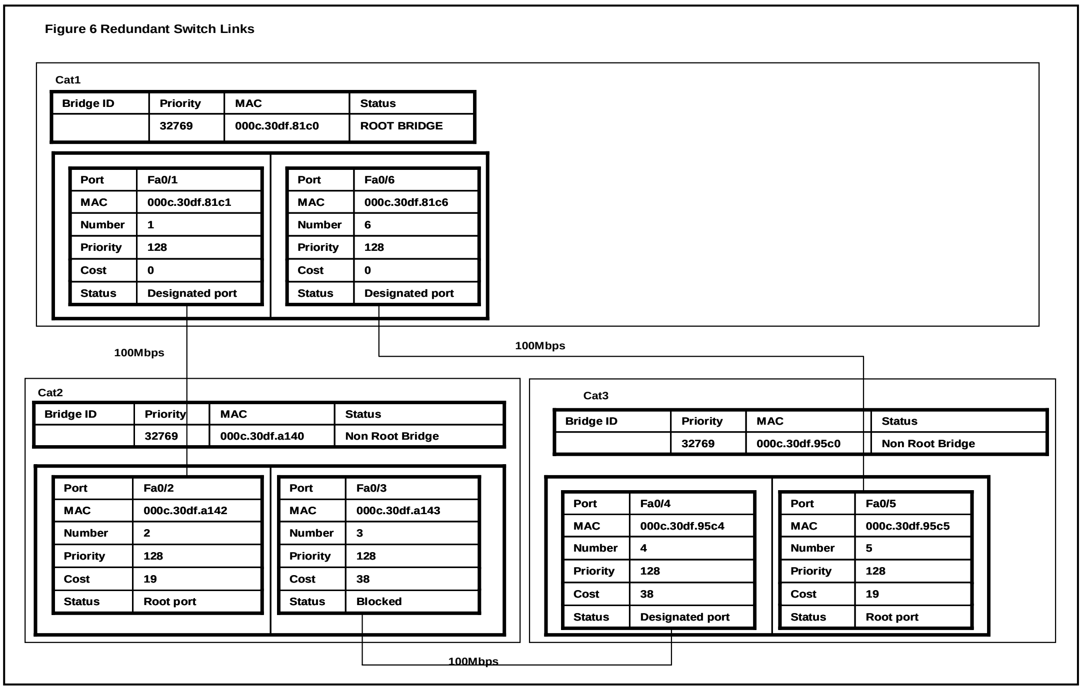


Figure 6: Redundant Switch Links

Conclusion

Postgraduate students, whose learning was based upon the state models, demonstrated a comprehension of devices comparable to a qualified and experienced expert in this field. Furthermore these students performed significantly better than other students. Postgraduate students are arguably more mature and are likely to have better study skills than undergraduate students. However one group of postgraduate students had completed only 24 hours of instruction in contrast to CCNP students who had successfully completed the CCNA (96 hours of instruction) and an additional semester of CCNP material (at least 96 hours of instruction). Furthermore the CNAP mandates continuous on-line assessment of CCNA and CCNP students. Further work is currently being undertaken extend the scope of this work but the results to date clearly indicate the diagrams have a significant impact on student learning. Based on this success the state models were further developed. Work to date suggests these models can be used to capture all relevant state information.

References

Ben-Ari, M. (2001). Constructivism in Computer Science Education. *Journal of Comuters in Mathematics and Science Teaching, 20(1)*, 45-73.

Cooling, J. E. (1991). *Software Design for Real-Time Systems*. Padstow, Cornwall: Chapman and Hall.

IEEE/ACM. (2001). *CC-2001 Task Force, Computing Curriculum (Final Draft),* from http://www.computer.org/education/cc2001/final

Kurose, J., Liebeherr, J., Ostermann, S., & Ott-Boisseau, T. (2002). *ACM SIGCOMM Workshop on Computer Networking: Curriculum Designs and Education Challenges,* from http://www-net.cs.umass.edu/sigcomm/education/education_workshop_report_final.pdf

Lewis, W. (2003). *CCNP Cisco Networking Academy Program: Multilayer Switching Guide*. Indianapolis IN: Cisco Press.

Unknown.*National Institute of Science and Technology*, from http://www.nist.gov/dads/html/finitestatemachine.html

von Glasersfeld, E. (1989). Cognition, Construction of Knowledge, and Teaching. *Synthese*(80), 121-140.

von Glasersfeld, E., & Steffe, L. P. (1991). Conceptual models in educational research and practice. *Journal of Educational Thought, 25*(2), 91 - 103.

Biographies

S. PAUL MAJ is A/Prof at the School of Computer and Information Science, Edith Cowan University, Perth, Australia, and also Adjunct Professor at the Department of Information Systems and Operations Management, University of North Carolina (Greensboro) in the USA. He is an internationally recognized authority in laboratory automation and has published a commissioned book in this field.

GURPREET KOHLI is a PhD student at Edith Cowan University with two years of experience in Lecturing and Developing Network and Data Communication units at Edith Cowan University. Gurpreet is currently looking into modelling Network as part of his research at Edith Cowan University.

5.9 State Models for Internetworking Technologies

Title of the paper : State Models for Internetworking Technologies

Authors: S P Maj, G Murphy, G Kohli

Conference: 34th ASEE / IEEE Frontiers in education conference

Conference Location: USA

Conference date: 20th-23rd October, 2004

This paper was presented at 34th ASEE/IEEE Frontiers in Education conference. This is an annual conference is held in the USA. The primary aim of this conference is to publish research, which highlights good teaching methods in engineering education. In total, 700 full papers and research in progress papers were received. Out of 700 only 310 full paper and 100 research in progress papers were accepted at this conference. The overall acceptance rate was around 57 %. The IEEE is a premier engineering body in United States and worldwide.

New, diagrammatic state models of a switch and router were designed. The router state model can be used for all the main Interior Gateway Protocols – distance vector (RIP, IGRP), link state (OSPF) and balanced hybrid. (EIGRP). Preliminary work indicates that it also supports an Exterior Gateway Protocol (BGP). The switch state model can be used for both basic and advanced switch operation (STP, VTP and VLANs). Both router and switch models employ modularity hence both these models provide a basic model whose functionality can be enhanced by the inclusion of additional state tables. Furthermore, these models are diagrammatic, self-documenting and easy to use. They employ leveling and hence provide hierarchical top-down decomposition thereby controlling technical detail. These models were used as the pedagogical foundation of network curriculum and the results evaluated. Importantly, the number of hours students had spent learning the topic was taken into account. It was found that even with only 20 and 40 hours of teaching with state models compared with 80 and 100 hours by CNAP program that these models significantly improved student learning.

STATE MODELS FOR INTERNETWORKING TECHNOLOGIES

S P Maj, G Murphy, G Kohli

Abstract

New, diagrammatic state models of a switch and router were designed. The router state model can be used for all the main Interior Gateway Protocols – distance vector (RIP, IGRP), link state (OSPF) and balanced hybrid. (EIGRP). Preliminary work indicates that it also supports an Exterior Gateway Protocol (BGP). The switch state model can be used for both basic and advanced switch operation (STP, VTP and VLANs). Both router and switch models employ modularity hence both these models provide a basic model whose functionality can be enhanced by the inclusion of additional state tables. Furthermore these models are diagrammatic, self-documenting and easy to use. They employ leveling and hence provide hierarchical top-down decomposition thereby controlling technical detail. These models were used as the pedagogical foundation of network curriculum and the results evaluated. It was found that these models very significantly improved student learning.

Index Terms – Modelling, Routers, Switches, State Machine.

INTRODUCTION

There are wide ranges of equally valid approaches to teaching networking. According to Kurose [1], '*Among the approaches towards networking curricula, one finds the more quantitative (electrical engineering) style of teaching networking versus a more software/algorithmic (computer science) approach, the more "hands-on" laboratory based approach versus the more traditional in-class lecture based approach; the bottom-up approach towards the subject matter versus a top-down approach.*' Based on a market analysis of employer expectations this university elected to implement a vendor-based, "hands-on" approach to curriculum [2]. It should be noted that vendor-based education has both strengths and weaknesses; critics and advocates [3].

Accordingly the Cisco Network Academy Program (CNAP) was introduced. The CNAP provides not only vendor specific curriculum and certification (Cisco Certified Networking Associate (CCNA) and Cisco Certified Networking Professional (CCNP)) but also low cost equipment (hubs, switches and routers). The CNAP is based upon an educational web site that cost US$25 million to develop and an extensive repertoire of Cisco sponsored textbooks.

Two main problems were found with the CNAP curriculum. Firstly, this curriculum primarily teaches internetworking device (routers and switches) functionality using the command line interface (CLI) in conjunction with various software diagnostic tools such as the Cisco Discovery Protocol, PING, TRACE, IP ROUTE and also Telnet. The CLI allows the user to determine and modify the status of the various components of a router such as routing table entries, Address Resolution Protocol (ARP) table entries, interface status etc. However, the hierarchical CLI commands may require considerable technical expertise. Furthermore the status information of the many different device tables, interfaces etc must typically be obtained by a number of different CLI commands. This may be problematic during the teaching process when it is necessary to integrate all of this information from a number of different CLI commands. It should be noted however that for experienced network engineers the CLI is a very powerful and useful tool.

Secondly, it was found that the CNAP on-line curriculum did not provide a coherent, diagrammatic, conceptual model of internetworking devices. Hence a large selection of Cisco sponsored textbooks and also non-Cisco textbooks were analyzed. It was found that internetworking devices are considered as 'black boxes' by introductory Cisco endorsed textbooks [4], [5], [6], [7] and introductory non Cisco endorsed textbooks [8], [9], [10], [11], [12]. The search was extended to Cisco endorsed switching books [13], [14], [15], [16] and non-Cisco endorsed switching books [17], [18]; Cisco endorsed textbooks on routing [19], [20], [21] and non Cisco books on Routing [22], [23], [24], [25]; remote access books [26], [27]l and finally both general [28], [29] and specialized books on networking [30] - all with the same result. It should be noted that Cisco use finite states to explain neighbor acquisition in BGP, route acquisition in EIGRP and also neighbor and route acquisition in OSPF. However this is not integrated with router management at the CLI interface.

There exist a number of network tools used for the management of large populations of devices. Such tools consolidate and automate common management tasks such as software distribution, change and audit, authorization, device inventory etc. Certainly the software management tools analyzed offered a comprehensive, modular approach to network management but do not provide an integrated, diagrammatic model of devices.

There exist a number of different modelling methods that have been used to teach networking that include: OPNET [31], CLICK [20], NsClick [32] etc. However none of those investigated integrates internetwork device status into a single hierarchical model.

Arguably what is needed is a simple, diagrammatic model that controls the technical complexity associated with the CLI and can be used as the pedagogical foundation of network technology curriculum. To assist in the development of such a model education theory was analyzed.

EDUCATIONAL THEORY

According to Constructivism, the dominant educational theory, students construct knowledge rather than merely receive and store knowledge transmitted by the teacher [33]. Von Glasersfeld states, *"… knowledge cannot simply be transferred by means of words. Verbally explaining a problem does not lead to understanding, unless the concepts the listener has associated with the linguistic components of the explanation are compatible with those the explainer has in mind. Hence it is essential that the teacher have an adequate model of the conceptual network within which the student assimilates what he or she is being told. Without such a model as a basis, teaching is likely to remain a hit-or-miss affair."* [34]. A conceptual model of a router and a switch should therefore allow the student to assimilate concepts. For such a model to be the pedagogical basis of networking curriculum it must also be valid for different levels of complexity thereby supporting not only introductory but also more advanced concepts. Different modelling methods were used to describe internetworking device operation and evaluated for their potential effectiveness as teaching tools.

MODELLING METHODS

There exist a wide range of modelling methods each with its own strengths and weakness. Modelling characteristics considered of particular importance in this application are: diagrammatic, ease of use, ability to control detail by means of hierarchical top-down decomposition and also the integration of the status of the different components of a router. According to Cooling [35], there are two main types of diagram: high level and low level. High-level diagrams are task oriented and show the overall system structure with its major sub-units. Such diagrams describe the overall function of the design and interactions between both the sub-systems and the environment. The main emphasis is 'what does the system do' and the resultant design is therefore task oriented. By contrast, low-level diagrams are solution oriented and must be able to handle considerable detail. With low-level diagrams the main emphasis, applicable to this application, is 'how does the system work'. Furthermore, according to Cooling, methods must be: *'Formal (that is, there are defined rules that must be obeyed); Visible (no implicit information); Expressive (easy to state what we mean); Understandable (making communication simpler) and Easy to Use (an obvious need, yet frequently ignored).'* Internetworking devices were modeled using a number modelling techniques and the results evaluated. The techniques evaluated included: object-oriented modelling [36], function-oriented modelling i.e. Data Flow Diagrams [37], Process Specification [38], Jackson System Development [39, 40], Specification and Description Language [41], Structured Analysis and Design [42] and Design Approach for Real-Time Systems [43]. Most of these methods could capture information flow, and device behavior but it was difficult to relate the model results to the data extracted from the CLI.

A state-oriented approach was also attempted. According to Peters [44], '*In a state-oriented approach to specifying the behavior of software, an entity is modeled as a state machine. The behavior of an entity is described in terms of the passage of the entity through different states. The state of a software system is defined in terms of the value of the available information about the system at a particular instant in time*.' Furthermore, '*State-oriented specifications include variables that change each time a state change occurs.*' The standard approach to this type of specification is the finite-state machine. A finite-state machine can be defined mathematically and also employs a graphical notation. The FSM is one technique used for modelling communication protocols. According to Halsall [45], '*The three most common methods for specifying a communication protocol are state transition diagrams, extended event-state tables and high-level structured programs*.' He further states, '*Irrespective of the specification method, we model a protocol as a finite state machine or automaton. This means that the protocol – or, more accurately, the protocol entity – can be in just one a finite number of defined states at any instant. For example, it might be idle waiting for a message to send or waiting to receive and acknowledgment.*' A communication protocol can therefore be modeled as a protocol entity that can exist in one of a number of defined states. Internetworking devices implement a number of communication protocols. Accordingly state modelling was used as the basis of modelling internetworking devices – switches and routers.

STATE VARIABLE MODELS – ROUTER and SWITCH

A router is an OSI layer 3 device. Accordingly it implements layer 3, layer 2 and layer 1 protocols. Interface status is typically determined by the CLI command '*Show interface*' on the Router Command Line Interface (Figure 1). This output may be simplified to consider only selected state information from this CLI output i.e. '*FastEthernet e0/1 is up, line protocol is up.*' The OSI physical layer line status, of for example the Ethernet interface (E0/1), is triggered by a carrier detect signal. Hence, the Layer 1 Physical connectivity state is either up or down. The line protocol on E0/1, triggered by keepalive frames, is an OSI data link layer function. The Layer 2 Data Link Layer connectivity state can also be considered either up or down. The other key Layer 2 information is the Media Access Control (MAC) address. Key Layer 3 information is the IP address and subnet mask of E0/1. All this can be represented on a single router diagram (Figure 2).

```
Device#show interface fastethernet 0/1
FastEthernet0/1 is up, line protocol is up
  Hardware is Fast Ethernet, address is 000c.30df.81c1 (bia 000c.30df.81c1)
  MTU 1500 bytes, BW 100000 Kbit, DLY 1000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Auto-duplex, Auto-speed
  input flow-control is off, output flow-control is off
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 23:41:43, output 23:41:30, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue :0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     536 packets input, 72501 bytes, 0 no buffer
     Received 276 broadcasts, 0 runts, 0 giants, 0 throttles
```

FIGURE 1 : CLI SHOW INTERFACE OUTPUT

Consider the layer 3 Address Resolution Protocol (ARP).

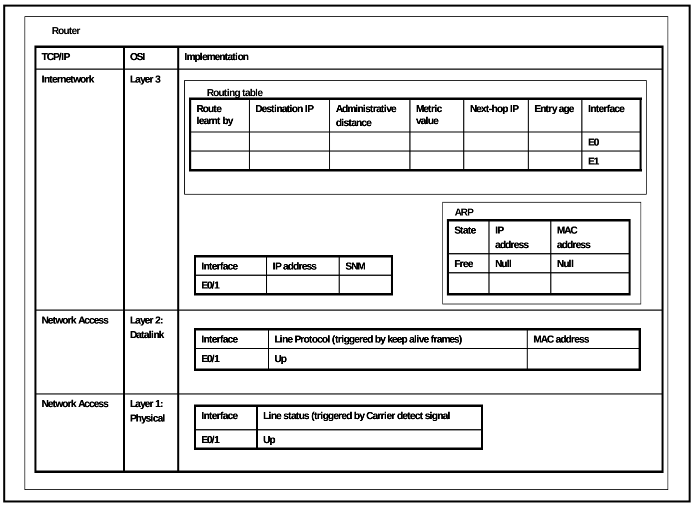


FIGURE 2: ROUTER STATE VARIABLE MODEL (RIP)

This protocol, or protocol entity, can exist in one of a three defined states – free, pending or resolved. In the free state the time-to-live entry has expired; the pending state means a request for an IP entry has been sent but no reply received and the resolved state means the physical (MAC) to logical (IP) mapping is complete. Each state has a state variable – Null, IP address and MAC address. All this information may be represented in an ARP State Variables Diagram (Figure 3). This diagram may be incorporated into a diagrammatic representation of a router (Figure 2). This information may be obtained by the CLI command 'show arp'. Similarly the state information for the routing protocol RIP may also be represented in a single table and incorporated the same diagram (Figure 2). For the sake of clarity the diagram only represent one interface (Fast Ethernet 0/1). It is possible to modify the diagrams to include multiple Ethernet interfaces and also interfaces employing different technologies such e.g. Serial point to point using HDLC.

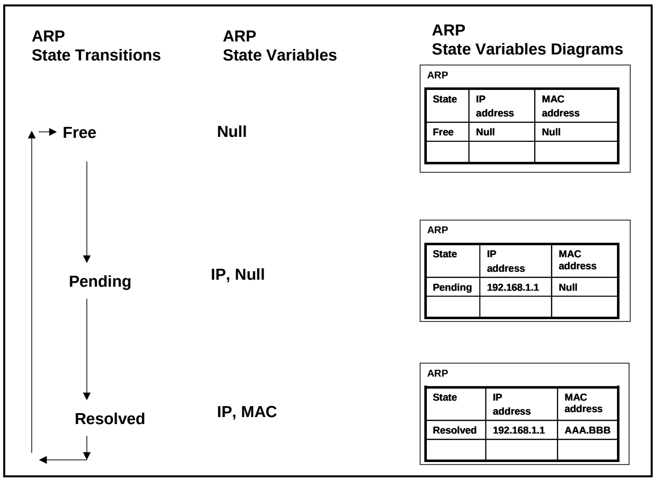


FIGURE 3: STATE TRANSITIONS, VARIABLES AND DIAGRAMS (ARP)

Similarly a state diagram has been designed for a switch (Figure 4). Using this single diagram it is possible to describe the state information and hence operation of both basic and advanced switch operation – Spanning Tree Protocol (STP), Virtual Local Area Networks (VLANs) and VLAN Trunking Protocol (VTP). As a layer 2 device no protocols are implemented in the upper layers 4 to 7. Again for the sake of clarity the diagram only represent one interface (Fast Ethernet 0/1). It is possible to modify the diagrams to include multiple Ethernet interfaces.

A state diagram has also been designed for a PC. A PC implements all 7 OSI layers however particular attention was paid to the lower 3 layers which interact with internetworking devices. Hence a PC state diagram has an interface table (IP address, Subnet mask and Gateway IP and an ARP table.

Using the state diagrams for the internetworking devices switch and router it is possible to capture on a single diagram the information from a number of different hierarchical CLI commands. Furthermore the diagrams can be used to represent state information that may not be explicitly provided by the CLI e.g. ARP states free, pending and resolved. The diagrams also include the OSI and TCP/IP layers. It is therefore possible to clearly see which protocols operate in which layer. Arguably these state diagrams are: visible, expressive, understandable and easy to use hence of value in supporting student learning. Accordingly the diagrams were evaluated as a teaching tool.

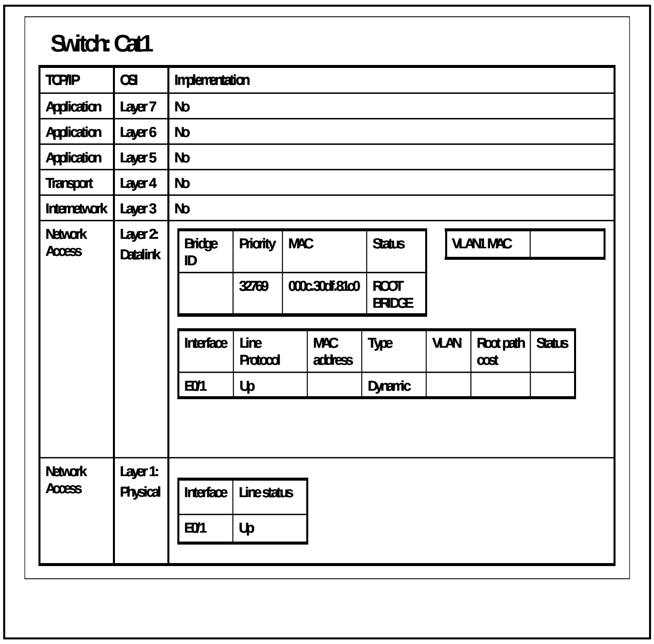


FIGURE 4: SWITCH STATE VARIABLE MODEL

PEDAGOGICAL EVALUATION

The state models of a switch and router were used as the pedagogical foundation of two postgraduate units in network technology and the results evaluated. The results indicated a significant educational benefit [46]. It was found that, '*Postgraduate students, whose learning was based upon the state models, demonstrated a comprehension of devices comparable to a qualified and experienced expert in this field. Furthermore these students performed significantly better than other students both within the same and a different institution. Postgraduate students are arguably more mature and are likely to have better study skills than undergraduate students. However one group of postgraduate students had completed only 24 hours of instruction in contrast to CCNP students who had successfully completed the CCNA (96 hours of instruction) and an additional semester of CCNP material (at least 96 hours of instruction).'* Further work is needed but this very positive result leads to the further development of these models.

ADVANCED STATE VARIABLE MODELS

The models were further developed in two ways. Firstly they were used to describe more advanced routing protocols. Secondly, in order to control even greater technical complexity diagram leveling was used. Leveling is the process of partitioning (or structuring) a complex system into a number of independent but linked units or modules of desirable size so that the system may be more easily understood. The highest level (level 0) module is a single diagram that describes the entire system. Subsequent diagrams are expansions of this level 0 diagram and numbered accordingly. Furthermore, all the diagrams must be linked thereby allowing navigation between them. Leveling is essential for controlling complexity in a hierarchical manner i.e. Information hiding.

All the main Interior Gateway Protocols – distance vector (Interior Gateway Routing Protocol - IGRP), link state (Open Shortest Path First - OSPF) and balanced hybrid (Enhanced Interior Gateway Routing Protocol - EIGRP) were analyzed to obtain the most essential routing state information and represent it on one or more tables. These tables were incorporated into a single state diagram for each protocol i.e. level 0 state diagrams. For example, the level 0 state diagram for the routing protocol EIGRP consists of the RIP state diagram with two additional tables – the topology and neighbor tables (Figure 5).

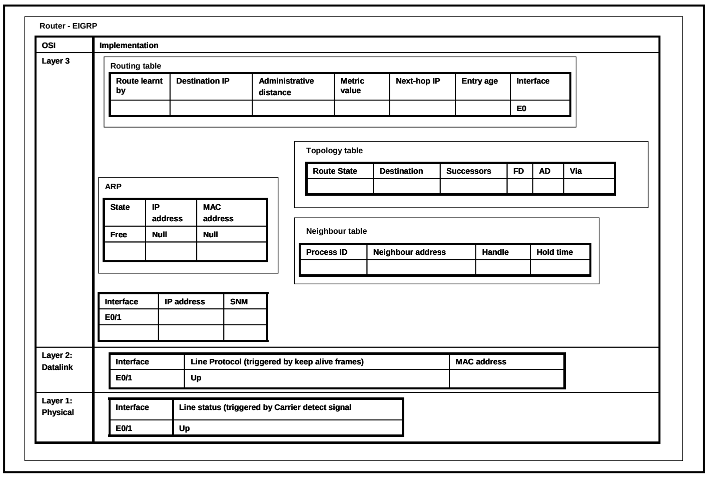


FIGURE 5 EIGRP

The neighbor table lists adjacent routers and associated details. The topology table includes route entries for all destinations that the router has learned including information about successor, feasible distance etc. Similarly a single, level 0, diagram was produced for IGRP (Figure 6) and OSPF.

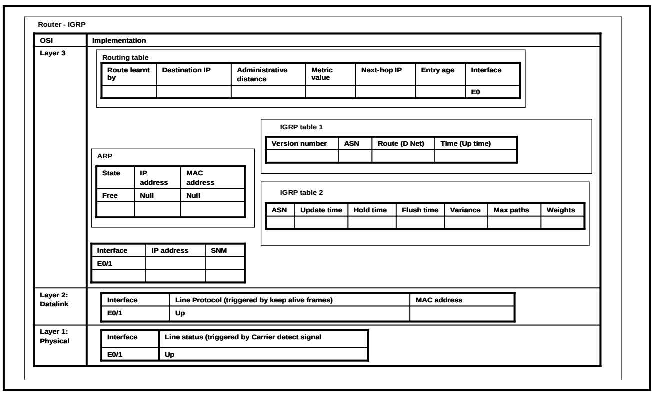


FIGURE 6 IGRP

It was also possible to design a level 0 state diagram for Hot Standby Routing Protocol (HSRP) (Figure 7).

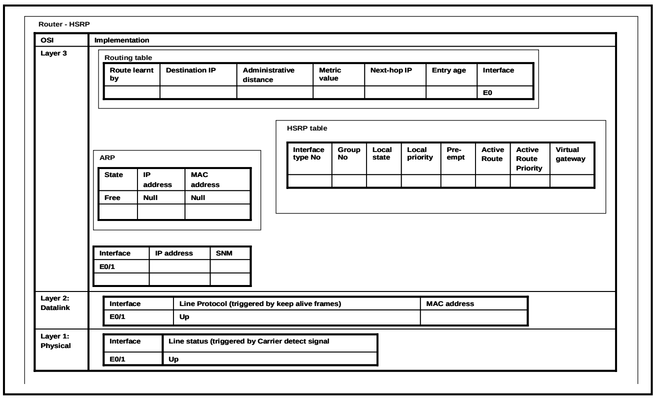


Figure 7 HSRP

Border Gateway Protocol (BGP) is a complex Exterior Gateway Protocol. However it was possible to design a single, level 0 state diagram for this protocol (Figure 8).

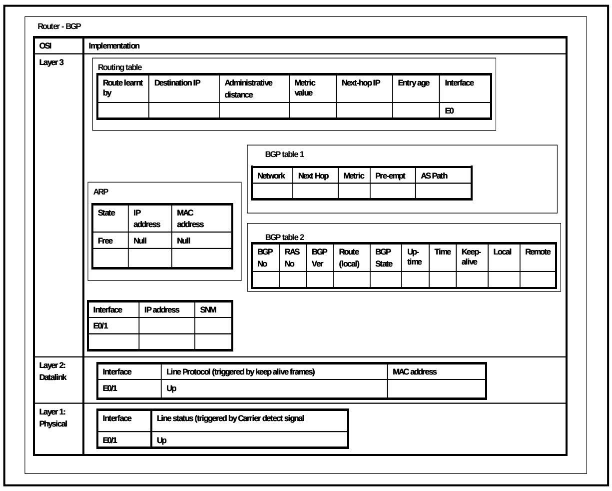


Figure 8 BGP

Significantly the diagrams are modular hence it is possible to employ the same basic router configuration used for the routing protocol RIP and then add state tables specific to more complex routing protocols.

There are a large number of switching technologies that include: Fast switching, Autonomous switching, Optimal switching, distributed switching, Cisco Express Forwarding (CEF), Netflow switching, Tag Switching etc. Preliminary work suggests that is it possible to design a single, level 0 state diagram for two Multi-layer Switching (MLS) and Cisco Express Forwarding (CEF).

For most of these protocols it was possible to further design one or more subsequent diagrams at different levels e.g. level 1, level 2 etc. Different level diagrams are connected by hyperlinks allowing the user to move up or down the different levels of complexity. By example, the level 0 switch state diagram in OSI layer 2 has three tables – Interface, Bridge and VLAN. It is possible, using a hyperlink to display the level 1 diagram that shows Bridge Protocol Data Unit (BPDU) details and associated port states (Blocking, Forwarding, Listening or Learning).

All experimental work was conducted on Cisco equipment – 2620XM and 2621XM routers; Catalyst 2950 and Catalyst 3550 switches. Further work needs to be conducted on different platforms to confirm that the diagrams are platform independent and hence portable. Preliminary investigations suggest this to be the case.

CONCLUSIONS

New, diagrammatic state models of a switch and router were developed and successfully used as the pedagogical foundation of a practical, ‘hands on’ curriculum in networking. It was found that they significantly improved student learning. Further more extensive evaluation is needed but the very positive results encouraged subsequent development of the models. The models are diagrammatic, self-documenting and easy to use. Using the models it is relatively easy to understand the purpose and structure of the devices. The models include implementation details, derived from CLI commands, hence it is possible to verify and validate device operation. The modular nature of these diagrammatic models allows the user to appreciate the interaction between the different protocols operating on a device. Furthermore the modularity allows one to have a basic model (e.g. a router running the RIP routing protocol) whose functionality can be enhanced by the inclusion of additional state tables. Hence the router state model can be used for all the main Interior Gateway Protocols – distance vector (RIP, IGRP), link state (OSPF) and balanced hybrid. (EIGRP). It can also model Hot Standby Routing Protocol (HSRP). Preliminary work indicates that it also supports an Exterior Gateway Protocol (BGP). The switch state model can be used for both basic and advanced switch operation (STP, VTP and VLANs). Both models employ leveling and hence provide hierarchical top-down decomposition thereby controlling technical detail. In effect, the diagrams integrate leveled diagrams with protocol finite state machines and the output of internetworking CLI command output. Preliminary results indicate that the models are platform independent and hence portable between different vendor devices. However further work is needed.

REFERENCES:

1. Kurose, J., et al. *ACM SIGCOMM Workshop on Computer Networking: Curriculum Designs and Educational Changes*. in *ACM SIGCOMM Workshop on Computer Networking*. 2002. Pittsburgh, PA: ACM.
2. Maj, S.P., et al., Computer and Network Installation, Maintenance and Management - A Proposed New Curriculum for Undergraduates and Postgraduates. The Australian Computer Journal, 1998. **30**(3): p. 111-119.
3. Maj, S.P. and J. Dharukeshwari, Vendor Based Network Engineering Education - An International Comparison. World Transactions on Engineering and Technology Education, 2003. **2**: p. 313-316.
4. Systems, C., *Cisco Network Academy Program: First Year Companion Guide*. 2nd ed. 2001, Indianapolis: Cisco Press.
5. Heap, G. and L. Maynes, *CCNA Practical Studies*. 2002, Indianapolis: Cisco Press.
6. Odom, W., *Cisco CCNA Exam #640-507 Certification Guide*. 2000, Indianapolis: Cisco Press.
7. Systems, C., *Cisco Network Academy Program: Second Year Companion Guide*. 2001, Indianapolis: Cisco Press.

8. Shaughnessy, T. and T.J. Velte, *Cisco, A Beginner's Guide*. 2000, New York: Osbourne/McGraw-Hill.
9. Hudson, K., K. Caudle, and K. Cannon, *CCNA Guide to Cisco Networking*. 2nd ed. 2000: Thomson Course Technology.
10. Lammle, T., *Cisco Certified Network Associate Study Guide*. 2nd ed. 1999, San Francisco: Sybex.
11. Velte, T.J. and A.T. Velte, *Cisco, A Beginner's Guide*. 2nd ed. 2001, New York: Osbourne/McGraw-Hill.
12. McQuerry, S., *Interconnecting Cisco Network Devices*. 2000, Indianapolis: Cisco Press.
13. Webb, K., *Building Cisco Multilayer Swtiched Networks*. 2000, Indianapolis: Cisco Press.
14. Lewis, W., *Multilayer Switching Companion Guide*. 2003, Indianapolis: Cisco Press.
15. Hucaby, D., *CCNP Switching Exam Certification Guide*. 2001, Indianapolis: Cisco Press.
16. Hucaby, D. and S. McQuerry, *Cisco Field Manual: Catalyst Switch Configuration*. 2003, Indianapolis: Cisco Press.
17. Lammle, T. and E. Quinn, *CCNP Switching Study Guide*. 2nd ed. 2001, San Francisco: Sybex.
18. Thomas II, T.M., J.C. Bass, and J.E. Robinson III, *Building Cisco Multilayer Switching Networks*. 2000, New York: McGraw-Hill.
19. Paquet, C. and D. Teare, *Building Scalable Cisco Networks*. 2001, Indianapolis: Cisco Press.
20. McGregor, M., *Semester Five Companion Guide, Advanced Routing*. 2001, Indianapolis: Cisco Press.
21. Benjamin, H., *CCNP Practical Studies: Routing*. 2002, Indianapolis: Cisco Press.
22. Lammle, T., C. Timm, and S. Odom, *CCNP/CCIP BSCI Study Guide*. 2002, San Francisco: Sybex.
23. Zinin, A., *Cisco IP Routing*. 2002, Boston: Addison Wesley.
24. Grice, M., *CCNP Guide to Advanced Cisco Routing*. 2001: Thomson Learning.
25. Thomas II, T.M., et al., *Building Scalable Cisco Networks*. 2000, Indianapolis: Cisco Press.
26. Paquet, C., *Building Cisco Remote Access Networks*. 1999, Indianapolis: Cisco Press.
27. Thomas II, T.M. and A. Quiggle, *Building Cisco Remote Access Networks*. 2000, New York: McGraw-Hill.
28. Shinder, D.L., *Computer Network Essentials*. 2001, Indianapolis: Cisco Press.
29. Systems, C., *Internetworking Technologies Handbook*. 2001, Indianapolis: Cisco Press.
30. Pignataro, C., R. Kazemi, and B. Dry, *Cisco Multiservice Swtiching Networks*. 2003, Indianapolis: Cisco Press.
31. Davies, N., S. Ransbottom, and D. Hamilton, *Teaching Networks through Modelling.* ACM SIGada Letters, 1998. **XVIII (5)**: p. 104-110.
32. Neufeld, M., A. Jain, and D. Grunwald. *Nsclick: Bridging Network Simulation and Deployment*. in *International Workshop on Modelling Analysis and Simulation of Wireless and Mobile Systems*. 2002. Atlanta, Georgia: ACM Press, New York, NY, USA.

33. Ben-Ari, M. *Constructivism in Computer Science Education*. in *Twenty Ninth SIGCSE Technical Symposium on Computer Science Education*. 1998. Atlanta, Georgia: ACM Press.
34. von Glasersfeld, E., *Cognition, Construction of Knowledge, and Teaching.* Synthese, 1989(80): p. 121-140.
35. Cooling, J.E., *Software Design for Real-Time Systems*. 1991, Padstow, Cornwall: Chapman and Hall.
36. Booch, G., *Object-Oriented Analysis and Design*. 2nd ed. 1978, Redwood City, CA: Benjamin/Cummings.
37. 830-1993, I.S., *Software Requirements Specification*. IEEE Standards Collecton, Sofware Engineering. 1993, NJ: IEEE.
38. Gane, C. and T. Sarson, *Structured Systems Analysis: Tools and Techniques*. 1979, Englewood Cliffs, NJ: Prentice-Hall.
39. Jackson, M., *Principles of Program Design*. 1975, New York: Academic Press.
40. Jackson, M., *System Development*. 1983, Englewood Cliffs, NJ: Prentice-Hall.
41. Rockstrom, A. and R. Saracco, SDL-CCITT specification and description language. IEEE Transaction on Communications, 1982. **30 (6)**: p. 1310-1318.
42. Marca, D. and C. McGowan, *SADT: Structured Analysis and Design Technique*. 1988, New York: McGraw-Hill.
43. Gomaa, H., *Software Design Methods for Concurrent and Real-Time Systems*. 1993, Reading, MA: Addisson-Wesley.
44. Peters, J.F. and W. Pedrycz, *Software Engineering - An Engineering Approach*. World Series in Computer Science, ed. D. Barron and W. P. 2000, New York: John Wiley &Sons.
45. Halsal, F., *Data Communications, Computer Networks and Open Systems*. 1996, Harlow, England: Addison-Wesley.
46. Maj, S.P. and G. Kohli, *New State Models for Internetwork Technology.* Issues in Informing Science and Information Technology Education, 2004. Publication pending.

5.10 A Pedagogical Evaluation of New State Model Diagrams for Teaching Internetworking Technologies

Title of the paper :	A Pedagogical Evaluation of New State Model Diagrams for Teaching Internetworking Technologies
Authors	S P Maj, G Kohli, T Fetherston
Conference:	ACSC 2005
Conference Location:	Newcastle, Australia
Conference date:	31st Jan – 3rd Feb 2005

This is an annual conference held in the Asia-Pacific region. The conference publishes some of the best work in computer science in Australia and New Zealand. According to the program director "*We received 165 abstracts and 122 full papers. Each paper was peer-reviewed by at least two independent reviewers, and in some cases three or four referees produced independent reviews. The program committee was impressed by the quality of the submissions. Only 41 papers were accepted. This is slightly less than in pervious years, and indeed reflects a higher standard since a conscious decision was made to select papers for which all reviews were positive and favorable. Although the Program committee made careful quantitative and qualitative assessment of the feedback from reviewers, it is remarkable that all those accepted papers had a weighted score of 5 or above (in a scale from 1 to 7, here 7 correspond to Strong Accept [award quality], 6 corresponds to Accept [I will argue for this paper] and 5 correspond to Weak Accept [I vote for this paper, but won't object reject])*". Furthermore, the proceeding was published in association with the American Computing Machinery digital library.

Curriculum based on internetworking devices is primarily based on the Command Line Interface (CLI) and case studies. However, a single CLI command may produce output that is not only hierarchical but must also be interpreted – both represent learning difficulties for novices. It should also be noted that device status requires many different CLI commands. Internetworking curriculum also typically defines devices as ‘black boxes’ - this is not a good teaching strategy. New state models were designed that diagrammatically integrated relevant output from different CLI commands with protocol finite state information and protocol stacks by means of tables. The diagrams are modular and hierarchical thereby providing top-down decomposition by means of levelling. Hyperlinks may be used to navigate between different state tables and diagrams.

There is unequivocal research evidence that conceptual models are powerful tools that support learning. Cisco use standard symbols for different devices - there is a symbol for a switch and one for a router. These symbols are used by the CNAP curriculum. However, they are only symbols and do not provide detail about the operation of the device i.e. ‘black boxes’. Primarily the Command Line Interface (CLI) provides this detail. However, a single CLI command may produce output that is not only hierarchical but must also be interpreted – both represent learning difficulties for novices. It should also be noted that device status requires many different CLI commands.

The new state models were designed to diagrammatically integrate relevant output from different CLI commands with protocol finite state information and protocol stacks by means of tables. Many of the major switching/routing protocols have been successfully implemented using these diagrams. Furthermore, their modular, hierarchical characteristics allow a technical detail to be introduced in an integrated and controlled manner thereby supporting student learning both at introductory and advanced level. Results to date suggest that students learning based on state diagrams demonstrates a richer conceptual understanding strongly aligned with that of an expert. However, further work is needed.

A Pedagogical Evaluation of New State Model Diagrams for Teaching Internetwork Technologies

S P Maj, G Kohli, T Fetherston
School of Computer and Information Science
Edith Cowan University
2 Bradford Street, Mount Lawley, 6050, Western Australia
p.maj@ecu.edu.au

Abstract

Curriculum based on internetworking devices is primarily based on the Command Line Interface (CLI) and case studies. However a single CLI command may produce output that is not only hierarchical but must also be interpreted – both represent learning difficulties for novices. It should also be noted that device status requires many different CLI commands. Internetworking curriculum also typically defines devices as 'black boxes' - this is not a good teaching strategy. New state models were designed that diagrammatically integrated relevant output from different CLI commands with protocol finite state information and protocol stacks by means of tables. The diagrams are modular and hierarchical thereby providing top-down decomposition by means of levelling. Hyperlinks may be used to navigate between different state tables and diagrams. The models were used as the pedagogical foundation of internetworking curriculum and results compared with control groups who were taught in the standard manner. The students who were taught using the state models clearly demonstrated an understanding that was comparable to an expert in this field. It is well documented in education research that after successfully completing an examination it is not uncommon for the majority of students to demonstrate very poor long term retention of concepts. Students taught using the new state models clearly demonstrated concept retention six weeks after the final semester examinations.

Keywords: Modelling, Routers, Switches

Introduction

Net-Centric computing is part of the ACM/IEEE Computing Curriculum 2001 Undergraduate Body of Knowledge (ACM, 2001). There are different but educationally equally valid approaches to teaching networking. According to (Kurose & Ross, 2000), '*Among the approaches towards networking curricula, one finds the more quantitative (electrical engineering) style of teaching networking versus a more software/algorithmic (computer science) approach, the more "hands-on" laboratory based approach versus the more traditional in-class lecture based approach; the bottom-up approach towards the subject matter versus a top-down approach*.'

Both within Australia and internationally there is a demand for the practical hands-on approach associated with teaching internetworking device technologies. Accordingly some universities have adopted the Cisco Network Academy Program (CNAP) and hence obtain access not only to vendor specific curriculum and certification (Cisco Certified Networking Associate (CCNA) and Cisco Certified Networking Professional (CCNP) but also low cost devices (switches and routers). It should be noted that the CCNP is based upon an educational web site that cost US$25 million to develop and an extensive repertoire of Cisco sponsored textbooks (Cisco, 2004).

Based on a market analysis of employer expectations this university implemented the vendor-based CCNA and CCNP curriculum (G. Murphy, Kohli, Veal, & Maj, 2004). Certainly vendor-based education has both strengths and weaknesses; critics and advocates; however this debate is beyond the scope of this paper (D Veal, 2004). It is not uncommon for a university, as is the case with this university, to use the Cisco equipment not only for teaching the CCNA and CCNP curriculum but also as the basis for teaching other non-vendor specific networking units.

An extensive analysis of CNAP curriculum (on-line and off-line material) found that this type of curriculum primarily teaches internetworking device functionality via case studies using the Command Line Interface (CLI) (Maj, Murphy, & Kohli, 2004). The CLI allows the user to determine and modify the status of the various components of a device such as routing table entries, Address Resolution Protocol (ARP) table entries, interface status etc. However a single CLI command may produce output that must be interpreted and also the hierarchical nature of CLI commands is often difficult for novices to understand. Furthermore the status information of the many different device protocols, interfaces etc must typically be obtained by a number of different CLI commands. This may be problematic during teaching when it is necessary to integrate all of this information from a number of different, and possibly complex, CLI commands. Because of this students have problems identifying and understanding the concepts underlying the use of the CLI. It should be noted however that for experienced network engineers the CLI is a very powerful and useful tool.

It was also found that these devices are typically defined as 'black boxes'. While the internal functioning of black boxes can be inferred from input and output behavior, this is not a good teaching strategy. A more efficacious approach may be to provide students with a good conceptual model at the start of their studies. Accordingly Maj et al designed new state models of a switch and a router (Maj et al., 2004) (Kohli, Maj, Murphy, & Veal, 2004).

New State Models – Switch and Router

These new state models diagrammatically integrate relevant output from different CLI commands with protocol finite state information and also the OSI-TCP/IP protocol stacks by means of tables (Figure 1 and Figure 3).

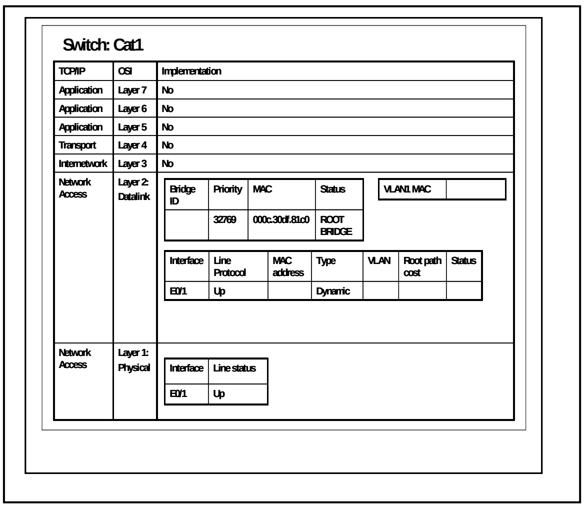


Figure 1: Switch State Model Diagram

Protocol finite state data allows dynamic, transitional states to be capture. For example the Address Resolution Protocol (ARP) exists in one of three states – free, pending and resolved (figure 2).

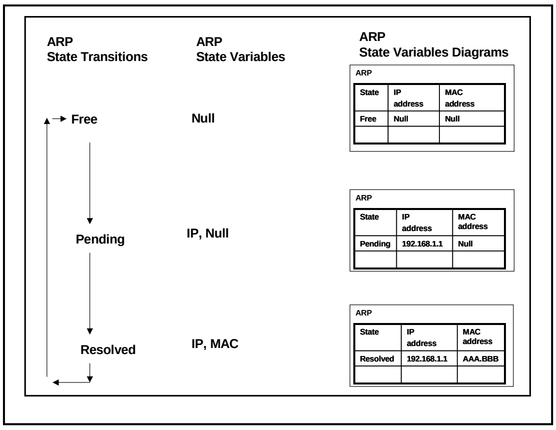


Figure 2: ARP States

In the free state the time-to-live entry has expired; the pending state means a request for an IP entry has been sent but no reply received and the resolved state means the physical (MAC) to logical (IP) mapping is complete.

Specific finite state information can also be captured for other protocols such as Spanning Tree Protocol (STP). It should be noted that Cisco use finite states to explain neighbour acquisition in BGP, route acquisition in EIGRP and also neighbour and route acquisition in OSPF. However this is not integrated with router management at the CLI interface.

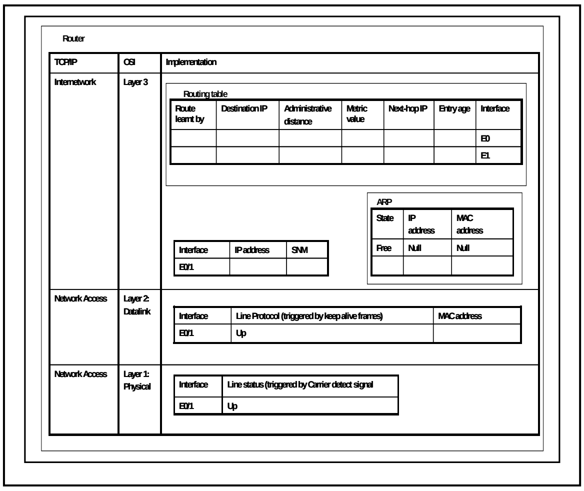


Figure 3: Router State Model Diagram

Each table in these diagrams has clearly defined functionality and is substantially independent from other tables i.e. the tables have high cohesion and low coupling. It is therefore possible to selectively include or exclude different tables. For example, the switch interface line status table (figure 1) may be removed from the diagram thereby allowing the diagram to more easily accommodate other tables associated with Spanning Tree Protocol.

In effect the tables are modular thereby providing top-down decomposition by means of levelling. One of the most important features of these diagrams is the ability to construct a variety of different levels according to the level of abstraction required. This means that the initial (level 0) diagram can be consulted to obtain a high level overview of device interconnectivity i.e. what switches and routers are connected together. When details of a specific device are needed, then greater technical complexity for that specific device can be obtained. Obviously the different levels of diagram are consistent. The diagrams may be decomposed to the level that is meaningful for the purpose that the diagram is required. For example, a router OSI layer 1 line status (carrier detect table) would not be decomposed any further – the interface is in effect either up or down. However for a more complex OSI layer 3 protocol such as Enhanced Interior Gateway Routing Protocol (EIGRP) further levels of diagram can be used to include EIGRP Topology and Neighbour tables. Hyperlinks may be used to navigate between not only different devices but also different tables in the same device.

It is possible, using the switch state model, to implement protocols that include: basic switch operation (address learning, forwarding/filtering), Spanning Tree, Virtual LAN's (VLANs), trunking, VLAN Trunking Protocol (VTP) and Ether-channelling (Maj et al., 2004). It is possible, using the router state model to implement protocols that include: all the Interior Gateway Protocols (Routing Information Protocol, Interior Gateway Routing Protocol), link state protocols (Open Shortest Path First) and balanced hybrid protocol (Enhanced Interior Gateway Routing Protocol) (Maj et al., 2004).

Pedagogical Application

These diagrams were used as the pedagogical foundation of a non-vendor based networking curriculum. The curriculum was designed based on the diagrams to progressively introduce and explicitly link networking concepts. Novice students were given a switch diagram relating the connection functionality to the OSI and TCP/IP model (figure 4). This is directly relevant to student's initial perceptions of a switch - a networking device for connecting together PCs. At this stage students do not 'know' how a switch works. They are also introduced to the CLI interface and a few simple commands. Using these commands it is possible to determine interface details – typically complex and lengthy (figure 5).

In the diagrams, an interface has a layer 1 table (Carrier Detect table) that can be used to determine the physical link status i.e. either up or down based on the carrier detect signal. Each interface has an associated layer 2 table (Line Protocol table) based on the Line Protocol Status which is either up or down based on the Line Protocol status. Every interface has both of these tables. This data is extracted from the associated CLI interface commands.

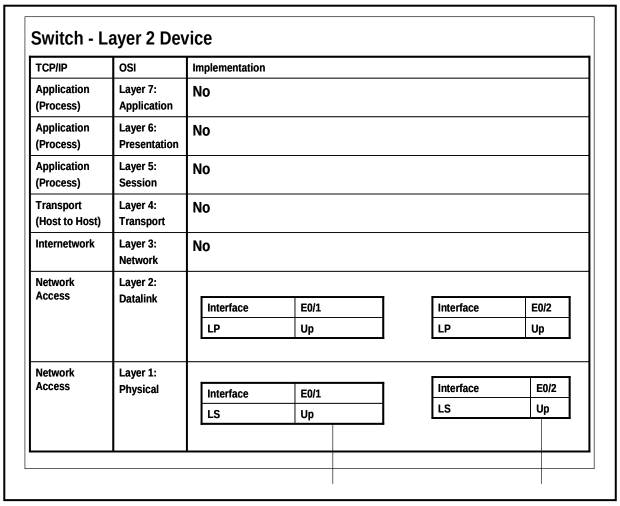


Figure 4: Switch – OSI and connections only

It can be clearly seen that a switch is an OSI layer 2 device with no implementation of any higher level protocols. Accordingly, the diagram may be simplified (figure 6).

```
Device#show interface fastethernet 0/1
FastEthernet0/1 is up, line protocol is up
  Hardware is Fast Ethernet, address is 000c.30df.81c1 (bia 000c.30df.81c1)
  MTU 1500 bytes, BW 100000 Kbit, DLY 1000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Auto-duplex, Auto-speed
  input flow-control is off, output flow-control is off
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 23:41:43, output 23:41:30, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue :0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     536 packets input, 72501 bytes, 0 no buffer
     Received 276 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 164 multicast, 0 pause input
     0 input packets with dribble condition detected
     3005 packets output, 212679 bytes, 0 underruns
     0 output errors, 0 collisions, 2 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 PAUSE output
     0 output buffer failures, 0 output buffers swapped out
```

Figure 5: Switch – show interface output

Removal of these non-essential features permits the introduction of the switch MAC address table (figure 7). It can be clearly seen from the diagram that the MAC address table is an OSI layer 2 protocol.

At this stage it is possible to connect PCs to the switch and demonstrate basic switch operation – address learning and address forwarding/filtering. The MAC addresses of the PCs can be entered into the MAC address table and the results confirmed by the CLI command '*show mac-address-table*'.

With a few minor modifications this diagram can be used to teach concepts that include: flooding, broadcasting, micro-segmentation (virtual circuits), broadcast and collision domains, address aging, static and dynamic addressing and VLANs (static and dynamic implementations).

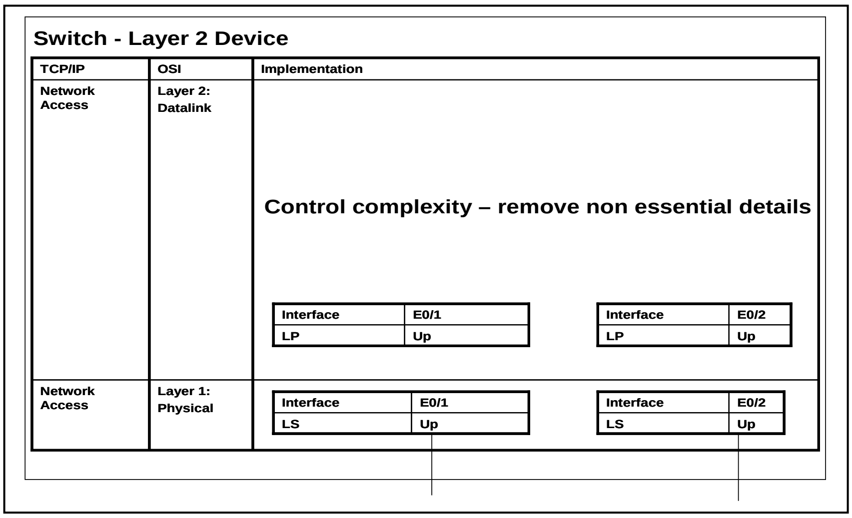


Figure 6: Switch – Simplified OSI and connections

In order to teach Spanning-Tree Protocol (STP) it is necessary to remove extend the interface table details and introduce the Bridge Identification table (figure 8).

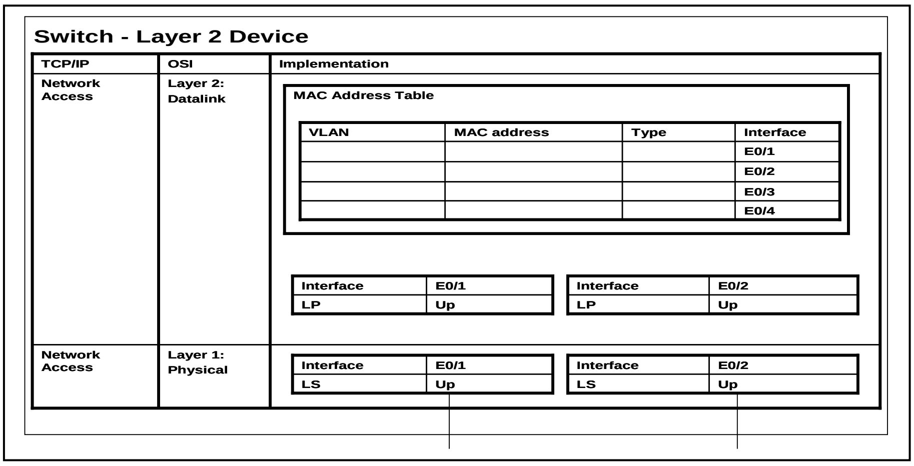


Figure 7: Switch and MAC address table

As an aid to clarity it is possible to remove the MAC address table and layer 1 and 2 interface details (figure 9). Using this diagram it is possible to teach all aspects of STP such as: Bridge Id (priority and address), Bridge status and timers, interface id and priority, switch states, best BPDU etc. The simplified diagram may be reduced in size allowing multiple switch diagrams to be included in a single Power Point slide image. The interaction between the switches and the associated changes can then be viewed concurrently. This concurrency is not possible if only the output from CLI commands is used.

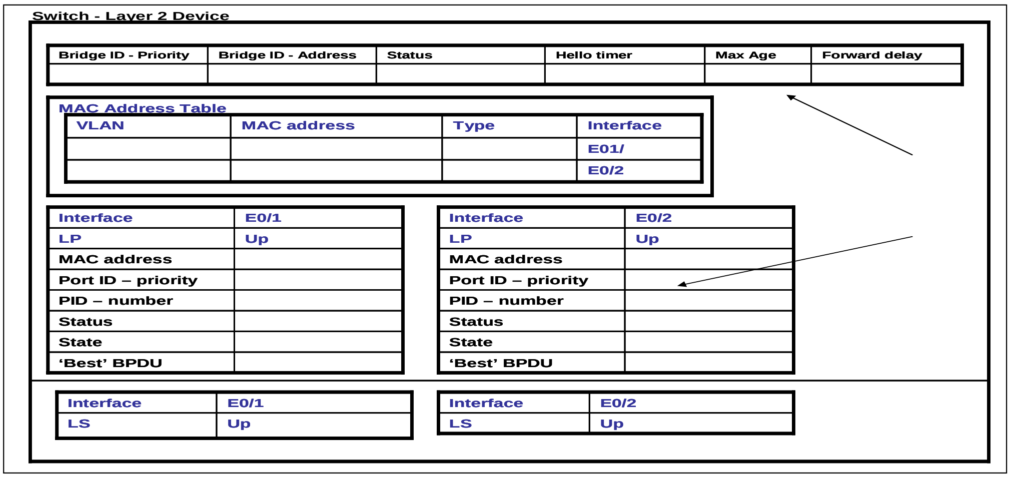


Figure 8: Switch – MAC address table and STP tables

State diagrams of a router were similarly developed and used as the pedagogical foundation of the new networking curriculum.

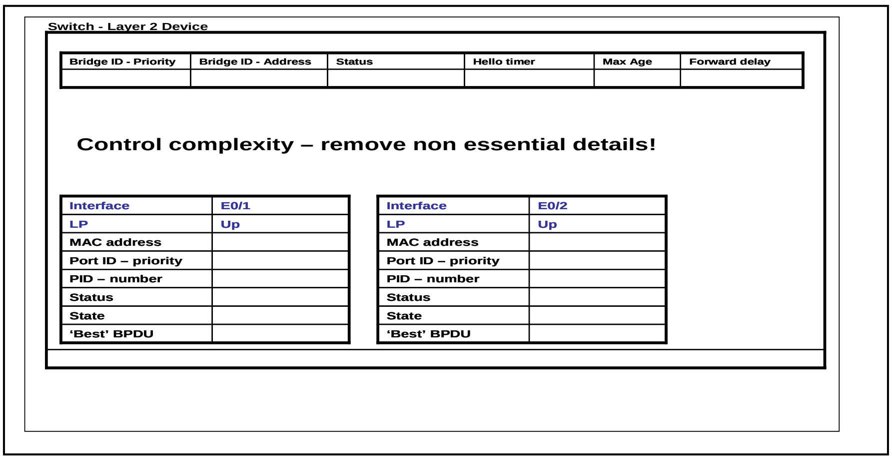


Figure 9: Switch STP tables only

Educational Theory

The above approach is predicated on the assumption that students encountering phenomena in computer science classes will construct their own mental model of the concept encountered. This constructivist view recognises that before learning can take place a student must engage with, and process, perceived information. This processing occurs with the assistance or in the presence of recalled prior knowledge and such prior knowledge will include any existing mental models of the student. Such a view of learning accepts that initial constructions of perceived information can be quite limited in breadth and depth and that subsequent building on these ideas to form higher-level conceptual structures requires effort and purpose on behalf of the learner. The process of building better conceptual structures can be assisted by the provision of good models that form the bridge between students' existing ideas and ideas that form part of the body of knowledge of the particular domain. These models are probably essential for understanding complex networking systems, to be able to predict and hypothesise about these systems and in general to be able to understand such systems.

Good mental models that allow the construction of richer ideas are also important as they can assist conceptual change. Research conducted in science education over at least the last 25 years has demonstrated across many contexts that students' existing ideas are often firmly held and unfortunately often wrong. For example Driver and Erikson suggested as long ago as 1984 that students' existing ideas often interfere with instruction and their alternative conceptualisations can hinder further learning (R. H. Driver & Erikson, 1984).

At about the same time a group of researchers at Cornell University developed a theory of conceptual change (Posner et al., 1982). Posner et al. applied (Kuhn, 1970) ideas about scientific revolutions to individual learning, and derived the following conditions for bringing about conceptual change:
1. There must be dissatisfaction with a currently held conception.
2. The alternative conception must be intelligible.
3. The alternative conception must appear plausible.
4. The alternative conception must appear fruitful.

They reviewed their ideas about conceptual change in 1992 but in both versions the importance of good mental models was apparent. The attributes above give clues as to what constitutes a good (alternative to student's existing) mental model – it must be intelligible, plausible and potentially fruitful.

The ability to recall and explain a concept does not necessarily reflect understanding, nor does it guarantee that students can apply and use the concept in a meaningful way (Julyan & Duckworth, 1996). One main goal of teaching is to enable the students to create a valid mental model of new concepts, consistent with formal Computer Networking definitions. The student's mental model should be built on a set of correct pieces of knowledge that are concerned with the learned concepts. Models can be employed as an aid to conceptual understanding. Thomas notes in respect to student difficulty in understanding concepts that: "*The use of* diagrammatic representation provides an alternative to just offering more words which may only compound their difficulties" (Thomas, 2000). Gilbert notes that "a model is a simplified representation of a system, which concentrates attention on specific aspects of the system, Moreover, models enable aspects of the system, i.e. objects, events, or ideas which are either complex, or on a different scale to that which is normally perceived, or abstract to be rendered either visible or more readily visible" (Gilbert, 1995).

Models have been used in science education and the use of the models has been well documented in science education research (Gilbert & Boulter, 1998; Linn & Muilenberg, 1996). Many authors suggest that providing an individual with a conceptual model of a system before instruction enhances user learning (Bayman & Mayer, 1984; Carroll & Mack, 1992; Moran, 1981). Conceptual models supports superior learning over giving only procedural instruction and this has been noted by (Davis & Bostrom, 1993).

Before a conceptual change can take place, the naïve concepts that a students possess have to made explicit (Wichmann et al., 2003). Good mental models can assist this. For example a switch will have a 'mac-address-table' and router will have 'routing table' and these could be integrated into conceptual model. Mental models provide a powerful mechanism for storing knowledge within the human mind (Norman, 1983). Because of the ways in which they can influence behaviour, such structures have a significant impact on virtually all forms of human activity (Barker & van Schaik, 1999). Mental models need to utilize abstraction: "*Schemas are conceptual structures and processes which enable human beings to store perceptual and conceptual information about the world and make interpretations of events through abstraction*" (D'Andrade, 1992). The mental image of the world around us that we carry in our heads is a model (von Glasersfeld, 1992). "*One does not have a city or a government, or a country in his head. One has only selected concepts and relationships, which one uses to represent the real system*" (Forrester, 1971). He further states that "*The mental model is fuzzy. It is incomplete. It is imprecisely stated. Furthermore, within one individual, a mental model changes with time and even during the flow of a single conversation*" (Forrester, 1994). A student's mental model can be inferred from a concept map provided by them.

In learning to construct a concept map, it is important for the students to begin with a domain of knowledge that is very familiar to them and which contains recognisable terms. Since concept map structures are dependent on the context in which they will be used, students doing the networking units should be provided with schemes of networking devices which they can use to learn about how a internetworking device operators. Using a layer based system similar to the OSI model will help students to create a mental schema of the technology they are learning. By doing this one creates a context that will help to determine the hierarchical structure of the concept map. It is also helpful to select a limited domain of knowledge for the first concept maps. In the case of internetworking it should follow a progression from the introduction of the switch concept followed by the routing concept. Switch works at layer 2 of the OSI model and are part of the LAN. On the other hand router works at Layer 3 and are part of both LAN and WAN.

Experimental Results

Maj and Kohli conducted a comparison of standard CNAP curriculum to curriculum based on the new state models (Maj & Kohli, 2004). A network expert was interviewed and his responses to a series of questions recorded. The expert listed a wide range of different terms relevant to the various technologies. The purpose of the interview was to compare students' conceptions with those of the expert with the assumption that teaching should allow students to learn concepts that would eventually enable them to progress to the expert's ideas. The same questions were given to CNAP students and also students taught using the state models and the results evaluated. The CCNA students were able to provide standard definitions but appeared to lack a detailed understanding of device operation. The more advanced CCNP students performed only marginally better. This indicates that students were just recalling learnt material. The students taught using the state model diagrams were also able to provide accurate definitions. But according to Maj, '*However they clearly demonstrated a far better understanding of device operation*.' Furthermore, '*It is highly significant that the responses of the postgraduate students matched those of the network expert*.' (Maj & Kohli, 2004). This indicates a degree of conceptual change and it can be inferred that the model provided may have assisted this. It should be noted that the students taught using the state model had received only 24 hours of instruction compared to the CNAP students who had received over 96 hours of instruction. In a further study by (Kohli et al., 2004) it was found, compared to the CCNP students, that students taught using the state model:

- Were able to correctly use significantly more terms (e.g. routing table, ARP table, broadcast domain etc).
- Gave significantly more responses that matched those from a networking expert.

These two studies did not however provide any measure of understanding. From education research we know that if students are taught with an emphasis on just learning and recalling facts and not concepts then retention of those facts is quite poor – if they are not constantly used.

Two groups of students were evaluated – CNAP students and those taught using the state models. Both groups were evaluated at the start of the semester, the end of the semester and then again six weeks after the final examinations and hence two weeks into the following semester. All students interviewed had enrolled in subsequent networking units – CNAP students enrolled in CNAP units and students taught using the state model enrolled in further units based on the state model. In addition to answering questions students also had to sketch any diagrams to relate different networking concepts. The following results were found:

Start of semester

- Diagrams provided by the state model students were rudimentary.
- Diagrams provided by the CNAP students were rudimentary and primarily consisted of the Cisco symbols for a switch and a router.

End of semester (before final exams)

- Students taught using the state model correctly provide more terms than the CNAP students.
- Students taught using the state model provided more terms that matched those of a networking expert.
- Diagrams provided by the state model students closely matched those provided during the lectures.
- Diagrams provided by the CNAP students were rudimentary and primarily consisted of the Cisco symbols for a switch and a router.

Six weeks after final examinations

- Students taught using the state model gave the same number of correct responses to those at the end of the semester and the same number matched those of the expert
- The CNAP students correctly gave more correct responses than those at the end of the semester but fewer matched those of the expert.
- Diagrams provided by the CNAP students were rudimentary and primarily consisted of the Cisco symbols for a switch and a router.
- Diagrams provided by the state model students correctly contained a lot of technical detail but there were variations from the diagrams that were used in the lectures.

This indicates that both groups of students learnt the required material equally well. However from the diagrams of the state model students it can be inferred that they have richer conceptual understandings and these were aligned with those of the expert. Consequently they will be more able in future learning to progress towards the end state of the expert's understandings. They are also more likely to retain learnt material as this material is linked to more and better concepts thus enhancing recall. To further check the use of the state model as a pedagogical tool a one-hour lecture on Spanning Tree Protocol using the state diagrams was given to practicing networking professionals currently undertaking part time studies towards their CCNP qualification at another institution. There was unanimous and clear support for the models. All comments were recorded and were:

'*We should have more of these*.'
'*Nice to have a conceptual model to aid understanding*.'
'*Yes! The diagrams illustrate the process in a very easy to understand format to allow the subject to be learned*.'
'*Excellent, far clearer than any Cisco material*.'
'*Yes, I have learnt more in this period that in the whole of the semester*.'

Conclusions

There is unequivocal research evidence that conceptual models are powerful tools that support learning. Cisco use standard symbols for different devices - there is a symbol for a switch and one for a router. These symbols are used by the CNAP curriculum. However they are only symbols and do not provide detail about the operation of the device i.e. 'black boxes'. Primarily the Command Line Interface (CLI) provides this detail. However a single CLI command may produce output that is not only hierarchical but must also be interpreted – both represent learning difficulties for novices. It should also be noted that device status requires many different CLI commands.

The new state models were designed to diagrammatically integrate relevant output from different CLI commands with protocol finite state information and protocol stacks by means of tables. Many of the major switching/routing protocols have been successfully implemented using these diagrams. Furthermore, their modular, hierarchical characteristics allow a technical detail to be introduced in an integrated and controlled manner thereby supporting student learning both at introductory and advance level. Results to date suggest that students learning based on state diagrams demonstrates a richer conceptual understanding strongly aligned with that of an expert. However further work is needed.

References

ACM (2001). Computing curricula 2001 ACM/IEEE Joint Task Force Computer Science Final Report, ACM Press.

Barker, P. G. and P. van Schaik (1999). Mental Models and Their Implications for the Design of Computer-Based Learning Resources. Proceedings of the 4th International Conference on Computer-Based Learning in Science (CBLIS '99), University of Twente, Enschede, The Netherlands.

Bayman, P. and R. E. Mayer (1984). "Instructional manipulation of users' mental models for electronic calculators." International Journal of Man-Machine Studies **20**: 189-199.

Carroll, J. M. and R. L. Mack (1985). "Metaphor, computing systems and active learning." International Journal of Man-Machine Studies **22**: 39-57.

Cisco (2004). Cisco Networking Academy Program, Cisco Systems. **2004**.
D'Andrade, R. G. (1992). "Schemas and motivation." In R.G.D'Andrade and C.Strauss (Eds.) Human Motives and Cultural Models. Cambridge.
Davis, S. A. and R. P. Bostrom (1993). "Training End Users: An Experimental Investigation of the Roles of the Computer Interface and Training Methods." MIS Quarterly **17**(1): 61-85.
Driver, R. H. and G. L. Erikson (1984). "Theories in action: Some theoretical and empirical issues in the study of students' conceptual frameworks in science." Studies in Science Education **10**: 37-60.
Forrester, J. (1971). Counterintuitive behavior of social system. Cambridge, MA, Wright-Allen Press, Inc.
Forrester, J. (1994). "Policies, decisions, and information sources for modelling." In Morecroft, J.D.W., and Sterman, J.D. (Eds.), Modelling for Learning Organizations: 51-84.
Gilbert, J. (1995). The role of models and modelling in some narratives in science learning. The Annual Meeting of the American Educational Research Association, San Francisco, CA, USA.
Gilbert, J. and C. Boulter (1998). "Learning science through model and modelling." In B.Fraser and K.Tobin (eds), International Handbook of Science Education: 52-66.
Julyan, C. and E. Duckworth (1996). A constructivist perspective on teaching and learning science. New York, Teacher College Press.
Kohli, G., S. P. Maj, et al. (2004). Abstraction in Computer Network Education: A model based approach. ASEE, Salt Lake City, Utah.
Kuhn, T. (1970). The Structure of Scientific Revolutions. Chicago, University of Chicago Press.
Kurose, J. and K. Ross (2000). Computer Networking A Top-Down Approach Featuring the Internet. New York, Addision-Wesley.
Linn, M. C. and L. Muilenberg (1996). "Creating Lifelong Science Learners: What Models form a firm Fonduation." Educational Researcher **25**(5): 18-24.
Maj, S. P. and G. Kohli (2004). A New State Model for Internetworks Technology. Informing Science + Information Technology Education Joint Conference, Rockhampton, OLD Australia.
Maj, S. P., G. Murphy, et al. (2004). State Models for Internetworking Technologies. 34th ASEE/IEEE Frontiers in Education Conference, Savannah,GA.
Moran, T. P. (1981). "The command langauge grammar: a representation for the user inteface of interactive computer systems." International Journal of Man-Machine Studies **15**: 3-50.
Murphy, G., G. Kohli, et al. (2004). An Examination of Vendor-Based Curricula in Higher and Further Education in Western Australia. American Society for Engineering Education (ASEE), Salt Lake City, Utah.
Norman, D. R. (1983). "Some observartions on mental models." In D.Gentner and A.Stevens (eds), Mental Models.
Posner, G. J., K. A. Strike, et al. (1982). "Accomodation of a scientific conception. Toward a theory of conceptual change." Scince Education **66**(2): 211-227.
Thomas, R. (2000). Fewer Words, clearer picture: The value of a diagrammatic representation of ideas. Third Biennial Communication skills in University education (CSUE), Fremantle, WA, Australia, Teaching and Learning Centre, Murdoch University, Murdoch, WA, Australia.

Veal, D. (2004). An Investigation into Computer and Network Curricula. School of Computer and Information Science. Perth, Western Australia, Edith Cowan University.
von Glasersfeld, E. (1992). Sensory Experience, Abstraction, and Teaching. Alternative Epistemologies in Education Conference, University of Georgia, Lawrencec Erlbaum Associates Inc Hillside NJ.
Wichmann, A., J. Gottdenker, et al. (2003). Developing a Framework for Conceptual Change within Scientific Inquiry. The 3rd IEEE International Conference on Advanced Learning Technologies (ICALT'03).

6. DISCUSSIONS

This section provides discussion on the experimental work undertaken as part of this research and what conclusions may be made.

It is acknowledged by the author that rigorous experimental design is difficult to apply in educational research. A potentially valid criticism of the experimental work in this thesis is that the students who were taught using the state model could have merely mimicked the description provided by their lecturer. Furthermore, students provided with instruction based on the state model diagrams may have been taught by a lecturer who employed a richer and more diverse vocabulary of Internetworking terminology than the lecturer who taught the Cisco curriculum.

However, the experiments consisted of two student groups; one taught using the Cisco material and the other taught using the state model. These two groups were taught by two separate lecturers, neither of whom taught on the other's units. Furthermore, both of these lecturers studied internetworking theory and practice from the same Cisco instructors at the same time and in the same classes. They were required to pass the same assessments and practicals that tested and made use of such vocabulary over the same of period of time. Therefore, as a consequence of these facts, the vocabulary used was strictly standardised and it is unlikely that any major divergence of descriptive terminology would have occurred. Hence the author is reasonably assured both academics would use a very similar vocabulary. However, expert knowledge is derived from years of training and experience. The experiment on expert's metal model was designed, at least in part, to capture key conceptual terms as used by an expert. It was found that expert formed conceptual links not explicitly made in the Cisco course material. An expert's view therefore, consists of tried and tested mental models and as such represents the ideal final outcome of the educational process. It is accepted that mimicking an expert does not make you an expert.

Students' mental models will be constrained by such things as their technical background, previous experience, and study etc. It was observed that students taught using these models provided highly representative and technically accurate descriptions of concepts, even though they had little or no experience in internetworking. In another experiment to evaluate whether the state model diagrams are intrinsic part of student learning six students were identified. These students were able to apply this knowledge when dealing with other networking units that they were studying during this time. Information was sought from students who were enrolled in Masters Units on networking and at the same time these students were enrolled in other networking units such as those based on Cisco and Microsoft networking. During the interviews this group of students were specifically asked whether the state models had helped them in other networking units they were studying at the time. Five out of six students indicated that state models had helped their understanding of Cisco Internetworking. Additionally, all students also stated that by using the state models they were able to explain the routing process and implementation using a Microsoft operating system. These two systems (Cisco and Microsoft) present the user with different forms of implementation even though they make use of the same protocols. Hence this may be indicative that it is not merely memorisation of particular implementation details but of enhanced conceptual understanding.

It is important to note that some of these students had studied on both the vendor based Cisco units and vendor based Microsoft units as well as on the units taught via the use of state models. These six students firmly believed that having the state models improved their understanding of the information presented in both styles. However, further investigations need to be undertaken. This will be conducted using CNAP students who have not studied using the new state model, to test whether or not they have a similar level of enhanced understanding when implementing routing on a Microsoft system and vice versa.

6.1 State Models and Finite State Machine

All communication protocols are based on FSMs. Within Internetworking operating system interaction with the protocols is via the use of CLI. However, the problems with the CLI are:

- CLI produce extensive output that requires considerable skill to interpret and evaluate;
- Different CLI commands are used to understand the different states of a communication protocol;
- Learning the CLI command structure is often difficult for novices

Hence state model diagrams extract and group information into tables for the basic understanding of protocol under investigation.

According to the National Institute of Science and Technology, a non-regulatory federal agency within the United States provide the following definition of FSM

> A model of computation consisting of a set of states, a start state, an input alphabet, and a transition function that maps input symbols and current states to a next state. Computation begins in the start state with an input string. It changes to new states depending on the transition function. There are many variants, for instance, machines having actions (outputs) associated with transitions (Mealy machine) or states (Moore machine), multiple start states, transitions conditioned on no input symbol (a null) or more than one transition for a given symbol and state (nondeterministic finite state machine), one or more states designated as accepting states (recognizer), etc. (NIST, n.d.).

Accordingly, a state model diagrams are based upon the FSM as an aid to device level management.

In comparison to the standard definition the state model diagram has the following FSM characteristics:

- It has a start state and end state.
- It consists of a set of states e.g. Spanning Tree Protocol states Blocking, listing, learning and forwarding.
- Changes to new states depending on the transition function.

Where as the state model diagrams do not have the following FSM characteristics;

- An input alphabet and a transition function that maps input symbols and current states to the next state.
- Multiple start states, transitions conditioned on no input symbol (a null) or more than one transition for a given symbol and state (*nondeterministic finite state machine*).

Furthermore, a state model diagram represents different stages in the communication protocols wherein the output of each state can be represented in the relevant state table. According to Tanenbuam (2003, p.229)

> The key concept used in many protocol models is the finite state machine. With this technique, each protocol machine (i.e., send or receiver) is always in specific state at every instance of time.

Additionally, Halsal notes:

> Irrespective of the specification method, we model a protocol as a finite state machine or automaton. This means that the protocol – or, more accurately, the protocol entity – can be in just one a finite number of defined states at any instant. For example, it might be idle waiting for a message to send or waiting to receive and acknowledgment (Halsal, 1996, p.142).

Consider the (OSI) layer 3 Address Resolution Protocol (ARP). This protocol, or protocol entity, can exist in one of a three defined finite states – free, pending or resolved. In the free state the time-to-live entry has expired; the pending state means a request for an IP entry has been sent but no reply received and the resolved state means the physical (MAC) to logical (IP) mapping is complete. Each state has a state variable – Null, IP address and MAC address. All this information may be represented in an ARP State table in a state model diagram as shown in figure 6.1 and figure 6.2

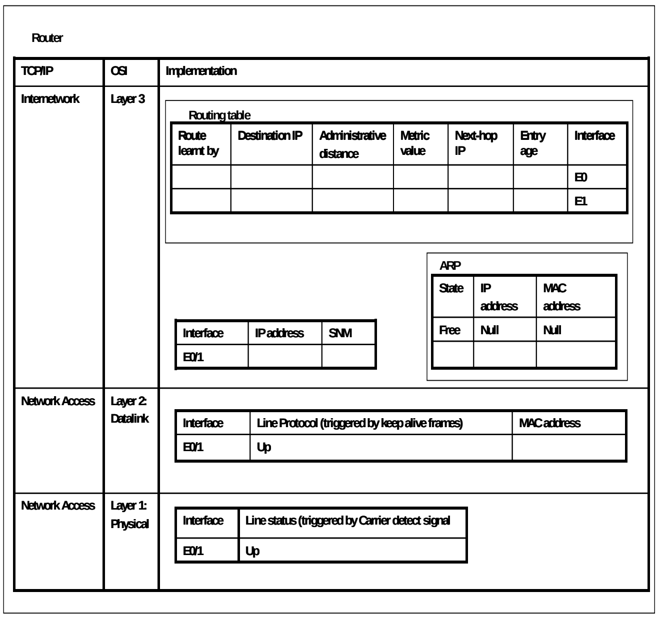


Figure 6.1 State Model of a Router

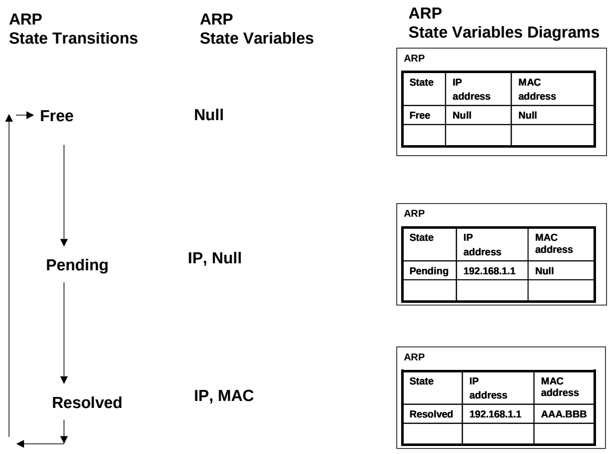


Figure 6.2 ARP State Transitions

The basic concepts can further be extended to show the different states of distance vector routing protocols e.g. Enhanced Interior Gateway Routing Protocol (EIGRP). EIGRP contains several important protocols that greatly increase its operational efficiency relative to other routing protocols (Cisco, 2004). In addition to the routing table, EIGRP maintains two state variable tables – a topology table and a neighbor table. The topology table contains all learned routes to all destinations. The neighbor table lists adjacent routers to ensure bidirectional communication between directly connected neighbours. Both tables included in state model diagrams shows what is considered to be the key information required for basic understanding of EIGRP protocol. As with any other state diagram model the information for EIGRP is captured from different CLI and debug commands on a Cisco router running EIGRP routed protocol.

OSI	Implementation
Layer 3: Network	see tables below
Layer 2: Datalink	see tables below
Layer 1: Physical	see tables below

Layer 3: Network

Route learnt	Destination IP	Admin Distance	Metric Value	Next-Hop IP	Entry Age	Interface
D	192.168.3.1	90	30720	192.168.2.1	35	E0

Topology Table

Route State	Destination	Successors	Mertics	Via
A	192.168.2.1	1	30720	192.168.2.1

State	IP address	MAC Address
Free	Null	Null

Interface	IP address	SNM
E0	192.168.1.1	255.255.255.0

Layer 2: Datalink

Interface	Line Protocol	MAC Address
E0	Up	1234 5678 9ABC

Layer 1: Physical

Interface	Line status
E0	Up

Figure 6.3 EIGRP Topology Table

OSI	Implementation
Layer 3: Network	see tables below
Layer 2: Datalink	see tables below
Layer 1: Physical	see tables below

Layer 3: Network

Route learnt	Destination IP	Admin Distance	Metric Value	Next-Hop IP	Entry Age	Interface
D	192.168.3.1	90	30720	192.168.2.1	35	E0

Route State	Destination	Successors	Mertics	Via
A	192.168.2.1	1	30720	192.168.2.1

Neighbour Table

Process ID	Neighbour Address	Handle	Hold Time
1	192.168.3.1	0	11

State	IP address	MAC Address
Free	Null	Null

Interface	IP address	SNM
E0	192.168.1.1	255.255.255.0

Layer 2: Datalink

Interface	Line Protocol	MAC Address
E0	Up	1234 5678 9ABC

Layer 1: Physical

Interface	Line status
E0	Up

Figure 6.4 EIGRP Neighbour Table

In effect the state model diagrams provide the property of leveling in which complex systems can be progressively decomposed to provide further details whilst maintaining links to higher abstraction levels. For example a level 0 state model diagram of a network would consist of the different devices (routers, switches, wireless access points etc) and their associated connections. It is then possible to select a given device to obtain further details about its operational status (e.g. figure 6.3). This state model diagram can further be expanded to obtain more specific operational details about the EIGRP protocol (e.g. figure 6.4). Navigation between the levels and devices is by means of hyperlinks. A simple hyperlink based interface has been developed. Investigations indicated there is no comparable product available.

7. CONCLUSIONS

B-Nodes provide a good constructivist framework allowing technical detail to be introduced in a controlled, top-down manner that is readily understandable. This model represents a new, higher level of abstraction that is also valid for increasing levels of technical complexity and hence suitable for more advanced technologies. Given this model is based upon abstraction it is therefore independent of architectural detail and can therefore accommodate rapid changes in technology.

Previous work suggests that the B-node model may provide the basis of a suitable conceptual map and hence a pedagogical framework for introductory curriculum on computer technology. This model is conceptually simple and controls the level of detail by abstraction. Hence the model can be used to examine PC performance. Furthermore, the B-Node model may be used for resource modelling (server hardware, switches and routers) and, hence provide an improved way to plan the capacity of infrastructure running critical applications e.g. e-commerce sites. This has been demonstrated by published paper as presented as part of this thesis.

The B-Node model and other conceptual models can be useful in improving understanding of computer and network technologies. All investigations indicates computer networking education both within Australia and internationally is based upon the CLI. The CLI is complex and provides the user with detailed information about the device under investigation. Deriving useful information has been found to be problematic. Furthermore, this problem is exacerbated during teaching when it is necessary to integrate all of the information from a number of different CLI commands. It was also found, for all the curriculum analyzed, that routers and switches are usually treated as 'black boxes'. This is contrary to educational theory that supports the need for a conceptual model as the foundation of curriculum design.

Attempts have been made to further develop the B-Nodes into state models, which can be used in promoting students' learning of computer and network technology. Furthermore, these models can be used to model various protocols e.g. Spanning Tree Protocol (STP), Open Shortest Path First (OSPF), etc.

A possible major contribution to new knowledge of the published papers relates to the integration of state models of switches and routers with students' learning.

Within education it is well documented that after successfully completing an examination it is not uncommon for the majority of students to demonstrate very poor retention of not only factual information but also concepts. Students whose learning was based upon state models clearly demonstrated that they had internalized the model. This conclusion is made for two reasons: the students' model was not an exact copy of the state model provided in class; and the students demonstrated an understanding of the concept they had been taught. However, further work is needed.

The state models appear to provide richer understanding of device operations and hence provide better teaching models for teaching both undergraduate and postgraduate units in computer and network technology. Furthermore, state models may also provide an improved way of troubleshooting complex routing protocols and hence provide a better way of managing routers and switches.

8. REFERENCES

Abelman, C. (2000). A parallel Universe. *Change, 2000, 32*(3), 20-29.

ACM68. (1968). ACM Curriculum Committee on Computer Science. Curriculum 68: Recommendations for the undergraduate program in computer science. *Communications of the ACM* (March 1968).

ACM78. (1979). ACM Curriculum Committee on Computer Science. Curriculum 78: Recommendations for the undergraduate program in computer science. *Communications of the ACM* (March 1979).

ACM2004. (2004). Joint Task Force for Computing Curricula 2004.

ACM. (1991). Computing Curricula 1991. *Communications of the ACM, 43*, 69 - 84.

ACM. (2001). *Computing curricula 2001 ACM/IEEE Joint Task Force Computer Science Final Report*: ACM Press.

ACM/IEEE. (2001). Computing curricula 2001 ACM/IEEE Joint Task Force Computer Science Final Report: ACM.

Aljareh, S., & Rossiter, N. (2002). *A task-based security model to facilitate colloboration in trusted multi-agency networks.* Paper presented at the Proceedings of the 2002 ACM symposium on Applied computing, Madrid, Spain.

Almedia, V., Bestravos, A., Crovella, M., & Oliveira, A. (1996, December). *Characterization reference locality in the WWW.* Paper presented at the IEEE-ACM PDIS.

Almeida, J. M., Virgilio Almeida, & Yates, D. J. (1997). WebMonitor: a Tool for Measuring World-Wide Web Server Performance.

Arlitt, M., & Williamson, C. (1996, May 1996). *Web server Workload Characterization.* Paper presented at the ACM SIGMETRICS Conference of Measurement of Computer Systems, Philadelphia, PA.

Armitage, C. (1995). 'Irrelevant Degree Factories' must change with times. *The Australian*, 23.

Ashworth, C., & Goodland, M. (1990). *SSADM: A practical Approach*. London: McGraw-Hill.

Bailey, D. H. (1999). Challenges of future High-End Computing.

Bailey, J., & Bakos, Y. (1997). An Exploratory study of the emerging role of electronic intermediaries. *International Journal Electronic Commerce, 1*(3).

Baker, D. (2002). *Communications and Networking*. Retrieved 30-August-2005, 2002, from http://docs.rinet.ru/UJ11/ch23.htm

Bakos, Y. (1998). The Emerging Role of Electronic Market Places on the Internet. *ACM, 41*(8).

Bandura, A. (1986). *Social Foundations of Thought and Action*: Prentice-Hall Humanities /Social Science.

Barker, P. G., & van Schaik, P. (1999, 1-6 July, 1999). *Mental Models and Their Implications for the Design of Computer-Based Learning Resources.* Paper presented at the Proceedings of the 4th International Conference on Computer-Based Learning in Science (CBLIS '99), University of Twente, Enschede, Netherlands.

Barnett, B. L. (1993). *An Ethernet Performance Simulator for Undergraduate Networking.* Paper presented at the Twenty-Four SIGCSE Technical Symposium on Computer Science Education.

Barnett, R. (2000). *Realizing the University in an age of Super complexity*. Milton Keynes UK: Society for Research into Higher Education and Open University Press.

Bayman, P., & Mayer, R. E. (1984). Instructional manipulation of users' mental models for electronic calculators. *International Journal of Man-Machine Studies, 20*, 189-199.

BCS. (2004, 5 May 2004). *BCS Thought Leadership Debate*. Retrieved 20 Sept 2004, 2004, from http://www.bcs.org/BCS/News/ThoughtLeadership/pastdebates/ITSystems.htm

Beasley, J. (2004). Networking. In (pp. 338). Upper Saddle River, NJ: Pearson Press.

Beeson, C., & Gay, W. (2000). *21st Century Skills Training: Computer Systems Support.* Paper presented at the 2000 ASEE Annual Conference & Exposition, St. Louis, MO.

Ben-Ari, M. (1998). *Constructivism in Computer Science Education.* Paper presented at the Twenty Ninth SIGCSE Technical Symposium on Computer Science Education, Atlanta, GA.

Ben-Ari, M. (2001). Constructivism in Computer Science Education. *Journal of Comuters in Mathematics and Science Teaching, 20(1)*, 45-73.

Benjamin, H. (2002). *CCNP Practical Studies: Routing*. Indianapolis: Cisco Press.

Bhimani, A. (1996). Securing the Commercial Internet. *Communications of the ACM, 39*(6).

Bjork, R. A. (1978). The Updating of Human Memory. *In G.H.Bower (Ed.) The Psychology of Learning and Motivation, 12*.

Boehm, B. W. (1984). *A software development environment for improving productivity*.

Bonald, T., & Proutiere, A. (2003). Insensitive Bandwidth Sharing in Data Networks. *Queueing Systems, 44*(1), 69-100.

Booch, G. (1978). *Object-Oriented Analysis and Design* (2nd ed.). Redwood City, CA: Benjamin/Cummings.

Booch, G., James Rumbaugh, & Jacobson, I. (1999). *The Unified Modeling Language User Guide*: Addison Wesley longman, Inc.

Breslau, L., Estrin, D., Fall, K., Floyd, S., Heidemann, J., Helmy, A., et al. (2000). Advances in Network Simulation. *IEEE Computer, 33*(5), 59-67.

Bruno, A. (2003). *CCIE Routing and Switching*. Indianaplois IN: Cisco Press.

Burgess, M. (2000). Principles of Network and System Administration. In (pp. 293). Chichester, England: John Wiley & Sons Ltd.

Buzen, J. P. (1978). Operational Analysis: an Alternative to Stochastic Modeling. *Performance of Computer Installations*, 175-194.

Buzen, J. P. (1984). *A Simple Model of Transaction Processing.* Paper presented at the CMG Proceedings.

Buzen, J. P., & Shum, A. N. (1996, December 16 -18). *Beyond Bandwidth: Mainframe Style Capacity Planning for Networks and Windows NT.* Paper presented at the Proceedings of the 1996 Computer Measurement Group Conference., Olando Florida USA.

Cahners (1997). *Causes of Downtime*. Cahners Instat Group. Retrieved March, 2002, from http://www.instat.com/

Caliri, G. V. (2000). *Introduction to Analytical Modeling*. Waltham MA: BMC Software, Inc.

Carroll, J. M., & Mack, R. L. (1992). Metaphor, computing systems and active learning. *International Journal of Man-Machine Studies, 22*, 39-57.

Cassel, L., & Austing, R. (2000). *Computer Networks and Open Systems An Application Development Perspective*. Massachusetts: Jones and Bartlett Publishers.

Cervero, R. M. (1992). Professional practice, learning and continuing education: an integrated perspective. *International Journal of Lifelong Education, 11*(2), 91-101.

Chang, C. K., Cross, J., Hang, L., Mei, H., Mulder, F., Shackelford, R., et al. (2000). *CC 2001: a computing curriculum for the next ten years.* Paper presented at the ICEUT 2000.

Chang, X. (1999). *Network simulations with OPNET.* Paper presented at the Proceedings of the 31st conference on winter simulations: Simulation-a bridge to future, Phoenix, AZ.

Chen, C. (2003). A Constructivist Approach to Teaching: Implications in Teaching Computer Networking. *Information Technology, Learning and Performance, 21*(2).

Chi, M. T. H., & Roscoe, R. D. (2002). The processes and challenges of conceptual change. *In M.Limon & L. Mason (Eds.), Reconsidering conceptual change: Issues in theory and practice*, pp.3-27.

Cisco. (2002). *Cisco Networking Academy Program: Second Year Companion Guide* (2 ed.). IN: Cisco Press.

Cisco. (2003). Securing Cisco IOS Networks. In C. L. Product (Ed.): Cisco Systems, Inc.

Cisco. (2004). *Cisco Networking Academy Program*. Retrieved 18 August, 2004, from http://cisco.netacad.net

Clarke, D., & Pasquale, J. (1996). Strategic Directions in Networks and Telecommunications. *Computing Surveys*, *28*(4), 679 - 690.

Clements, A. (2000). The Undergraduate Curriculum in Computer Architecture. *IEEE Micro*, 13-22.

Cobern, W. (1991). *Contextual Constructivism: The Impact of Culture on the Learning and Teaching of Science.* Paper presented at the The Annual Meeting of the National Association for research in Science Teaching, Lake Geneva WI.

Coe, P. S., Williams, L. M., & Ibett, R. N. (1996). An Interactive Environment for the Teaching of Computer Architecture. *ACM SIGCSE Bulletin, 28*, 33-35.

Coleman, J. (1995). *OSI Reference Model*, from http://www.cs.ntu.edu.au/sit/resources/dc100www/dc2001.htm

Coles, R. (1992). *The Spiritual Life of Children*: Continuum International Publishing Group.

Comer, D. (1999). *Computer Network and Internets*. NJ: Prentice Hall.

Commer, D. E. (2004). *Computer Networks and Internets*. NJ: Prentice Hall, Upper Saddle River.

Confrey, J. (1991). Learning to Listen: A Student's Understanding of Powers of Ten. In E. von Glasersfeld (Ed.), *radical Constructivism in Mathematics Education* (pp. 111 - 138). Dordrecht, Holland.: Kluwer Academic Publishers.

Consulting, A. (1999). *A Review of eCommerce in Australia*, from http://www.ac.com/services/ecommerce/ecom_australia.html

Cooling, J. E. (1991). *Software Design for Real-Time Systems*. Cornwall: Chapman and Hall.

D'Andrade, R. G. (1992). Schemas and motivation. *In R.G.D'Andrade and C.Strauss (Eds.) Human Motives and Cultural Models. Cambridge*.

Davies, N., Ransbottom, S., & Hamilton, D. (1998). Teaching Networks through Modelling. *ACM SIGada Letters, XVIII (5)*, 104-110.

Davies, P. T. (2002). *Securing and Controlling Cisco Routers*. Boca Raton FL: Auerbach Publications.

Davis, N., Ransbotton, S., & Hamilton, D. (1998). Teaching Computer Networks Through Modeling. *Ada Letters, XVII*(Sept/Oct).

Davis, S. A., & Bostrom, R. P. (1993). Training End Users: An Experimental Investigation of the Roles of the Computer Interface and Training Methods. *MIS Quarterly, 17*(1), 61-85.

Dell, M. (1999). Keynote Address at DirectConnect Conference. Austin,TX

Denning, P., & Buzen, J. P. (1978). The Operational Analysis of Queuing Network Models. *Computing Surveys, 10*(3), 225-261.

Dennis, A. (2002). Networking in the Internet Age. In. NY: John Wiley & Sons, Inc.

Dennis, S., & Hofer, J. (2000). Capacity Planning for e-commerce system with Benchmark Factory.

DETYA. (2000, Thursday, 18 January 2001). *Learning for the knowledge society: An education and training action plan for the information economy*. Retrieved 1/2/2001, 2001, from http://www.detya.gov.au/schools/Publications/reports/learning/learning.htm# The Way Forward

Devlin, B., Gray, J., Laing, B., & Spix, G. (1999). *Scalability Terminology: Farms, Clones, Partitions and Packs: RACS and RAPS,* (No. MS-TR-99-85): Microsoft Research.

DharuKeshwari, J., & Maj, S. P. (2003). Vendor-based network engineering education: an international comparison. *World Transaction on Engineering and Technology Education, 2*(2).

Dixon, M., McGill, T., & Karlsson, J. (1997). *Using a network simulation package to teach the client-server model.* Paper presented at the Proceedings of the conference on Integrating Technology into Computer Science Education.

Doyle, J. (2002). *Routing TCP/IP* (Vol. 1). Indianapolis IN: Cisco Press.

Driver, R., & Bell, B. (1985). *Students' thinking and the learning of science: A constructivist view*: Centre for Studies in Science Education University of Leeds.

Driver, R. H., & Erikson, G. L. (1984). Theories in action: Some theoretical and empirical issues in the study of students' conceptual frameworks in science. *Studies in Science Education, 10,* 37-60.

Duffy, T. M., & Jonassen, D. (1992). *Constructivist and the technology of instruction: A conversation*. NJ: Lawrence Erlbaum Associates.

Eck, W. V.*Electromagnectic Radiation From Video Display Units: An Eavesdropping Risk?*, 1985, from http://web.archive.org/web/20000830130750/www.shmoo.com/tempest/emr.pdf

Edelson, D., Gordin, D., & Pea, R. (1999). Addressing the challenges of inquiry-based learning through technology and curriculum desgin. *The Journal of the Learning Sciences, 8*(3&4), 391-450.

Edmondson, K. M. (1993). *Concept mapping for the development of medical curricula.* Paper presented at the Paper presented at the Annual meeting of the International Communication Association conference, Sydney.

Engel, B., & Maj, S. P. (1999). *Towards Quality of Service on the Internet - an educational case study.* Paper presented at the 3rd Baltic Region Conference on Engineering Education, Goteborg, Sweden.

Fage, D., Agosta, J., Merchant, D., Foltz, R., & Barnes, J. (2000). *CCNP Remote Access Study Guide*. Berkley CA: Osborne/McGraw-Hill.

Fenik, H. (1998). Zona Research. *LAN Times*.

Ferrari, D., Serazzi, G., & Zeigner, A. (1983). *Measurement and Tunning Computer Systems*: Prentice-Hall.

Fetherston, T. (1999). Students Constructs About Energy and Constructivist Learning. *Research in Science Education*, *29*(4), 515-525.

Fetherston, T. (2002). What Information and Communication Technologies courses should be taught at a new Senior College. *Proceedings of the seventh world conference on computers in education conference on Computers in education, 8*.

Floyd, D., & Kohler, E. (2003). Internet Research Needs Better Models. *ACM SIGCOMM Computer Communications Review, 33*(1), 29-34.

Forrester, J. (1971). *Counterintuitive behavior of social system*. Cambridge, MA: Wright-Allen Press, Inc.

Forrester, J. (1994). Policies, decisions, and information sources for modeling. *In Morecroft, J.D.W., and Sterman, J.D. (Eds.), Modeling for Learning Organizations*, 51-84.

Gagne, R. (1985). *The Conditions of Learning* (Vol. 17). New York: Rinehart & Winston.

Gane, C., & Sarson, T. (1979). *Structured Systems Analysis: Tools and Techniques*. Englewood Cliffs, NJ: Prentice-Hall.

Gibb, T. (1988). *Principles of Software Engineering Management*. MA: Wesley, Reading.

Gibson, D. (2002). *Making simple work of Complex functions*. Retrieved 20 June 2004, 2004, from http://www.microconsultants.com/tips/fsm/fsmarticl.pdf

Gilbert, J. (1995, April 18-22). *The role of models and modelling in some narratives in science learning.* Paper presented at the The Annual Meeting of the American Educational Research Association, San Francisco, CA, USA.

Gilbert, J., & Boulter, C. (1998). Learning science through model and modelling. *In B.Fraser and K.Tobin (eds), International Handbook of Science Education,* 52-66.

Giuli, T. J., & Baker, M. (2003). *Narses: A Scalable Flow-Based Network Simulator*: Standford University.

Goldman, J. E., & Rawles, P. T. (2001). *Applied Data Communications: A business-orientated approach* (3 ed.). New York: John Wiley & Sons.

Goldsworthy, A. W. (1993). IT and the Competency Debate - Skill vs Knowledge A Major Issue. *The Australian Computer Journal, 25*(3), 113-122.

Gomaa, H. (1993). *Software Design Methods for Concurrent and Real-Time Systems*. Reading, MA: Addison-Wesley.

Greenwald, R. (1999). *Oracle Essentials*: O'Reilly.

Grice, M. (2001a). *CCNP Guide to Advanced Cisco Routing*. Boston MA: Thomson Course Technology.

Grice, M. (2001b). *CCNP Guide to Advanced Cisco Routing*: Thomson Learning.

Gupta, J. N. D., & Wachter, R. M. (1998). A Capstone Course in the Information Systems Curriculum. *International Journal of Information Management, 18*(6), 427-441.

Gutierrez, J. A. (1998). A connectionless approach to integrated network management. *International Journal of Network Management, 8*(4).

Haar, S. V. (1999). *Web Metrics: Go Figure*. Retrieved July, 2003, from http://www.cio.com/

Halsal, F. (1996). *Data Communications, Computer Networks and Open Systems*. Harlow, England: Addison-Wesley.

Havard, M., Hughes, M., & Clarke, J. (1998). The Introduction and Evaluation of Key Skills in Undergraduate Courses. *Journal of Further and Higher Education, 1*, 61-68.

Hawryszkiewycz, I. (2001). *Introduction to Systems Analysis & Design*. Malaysia: Prentice Hall.

Heap, G., & Maynes, L. (2002). *CCNA Practical Studies*. Indianapolis: Cisco Press.

Hennessy, J. L., & Patterson, D. A. (1996). *Computer Architecture A Quantitative Approach* (Second Edition ed.). San Francisco, CA: Morgon Kaufmann Publishers, Inc.

Heuring, V. P., & Jordan, H. F. (1997). *Computer Systems Design and Architecture*. Menlo Park CA: Addison Wesley.

Hiebert, J., & Carpenter, T. P. (1992). Learning and teaching with understanding. *In D.A. Grouws (Ed.)*(Handbook of research on mathematics teaching and learning), 65-126.

Hudson, K., Caudle, K., & Cannon, K. (2000). *CCNA Guide to Cisco Networking* (2nd ed.): Thomson Course Technology. Boston, MA.

Hudson, K., Caudle, K., & Cannon, K. (2003). *CCNA Guide to Cisco Networking* (2 ed.). Boston, MA: Thomson Course Technology.

Hughes, J. (2003). *Accreditation of Courses in Universities at the Professional Level: Guidelines for Universities* (No. ISBN 0909925 64 X). Sydney: Australian Computer Society.

Humphrey, W. S. (1990). *Managing the software process*. MA: Addison-Wesley, Reading MA.

Hung, C. K., & Kamalakar, K. (2003). *A secure workflow model*. Paper presented at the Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003, Adelaide, Australia.

Huy, V. V., & Chae, B. (2004). A Framework for MIS Curriculum Interdisciplinary: A Vietnamese University Case. *EJISDC, 16*(4), 1-12.

Hyde, D. C. (2000). Teaching Design in a Computer Architecture Course. *IEEE Journal*(May/June), 23-28.

IEEE. (1983). *IEEE Standard Glossary of Software Engineering Terminology*. New York, NY: The Institute of Electrical and Electronic Engineers.

IEEE/ACM. (2001). *CC-2001 Task Force, Computing Curriculum (Final Draft),* from http://www.computer.org/education/cc2001/final

Jackson, D., & Chapin, J. (2000). Redesigning air traffic control: An exercise in software design. *IEEE Software, 17*(3).

Jackson, M. (1975). *Principles of Program Design*. New York: Academic Press.

Jackson, M. (1983). *System Development*. Englewood Cliffs, NJ: Prentice-Hall.

Jakob, N. (2000). *Designing Web Usability, The Practice of Simplicity*. IN: New Riders Publishing.

Jeenings, D. M., Landweber, L. H., Fuchs, I. H., Farber, D. J., & Adrion, W. R. (1986). Computer Networking for Scientists. *Science, 231*, 943-950.

Jonassen, D. H. (1991). Evaluating Constructivist Learning. *Educational Technology, 31*(9).

Johnson-Laird, P. N. (1983). Mental models: Towards a cognitive science of language, inference, and consciousness. Cambridge, MA: Harvard University Press.

Julyan, C., & Duckworth, E. (1996). *A constructivist perspective on teaching and learning science*. New York: Teacher College Press.

Kain, J. B. (1999). The corporate benefits of a distributed component architecture Distributed Computing Architecture /e-Business Advisory Service Executive Report. *2*(11), 1-29.

Kargl, F., Maier, J., & Weber, M. (2001). *Protecting web servers from distributed denail of service attacks.* Paper presented at the tenth international conference on World Wide Web, Hong Kong.

Karmel, P. (1995). Education and the Economic Paradigm in Confusion Worse Confounded - Australian Education in the 1990s. *Academy of Social Sciences in Australia - Occasional Paper Series, 1*.

Keitel, C. (1992). *Is there a real Chance for "Mathematics and Science for All Americans"?* Paper presented at the Proceedings of the University of Chicago School Mathematics Project International Conference on Mathematics Education., Chicago, USA.

Khan, A., Kadir, K., Shaikh, A., & Kidwai, R. (2001). *Impact of Influx of computer science graduates from developing countries to Europe and North America.* Paper presented at the 31st ASEE/IEEE Frontiers in Education Conference, Reno, NV.

Kheir, N. A. (1996). Motivation and overview. *In N.A. Kheir (Ed.), System modeling and computer simulation*, 3-26.

Klienrock, L. (1975). *Queuing Systems Vol I Theory.* NY.: John Wiley and Sons.

Kneale, B., Horta, A. Y., & Box, I. (2004). *VELNET (Virtual Environment for Learning Networking).* Paper presented at the 6th Australasian Computing Education Conference, Dunedin, NZ.

Kocher, P., Lee, D., McGraw, G., & Rajhunathan, A. (2004). *Security as a new dimension in embedded system design.* Paper presented at the 41st annual conference on design automation, San Diego, CA, USA.

Kohler, E., Morris, R., Chen, B., Jannotti, J., & Frans Kaashoek, M. (2000). The Click Modular Router. *ACM Transactions on Computer Systems, 18*(3), 263-297.

Kohli, G., Maj, S. P., Murphy, G., & Veal, D. (2004). *Abstraction in Computer Network Education: A model based approach.* Paper presented at the ASEE, Salt Lake City, UT.

Kohli, G., Maj, S. P., & Veal, D. (2000, June). *Multi-media technology - an opportunity for modern engineering education.* Paper presented at the American Society for Engineering Education (ASEE) Conference, St. Louis, MO.

Kuhn, T. (1970). *The Structure of Scientific Revolutions*. Chicago: University of Chicago Press.

Kurose, J., Liebeherr, J., Ostermann, S., & Ott-Boisseau, T. (2002). *ACM SIGCOMM Workshop on Computer Networking: Curriculum Designs and Education Challenges*.

Kurose, J., & Ross, K. (2000). *Computer Networking A Top-Down Approach Featuring the Internet*. NY: Addision-Wesley.

Kurose, J. F., & Ross, K. W. (2004). *Computer Networking: A top-down approach featuring the Internet*: Pearson, Addison Wesley.

Lai, K., & Baker, M. (2000, August 2000). *Measuring link bandwidths using a deterministic model of packet delay.* Paper presented at the ACM SIGCOMM 2000 Conference, Stockholm, Sweden.

Lammie, T. (1999). *Cisco Certified Network Associate Study Guide* (2nd ed.). San Francisco, CA: Sybex.

Lammie, T., Tim, C., & Odom, S. (2002). *CCNP/CCIP BSCI Study Guide*. San Francisco, CA: Sybex.

Lankewicz, L. B. (1998). *Resources for Teaching Computer Networks.* Paper presented at the Twenty-ninth SICSE Technical Symposium of Computer Science Education, Atlanta, GA.

Laurillard, D. (2002). Rethinking teaching for the Knowledge society. *Education Review* (January/February).

Laver, P. (1994). *Overview of values: An Industrial Perspective In Training Minds: How well do Employers and Educators align their values and responsibilities?* Paper presented at the Proceedings of the Industry - Education Forum, Perth Australia.

Lazowska, E. D., John Zahorjan, Scott, G., & Graham, K. C. S. (1984). *Quantitative System Performance*. New Jersey: Prentice-Hall.

Lea, C. L., Tsui, C., Li, B., Kwan, L. C., Chan, S. M., & Chan, A. H. W. (1999). A/I Net: A Network that Integrates ATM and IP. *IEEE Network, January/February*, 48-55.

Lehrer, R., Horwath, J., & Schuable, L. (1994). Developing model-based reasoning. *Interactive Learning Environments, 4*(3), 219-231.

Lewis, W. (2003). *CCNP Cisco Networking Academy Program: Multilayer Switching Guide*. Indianapolis IN: Cisco Press.

Li, K., Shor, M. H., Walpole, J., & Pu, C. (2001). *Modeling the Transient Rate Behavior of Bandwidth sharing as a Hybrid Control System*: Oregan Graduate Institute Technical Report.

Lilja, D. J. (2000). *Measuring Computer Performance - A Practitioner's Guide*. Cambridge: Cambridge University Press.

Linn, M. C., & Muilenberg, L. (1996). Creating Lifelong Science Learners: What Models form a firm Foundation. *Educational Researcher, 25*(5), 18-24.

Little, J. (1961). A proof of the queuing formula L=YW. *Operations Research, 9*, 383-387.

Little, J. D. C. (1961). A proof of the queuing formula L = Lambda W. *Operations Research, 9*, 383 - 387.

Lovett, M. C., & Greenhouse, J. B. (2000). Applying cognitive theory to statistical instruction. *The American Statistician, 54*(3).

Maj, S. P. (2000). *Modeling Computer and Network Systems - a new 'soft' technique for soft systems analysis.* Paper presented at the Teaching and Learning Forum 2000, Curtin University of Technology, Perth.

Maj, S. P., Fetherston, T., Charlesworth, P., & Robbins, G. (1998). *Computer & Network Infrastructure Design, Installation, Maintenance and Management - a proposed new competency based curriculum.* Paper presented at the The Proceedings of the Third Australasian Conference on Computer Science Education, The University of Queensland, Brisbane, Australia.

Maj, S. P., & Kohli, G. (2002, May 27-28). *B-Node: A bridge between the business model, information model and infrastructure model.* Paper presented at the Seventh Caise/IFIP-WG8.1 International Workshop on Evaluation of Modeling Methods in System Analysis and Design, Toronto.

Maj, S. P., & Kohli, G. (2004, June 25-28, 2004). *A New State Model for Internetworks Technology.* Paper presented at the Informing Science + Information Technology Education Joint Conference, Rockhampton, QLD Australia.

Maj, S. P., & Kohli, G. (2004). A New State Model for Internetworks Technology. *Journal of Information Technology Education, 1*.

Maj, S. P., Kohli, G., & Fetherston, T. (2005). *A Pedagogical evaluation of new state model diagram for teaching internetworking technologies.* Paper presented at the 28th Australasian Computer Science Conference, Newcastle, Australia.

Maj, S. P., Murphy, G., & Kohli, G. (2004, 20-23rd Oct, 2004). *State Models for Internetworking Technologies.* Paper presented at the 34th ASEE/IEEE Frontiers in Education Conference, Savannah, GA.

Maj, S. P., Robbins, G., Shaw, D., & Duley, K. W. (1996). Computer and Network Installation, Maintenance and Management - A Proposed New Curriculum for Undergraduates and Postgraduates. *The Australian Computer Journal, 30*(3), 111-119.

Maj, S. P., & Veal, D. (2000a). Is Computer Technology Taught Upside Down? *5th Annual SIGCSE/SIGCUE conference on Innovation and Technology in computer Science Education Helsinki, Finland, ACM.*

Maj, S. P., & Veal, D. (2000b). Computer Technology Curriculum - A New Paradigm for a New Century. *Journal of Research and Practice in Information Technology, 32*(August/September), 200-214.

Maj, S. P., & Veal, D. (2001). *Controlling Complexity in Information Technology: System and Solutions.* Paper presented at the International Association of Science and Technology for Development (IASTED) conference on Computers and Advanced Technology in Education (CATE 2001), Banff, Canada.

Maj, S. P., Veal, D., & Boyanich, A. (2001a) *B-Nodes - A New Architectural Abstraction model for engineering students. Paper presented at the 4th UICEE Annual Engineering Education Conference, Bangkok, Thailand.*

Maj, S. P., Veal, D., & Boyanich, A. (2001b) *A New Method for controlling Information Technology Hardware Complexity.* Paper presented at the American Conference on Information Systems, Boston, MA.

Mandla, E. (2004). Pieces of Paper. *PC Authority,* 14-16.

Marca, D., & McGowan, C. (1988). *SADT: Structured Analysis and Design Technique*. New York: McGraw-Hill.

Marti, W. F., & Hamilton, J. J. A. (1996). *Packet tracing exercises.* Paper presented at the In Workshop at the 1996 Special Interest Group on Computer Science Education, Philadelphia.

Marti, W. F., Pooch, U. W., & Hamilton, J. A. (1996). *Packet Tracing: a new paradigm for teaching computer networking courses.* Paper presented at the Integrating Technology into Computing Science Education, Barcelona, Spain.

Mayer. (1982). *Putting General Education to Work: The Key Competencies Report.* . Melbourne: AEC/MOVEET.

McComas, B. (2001, 2/4/2001). *Can the Pentium 4 Recover?* Retrieved 19th April, 2001, from http://www.inqst.com/articles/p4bandwidth/p4bandwidthmain.htm

McDonald, M., Rickman, J., McDonald, G., Heeler, P., & Hawley, D. (2001). Practical experiences for undergraduates computer networking students. *JCSE, 16*(3), 261-270.

McGregor, M. (2001). *Semester Five Companion Guide, Advanced Routing*. Indianapolis: Cisco Press.

McQuerry, S. (2000). *Interconnecting Cisco Network Devices*. IN: Cisco Press.

Menasce, D. A., & Alemeida, A. F. (2002). *Capacity Planning for Web Services. Metrics, Models, and Methods.* Upper Saddle River NJ: Prentice Hall.

Menasce, D. A., Almeida, V. A. F., Fonseca, R. C., & Mendes, M. A. (1999). *A Methodology for Workload Characterization for E-Commerce Servers.* Paper presented at the ACM Conference in Electronic Commerce, Denver, CO.

Menasce, D. A., & Almendia, V. A. F. (1999). A methodology for workload characterization for e-commerce server. *ACM conference in Electronic Commerce, Denver, CO, ACM*.

Menasce, D. A., Almendia, V. A. F., & Dowdy, L. W. (1994). *Capacity Planning and Performance Modeling: From Mainframes to Client-Server Systems*. Upper Saddle Rive, NJ: Prentice Hall.

Menasce, D. A., Almendia, V. A. F., & L.Dowly. (1994). *Capacity Planning and Performance Modeling*. NJ: Prentice Hall.

Menasce, D. A., Daniel Barbara, & Dodge, R. (2001). *Preserving QoS of E-commerce Sites Through Self-Tuning: A Performance Model Approach.* Paper presented at the 3rd ACM conference on Electronic Commerce, Tampa, FL.

Menasce, D. A., Virgilio, A. F., & Almeida. (2000). *Scaling for E-Business: Technologies, Models, Performance, and Capacity Planning*.

Merriam, S., & Caffarella. (1991, 1998). *Learning in Adulthood. A comprehensive guide*. San Franscisco, CA: Jossey-Bass.

Merrill, M. D., Li, Z., & Jones, M. (1991). Instructional Transaction Theory: An Introduction. *Educational Technology, 31*(6), 7-12.

Milrad, M., Spector, M., & Davidsen, P. (2002). Model facilitated learning. In *Learning and teaching with technology: Principles and practices*. London, UK: Kogan Page Publishers.

Minch, R., & Tabor, S. (2003). Networking Education for the New Economy. *Journal of Information Technology Education, 2*.

Mirabile, R. J. (1997, August). Everything you wanted to know about competency modeling. *Training and Development, 51*, 73 -78.

Moallem, M. (2001). Applying Constructivist and Objectivist learning theories in the design of a web-based course: Implications for practice. *Educational Technology & Society, 4*(3).

Mogul, J. C. (1995, May). *Operating system support for busy Internet servers.* Paper presented at the Proceedings of the fifth workshop on Hot Topics in Operating Systems.

Moran, T. P. (1981). The command language grammar: a representation for the user interface of interactive computer systems. *International Journal of Man-Machine Studies, 15*, 3-50.

Morgan, M., & Dennis, C. (2001). *CCNP Remote Access Exam Certification Guide*. Indianapolis IN: Cisco Press.

Morris, C. (1999). *White Paper: The cost of security on Cisco Routers*. Retrieved Wednesday 20 October 1999, from http://www.computerweekly.com/Article42115.htm

Mudge, T. (1996). Strategic directions in computer architecture. *ACM Computing Surveys, 28*(4), 671-678.

Mueller, S. (2002). *Scott Mueller's Upgrading and Reparing PCs* (13 ed.). Indianapolis, IN: QUE.

Murphy, G., Kohli, G., Veal, D., & Maj, S. P. (2004). *An Examination of Vendor-Based Curricula in Higher and Further Education in Western Australia.*

Paper presented at the American Society for Engineering Education (ASEE), Salt Lake City, UT.

Murphy, K. (1998). *Western Australian Upper Secondary Schooling Scince the 1960s.* Murdoch, Perth, WA.

Nakagawa, Y., Suda, H., Masahiro, U., & Mida, Y. (2003, November 5-8). *An Innovative Hands-on Laboratory for teaching a networking course.* Paper presented at the 33rd ASEE/IEEE Frontiers in Education Conference, Boulder, CO.

Naraymann, R., & Neethi, S. (2001). *Building Software Engineering Profesionals: TCS Experience.* Paper presented at the Proceedings of the 14th Conference on Software Engineering Education and Training: "in search of a software engineering profession", Charlotte NC.

Naumann, J. D., & Jenkins, A. M. (1982). *Prototyping: the new paradigm for system development*: MIS Quarterly.

NEEB. (1995). *Demand for and Dimensions of Education and Training* (Commissioned No. 39). Canberra: National Board of Employment Education and Training (NBEET).

Neufeld, M., Jain, A., & Grunwald, D. (2002). *Nsclick: Bridging Network Simulation and Deployment.* Paper presented at the International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems, Atlanta, Georgia.

Nieh, J., & Leonard, O. C. (2000). Examining VMware. *Dr Dobb's Journal*(August 2000).

Nielsen, J. (2000). *The Usability Life Cycle*, from http://www.developer.ibm.com/library/articles/nielsen3.html

Norman, D. R. (1983). Some observations on mental models. *In D.Gentner and A.Stevens (eds), Mental Models.*

Norman, D. R. (1993). *Things that make us smart*: Perseus Books.

Novak, J. D. (1991). Clarify with concepts maps: A tool for students and teachers alike. *The Science teacher, 58*(7), 45-49.

Novak, J. D. (1993). How do we learn our lesson?: Taking students through the process. *The Science Teacher*, *60*(3), 50-55.

Nwana, H. S. (1997). Is Computer Science Education in Crisis. *ACM Computing Surveys, 29*(4), 322-324.

Odom, W. (2000). *Cisco CCNA Exam #640-507 Certification Guide*. Indianapolis, IN: Cisco Press.

Oppenheimer, P. (2001). *Top-down Network Design: A systems analysis approach to enterprise network design*. Indianapolis IN: Cisco Press.

O'Reilly Search. Retrieved 07-July-2004, 2004

Osterich, H. (2000). *CCNA 2.0 Prep Kit Routing and Switching*. Indianapolis IN: QUE.

Padjen, R., Lammie, T., & Edwards, W. (2002). *CCNP Remote Access Study Guide* (2 ed.). San Francisco CA.: SYBEX.

Pang, R., & Paxson, V. (2003). *A High-level Programming Environment for Packet Trace Anonymization and Transformation.* Paper presented at the SIGCOMM'03, Karlsruhe.

Paquet, C., & Teare, D. (2001). *Building Scalable Cisco Networks*. IN: Cisco Press.

Parker, B. C., & Drexel, P. G. (1996). A System-Based Sequence of Closed Labs for Computer Systems Organization. *ACM SIGCSE Bulletin, 28*(1), 53-57.

Pattinson, C., & Dacre, T. (1998). *A network model for network management teaching.* Paper presented at the Proceedings of the Third Australasian Conference on Computer Science Education.

Peters, J. F., & Pedrycz, W. (2000). *Software Engineering - An Engineering Approach*. New York: John Wiley &Sons.

Pfleeger, S. L. (2001). *Software Engineering theory and practice*. Upper Saddle, River, NJ: Prentice-Hall, Inc.

Phillip, M. (1997). *Successful e-Commerce*. Melbourne, Australia: Bookman.

Phye, G. D. (1997). *Learning and remembering: The basis for personal knowledge construction.* (Handbook of academic learning: Construction of Knowledge ed.). San Diego, CA: Academic Press.

Piaget, J. (1970). *Piaget's theory* (3rd ed.). New York: Wiley.

Pignataro, C., Kazemi, R., & Dry, B. (2003). *Cisco Multiservice Swtiching Networks*. IN: Cisco Press.

Pitts, J. M. (1995). Mental Models of Information: The 1993-94 AASL/Highsmith Research Award Study. *SLMR, 23*(3).

Pooch, U. W., & Wall, J. A. (1993). *Discrete event simulation: A practical approach*. Boca Raton, FL: CRC Press.

Posner, G. J., Strike, K. A., Hewson, P. W., & Gertzog, W. A. (1982). Accomodation of a scientific conception. Toward a theory of conceptual change. *Scince Education, 66*(2), 211-227.

Pressman, R. S. (1992). *Software Engineering A practitioner's Approach*. Singapore: McGraw-HILL International Editions.

Pressman, R. S. (2000). *Software Engineering A practitioner's approach*. New York: McGraw-Hill.

Quarterman, J. S., & Hoskins, J. C. (1986). Notable Computer Networks. *Communication ACM, 29*, 932-971.

Quinton, R. (1993). *An Introduction to Socket Programming, Computing and Communication Services*. Ontario: The University of Western Ontario.

Ramsden, P. (1992). *Learning to teach in higher education*. London: Rutledge.

Reed, D. (2003). *Applying the OSI Seven Layer Network Model To Information Security*: SANS Institute.

Reeder, E. (2002). *Measuring What Counts: Memorization Versus Understanding*. Retrieved 20th July, 2004

Reid, I. (1996). *Higher Education or Education for Hire? - Language and Values in Australian Universities*. Rockhampton, Queensland: Central Queensland University Press.

Review, C. (1996, May). Educating the Workforce for the New Millennium. *Campus Review*, 1-7.

Robbins, M., Schagaev, I., & Yip, J. (2003). *Effective Teaching in Theoretical IT modules.* Paper presented at the 33rd ASEE/IEEE Frontiers in Education Conference, Boulder, CO.

Rockstrom, A., & Saracco, R. (1982). SDL-CCITT specification and description language. *IEEE Transaction on Communications, 30 (6)*, 1310-1318.

Rouse, W. B., & Morris, M. M. (1986). On Looking into the Black Box: Prospects and Limits in the search of Mental Models. *Psychological Bulletin, 10*, 349-363.

Royce, W. W. (1970). *Managing the development of large software system: Concepts and Techniques*: Wescon.

Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., & Lorensen, W. (1991). *Object-Oriented Modeling and Design*. NJ: Prentice-Hall.

Maj, S.P. & Kohli, G. (2004). New State Models for Internetwork Technology. *Issues in Informing Science and Information Technology Education*.

Sandhu, R. (1992). *A lattice interpretation of the Chinese wall security policy*. Paper presented at the Proc. 15th NIST-NCSE National Computer Security Conference.

Sanjaya, K., Chirag, N., & John, G. (2000). A Benchmarking suite for evaluating configurable computing systems- Status, Reflections, and Future Directions. *Eighth International Symposium on Field- Programmable Gate Arrays, Monterey, CA*.

Schneider, G. P., & Perry, J. T. (2000). *Electronic Commerce*: Course Technology.

Searles, D. E. (1993). An Integrated Hardware Simulator. *ACM SIGCSE Bulletin, 25*(2), 24-28.

Security System Module. (2004). Retrieved 14-July-2004, from http://www.nsfocus.com/english/homepage/solutions/system.htm

Sevcik, K., & Mitrani, I. (1981). The Distribution of Queuing Network States at Input and Output Instants. *J.ACM, 28*(2).

Shaughnessy, T., & Velte, T. J. (2000). *Cisco, A Beginner's Guide*. New York: Osbourne/McGraw-Hill.

Shaw, M. (1984). Abstraction Techniques in Modern Programming Languages. *IEEE Software, 1*(4), 10-26.

Shinder, D. L. (2001). *Computer Network Essentials*. Indianapolis: Cisco Press.

Shklar, G. (1998). *The New York Times*.

Skadron, K., Martonosi, M., August, D. I., D.Hill, M., Lilja, D. J., & Pai, V. S. (2003). Challenges in Computer Architecture Evaluation. *Computer, 36*, 30-36.

Skinner, B. F. (1968). *The Technology of Teaching*. New York: Appletion-Century-Crofts.

Smith, P. L., & Ragan, T. R. (1999). *Instructional Design*: Prentice-Hall, Inc.

Smith, R. (2001). The computer networks course: will it all fit? *Journal of Computing Science in Colleges, 17*(2).

Smith, R., & Francia, G. (2000). *A Hybrid Computer Networks Course for CS and IS Majors*. Paper presented at the Proceedings of ISECON 2000, Philadelphia, PA.

Smith, W. D. (2000). *TPC-W: Benchmarking An E-commerce Solution*, from http://www.tpc.org

Srimani, P., Soldon, D. L., Impagliazo, J., Hughes, J., & Nelson, V. P. (2002). *Panel on computing curricula: Computer Engineering.* Paper presented at the 32nd ASEE/IEEE Frontiers in Education Conference, Boston, MA.

Staggers, N., & Forrcio, A. (1993). Mental models: Concepts for human-computer interaction research. *International Journal of Man-Machine Studies, 38*(4), 587-605.

Stein, A., & Criag, A. (2000). *The Dot.Com generation: IT Practices & Skills of Transition Students.* Paper presented at the Proceedings of the on Australaasian Computing Education Conference.

Stewart, J. H. (2001, 25- 28 March). *Inspired Inquiry!: Science Teaching and learning with MUSE.* Paper presented at the National Association for Research in Science Teaching annual meeting, St. Louis.

Stohr, E. A., & Kim, Y. (1998a). A Model for Performance Evaluation of Interactive Systems. *Proc. 31st. Annual Hawaii International Conference on System Sciences, 4: Internet and the Digital Economy*.

Stohr, E. A., & Kim, Y. (1998b, Jan 6-9). *A Model for Performance Evaluation of Interactive Systems.* Paper presented at the 31st. Annual Hawaii International Conference on System Sciences.

Systems, C. (1999). Internetworking Technologies Handbook. In (pp. Chap.51 p.51). Upper Saddle River, NJ: Cisco Press.

Systems, C. (2001a). *Cisco Network Academy Program: First Year Companion Guide* (2nd ed.). Indianapolis, IN: Cisco Press.

Systems, C. (2001b). *Internetworking Technologies Handbook*. IN: Cisco Press.

Systems, C. (2003). Securing Cisco IOS Networks. In (Vol. 1.0).

Tanenbaum, A. S. (2003). *Computer Networks* (4th edition ed.). NJ: Pearson Education International.

Tarrant, J. (2000). What is wrong with Competence? *Journal of Further and Higher Education, 24*(1), 77 - 83.

Thomas II, T. M., Aelmans, A., Houniet, F., & T, N.-K. (2000). *Building Scalable Cisco Networks*. Indianapolis: Cisco Press.

Thomas, R. (2000, 28-29 September). *Fewer Words, clearer picture: The value of a diagrammatic representation of ideas.* Paper presented at the Third Biennial

Communication skills in University education (CSUE), Fremantle, WA, Australia.

Toll, W. E. (1995). Socket Programming in the Data Communications Laboratory. *ACM SIGCSE Bulletin, 27*(1), 39-43.

Tranter, W., Shanmugan, K. S., Rappaport, T. S., & Kosbar.K.L. (2004). *Principles of Communication Systems Simulation with Wireless Application*: Prentice Hall Communications Engineering and Emerging Technologies Series.

Trowbridge, J. E., & Wandersee, J. H. (1994). Identifying critical junctures in learning a college course on evolution. *Journal of Research in Science Teaching, 31*, 459-473.

Tucker, A. B., Barnes, B. H., Aiken, R. M., Barker, K., Bruce, K. B., Cain, J. T., et al. (1991). Summary of the ACM/IEE-CS Joint Curriculum Task Force Report: Computing Curricula '91. *Communications of the ACM, 34*(6), 68-84.

Unknown.*National Institute of Science and Technology*, from http://www.nist.gov/dads/html/finitestatemachine.html

Veal, D. (2004). *An Investigation into Computer and Network Curricula.* Edith Cowan University, Perth, WA, Australia.

Veal, D., Engel, A., & Maj, S. P. (2001). *Multiple Choice Questions on a Computer Installation and Maintenance Unit.* Paper presented at the UICEE 5th Baltic Region Seminar on Engineering Education, Gdynia, Poland.

Veal, D., Maj, S. P., Fetherston, T., & Kohli, G. (1999). *Competency Based Assessment Techniques for use in a Computer Installation & Maintenance Unit.* Paper presented at the 3rd Baltic Region Seminar on Engineering Education, Goteborg, Sweden.

Veal, D., Maj, S. P., & Swan, G. (2000). *Physics: Implications for Computer Technology Education.* Paper presented at the American Society for Engineering Education (ASEE) Annual Conference, St Louis, Missouri.

Velte, T. J., & Velte, A. T. (2001). *Cisco, A Beginner's Guide* (2nd ed.). New York: Osbourne/McGraw-Hill.

von Glasersfeld, E. (1989). Cognition, Construction of Knowledge, and Teaching. *Synthese*(80), 121-140.

von Glasersfeld, E. (1992, Feburary 19-23). *Sensory Experience, Abstraction, and Teaching.* Paper presented at the Alternative Epistemologies in Education Conference, University of Georgia. GA.

von Glasersfeld, E., & Steffe, L. P. (1991). Conceptual models in educational research and practice. *Journal of Educational Thought, 25*(2), 91 - 103.

Vygotsky, L. S. (1978). *Mind in society: the development of higher psychology processes.* Cambridge, MA:: Harvard University Press.

Waters, J., Rees, M. J., & Coe, J. T. (2000). *CCNA Routing and Switching* (2 ed.). Scotsdale Arizona: Coriolis.

West, L., & Bogumil, W. (2001). Immigration and the global IT work force. *Communication ACM, 44*(7), 34-38.

Wheatley, G. H. (1991). Constructivist Perpectives on Science and Mathematics Learning. *Science Education, 75*(1), 9 - 21.

Wichmann, A., Gottdenker, J., & Jonassen, D. (2003). *Developing a framework for conceptual change within scientific inquiry.* Paper presented at the The 3rd IEEE International Conference on Advanced Learning Technologies (ICALT'03).

Williams, R. (2003). *Preparing for TCP/IP: The Role of CSA.* Paper presented at the 33rd ASEE/IEEE Frontiers in Education Conference, Boulder, CO.

Wilson, B. G. (1992). *Higher Education.* Paper presented at the Higher Education and the Competency Movement, ACT, Australia.

Woolfolk, A. E. (1998). *Educational Psychology* (2ed ed.). Boston. MA. Allyn and Bacon.

Yurcik, W. (2001). *Different Approaches in the Teaching of Information Systems Security*. Paper presented at the Information Systems Educational Conference (ISECON), Cincinnati.

Zamg. (2004). *Conceptual Models - Defination*. Retrieved 12 April 2004, 2004, from http://www.zamg.ac.at/docu/Manual/SatManu/Basic/CM/CM.htm

Zinin, A. (2002). *Cisco IP Routing*. Boston: Addison Wesley.

Zipf, G. (1949). *Human Behavior and the Principle of least efforts*. Cambridge, MA: Addision-Wesley.

9. APPENDIX - A

The matrices presented in Appendix - A indicates the relative inputs of the contributing authors to the multiple author publication considered. Minor indicates a substantial contribution within this category was not performed by this author, whereas major indicates that a substantial contribution was made by the author in this category. Equal denotes an equal contribution in that category by the contributing authors.

A1: E-commerce curriculum – is it secure 4th Baltic region seminar on engineering education

Authors: J.Yee, S. P. Maj, G.Kohli (2000). Presented at 4th Baltic Seminar on Engineering Education, Denmark

	Contributions (Major/equal/minor)	Author 1 Yee	Author 2 Maj	Author 3 Kohli
1.	Problem definition	Minor	Major	Major
2.	Literature review	Equal	Major	Equal
3.	Hypothesis formulation	Minor	Major	Minor
4.	Experimental design	Minor	Major	Minor
5.	Establish experimental platform (e.g. design and implement hardware/software)	Major	Major	Major
6.	Data collection	Major	Minor	Major
7.	Data interpretation	Minor	Major	Major
8.	Critical analysis	Minor	Major	Major
9.	Drafting of paper	Minor	Major	Minor
10.	Revisions of final paper	Minor	Major	Major
11.	Submission of final paper	N/A	Minor	Major
12.	Presentation of paper (conference)	N/A	Major	N/A

A2: B-Nodes: A proposed new technique for database design and implementation

Authors: S. P. Maj, G. Kohli (2001) Presented at 12th Australasian Conference on Information Systems, Coffs Harbour, NSW.

	Contributions (Major/equal/minor)	Author 1 Maj	Author 2 Kohli
1.	Problem definition	Major	Minor
2.	Literature review	Equal	Equal
3.	Hypothesis formulation	Equal	Equal
4.	Experimental design	Equal	Equal
5.	Establish experimental platform (e.g. design and implement hardware/software)	Equal	Equal
6.	Data collection	Equal	Equal
7.	Data interpretation	Equal	Equal
8.	Critical analysis	Equal	Equal
9.	Drafting of paper	Major	Minor
10.	Revisions of final paper	Equal	Equal
11.	Submission of final paper	Minor	Major
12.	Presentation of paper (conference)	N/A	Major

A3: B-Nodes: A bridge between Business model, information model and infrastructure model

Authors: S. P. Maj, G. Kohli (2002) Presented at 7^{th} International workshop on evaluation of modelling method in system analysis and design, Toronto, Canada

	Contributions (Major/equal/minor)	Author 1 Maj	Author 2 Kohli
1.	Problem definition	Major	Minor
2.	Literature review	Equal	Equal
3.	Hypothesis formulation	Equal	Equal
4.	Experimental design	Equal	Equal
5.	Establish experimental platform (e.g. deign and implement hardware/software)	Equal	Equal
6.	Data Collection	Minor	Major
7.	Data interpretation	Equal	Equal
8.	Critical analysis	Equal	Equal
9.	Drafting of paper	Major	Minor
10.	Revisions of final paper	Minor	Major
11.	Submission of final paper	Minor	Major
12.	Presentation of paper (conference)	N/A	Major

A4: Modelling global IT structures using B-Nodes

Authors: S. P. Maj, G. Kohli (2002) Presented at 3rd Annual Global Information Technology Management (GITM) World conference

	Contributions (Major/equal/minor)	Author 1 Maj	Author 2 Kohli
1.	Problem definition	Major	Minor
2.	Literature review	Equal	Equal
3.	Hypothesis formulation	Equal	Equal
4.	Experimental design	Equal	Equal
5.	Establish experimental platform (e.g. deign and implement hardware/software)	Equal	Equal
6.	Data Collection	Minor	Major
7.	Data interpretation	Equal	Equal
8.	Critical analysis	Equal	Equal
9.	Drafting of paper	Major	Minor
10.	Revisions of final paper	Minor	Major
11.	Submission of final paper	Minor	Major
12.	Presentation of paper (conference)	N/A	Major

A5: Modelling web site infrastructure using B-Nodes theory

Authors: G. Kohli, D. Veal, S. P. Maj, (2003) Presented at 9th ANZSYS conference, Melbourne, Australia

	Contributions (Major/equal/minor)	Author 1 Kohli	Author 2 Veal	Author 3 Maj
1.	Problem definition	Major	Minor	Minor
2.	Literature review	Major	Major	Minor
3.	Hypothesis formulation	Major	Minor	Major
4.	Experimental design	Major	Major	Minor
5.	Establish experimental platform (e.g. deign and implement hardware/software)	Major	Major	Minor
6.	Data Collection	Major	Major	Minor
7.	Data interpretation	Major	Major	Minor
8.	Critical analysis	Minor	Major	Minor
9.	Drafting of paper	Major	Minor	Minor
10.	Revisions of final paper	Major	Major	Minor
11.	Submission of final paper	Major	Minor	Minor
12.	Presentation of paper (conference)	Major	N/A	N/A

A6: An examination of vendor based curricula in higher and further education in Western Australia

Authors: G.Murphy, G. Kohli, D.Veal S. P. Maj, (2004) Presented at 4th ASEE annual conference, USA

	Contributions (Major/equal/minor)	Author 1 Murphy	Author 2 Kohli	Author 3 Veal	Author 4 Maj
1.	Problem definition	Major	Major	Minor	Minor
2.	Literature review	Major	Minor	Minor	Minor
3.	Hypothesis formulation	Major	Minor	Major	Major
4.	Experimental design	Major	Major	Minor	Minor
5.	Establish experimental platform (e.g. deign and implement hardware/software)	Major	Major	Minor	Minor
6.	Data Collection	Major	Major	Minor	Minor
7.	Data interpretation	Major	Major	Minor	Minor
8.	Critical analysis	Equal	Equal	Equal	Minor
9.	Drafting of paper	Equal	Equal	Equal	Minor
10.	Revisions of final paper	Minor	Major	Minor	Minor
11.	Submission of final paper	Minor	Major	Minor	Minor
12.	Presentation of paper (conference)	N/A	Major	N/A	N/A

A7:Abstraction in computer network education: A model based approach

Authors: G. Kohli, D. Veal, G. Murphy, S. P. Maj, (2004) Presented at 4th ASEE annual conference, USA

	Contributions (Major/equal/minor)	Author 1 Kohli	Author 2 Veal	Author 3 Murphy	Author 4 Maj
1.	Problem definition	Major	Minor	Minor	Major
2.	Literature review	Major	Minor	Minor	Major
3.	Hypothesis formulation	Major	Minor	Minor	Major
4.	Experimental design	Major	Minor	Minor	Major
5.	Establish experimental platform (e.g. deign and implement hardware/software)	Major	Minor	Minor	Major
6.	Data Collection	Major	Minor	Minor	Major
7.	Data interpretation	Major	Minor	Minor	Major
8.	Critical analysis	Major	Minor	Major	Major
9.	Drafting of paper	Major	Minor	Minor	Major
10.	Revisions of final paper	Major	Minor	Minor	Major
11.	Submission of final paper	Major	N/A	N/A	N/A
12.	Presentation of paper (conference)	Major	N/A	N/A	N/A

A8: A new state model for internetwork technology

Authors: S.P. Maj, G. Kohli (2004) Journal of Information Technology Education

	Contributions (Major/equal/minor)	Author 1 Maj	Author 2 Kohli
1.	Problem definition	Equal	Equal
2.	Literature review	Equal	Equal
3.	Hypothesis formulation	Equal	Equal
4.	Experimental design	Equal	Equal
5.	Establish experimental platform (e.g. deign and implement hardware/software)	Equal	Equal
6.	Data Collection	Equal	Equal
7.	Data interpretation	Equal	Equal
8.	Critical analysis	Equal	Equal
9.	Drafting of paper	Major	Minor
10.	Revisions of final paper	Equal	Equal
11.	Submission of final paper	Equal	Equal
12.	Presentation of paper (conference)	N/A	Major

A9: State Models for Internetworking Technologies

Authors: S. P. Maj G.Murphy, G.kohli (2004) Presented at 34th ASEE / IEEE Frontiers in education conference, USA

	Contributions (Major/equal/minor)	Author 1 Maj	Author 2 Murphy	Author 3 Kohli
1.	Problem definition	Major	Major	Major
2.	Literature review	Major	Minor	Major
3.	Hypothesis formulation	Major	Major	Major
4.	Experimental design	Major	Minor	Major
5.	Establish experimental platform (e.g. deign and implement hardware/software)	Major	Major	Major
6.	Data Collection	Major	Major	Major
7.	Data interpretation	Major	Minor	Major
8.	Critical analysis	Major	Minor	Major
9.	Drafting of paper	Major	Minor	Minor
10.	Revisions of final paper	Minor	Minor	Major
11.	Submission of final paper	Major	Minor	Minor
12.	Presentation of paper (conference)	Major	N/A	N/A

A10: A pedagogical evaluation of new state model diagrams for teaching internetwork technologies

Authors: S.P. Maj, G. Kohli, T. Fetherston (2004) Presented at 2005 ACSC, Australia

	Contributions (Major/equal/minor)	Author 1 Maj	Author 2 Kohli	Author 3 Fetherston
1.	Problem definition	Major	Major	Minor
2.	Literature review	Major	Major	Major
3.	Hypothesis formulation	Major	Major	Minor
4.	Experimental design	Major	Major	Minor
5.	Establish experimental platform (e.g. deign and implement hardware/software)	Major	Minor	Major
6.	Data Collection	Minor	Major	Minor
7.	Data interpretation	Minor	Major	Major
8.	Critical analysis	Major	Minor	Major
9.	Drafting of paper	Major	Minor	Major
10.	Revisions of final paper	Minor	Major	Minor
11.	Submission of final paper	Major	Minor	N/A
12.	Presentation of paper (conference)	Major	Major	N/A

10.APPENDIX - B

Work in Progress

The following papers are part of the ongoing research on state models. The paper titled ‘A New State Model for Router Management’ was blind peer reviewed for the International conference on Computational Sciences (ICCS 2004). Due to issues with conference management the stream under which this paper was accepted was cancelled. The details of the papers are as follows.

A New State Model for Router Management

S.P. Maj, G. Murphy, G. Kohli
Edith Cowan University
2 Bradford Street
Mt Lawley, WA 6050

ABSTRACT

Routers are typically managed through the command line interface in conjunction with software diagnostic protocols. Populations of these devices can be managed by various software tools that automate common management tasks. However, effective device management is potentially enhanced by considering a router as an integrated collection of active processes. None of the tools examined provide this integrated approach. This paper treats a router as an integrated, discrete, dynamic system described as a hierarchical state model. This model provides top-down decomposition thereby aiding router management. It has been successfully used for a number of routing protocols including EIGRP.

General Terms

Keywords
Router modelling, state machine

INTRODUCTION

Routers are an integral part of local area networks, wide area networks and the Internet. Routers must be configured to run routing protocols and then monitored for performance. Network problems must also be identified and corrected. Routers are typically controlled through the command line interface (CLI) in conjunction with various software diagnostic tools such as the Cisco Discovery Protocol (OSI Layer 2), PING (OSI Layer 3), TRACE (OSI Layer 3), IP ROUTE (OSI Layer 3) and also Telnet (OSI Layer 7). The CLI also allows the user to determine the status of the various components of a router such as routing table entries, Address Resolution Protocol (ARP) table entries, interface status etc. However, the hierarchical CLI commands may require considerable technical expertise. Furthermore the status information of the many different tables, interfaces etc must typically be obtained by a number of different CLI commands. This may be problematic during fault diagnosis when it is necessary to integrate the status information from a number of separate router components.

A large selection of Cisco sponsored textbooks; on-line Cisco curriculum and also non-Cisco textbooks were analyzed for an alternative, more integrated approach to router management. Cisco are the world's largest suppliers of internetworking equipment and endorse an extensive range of technical and educational material. It was found that internetworking devices (switches and routers) are considered as 'black boxes' by the Cisco on-line curriculum and also by introductory Cisco endorsed textbooks (Systems, 2001a), (Systems, 2001b), (Heap & Maynes, 2002), (Odom, 2000) and non Cisco endorsed textbooks (Shaughnessy & Velte, 2000), (Hudson, Caudle, & Cannon, 2000), (Lammie, 1999), (Velte & Velte, 2001), (McQuerry, 2000). All the material analyzed provided descriptions of device (router and switch) operation and the details of the associated CLI commands but did not use an integrated modelling method. Cisco endorsed textbooks on routing (Paquet & Teare, 2001), (McGregor, 2001), (Benjamin, 2002) and non Cisco books (Lammie, Timm, & Odom, 2002), (Zinin, 2002), (Grice, 2001b), (Thomas II, Aelmans, Houniet, & T, 2000) gave the same result. The same was found for both general (Shinder, 2001), (Systems, 2001b) and specialized books on networking (Pignataro, Kazemi, & Dry, 2003). It should be noted that Cisco use finite states to explain neighbor acquisition in BGP, route acquisition in EIGRP and also neighbor and route acquisition in OSPF. However this is not integrated with router management at the CLI interface.

When networks consist of a number of interconnected routers under a single management jurisdiction more sophisticated traffic monitoring techniques such as Simple Network Management Protocol (SNMP) must be used. SNMP is a protocol that allows network data to be centrally managed. Large populations of routers typically require routine, repetitive, error-prone and potentially costly maintenance. Router populations can be managed by various software tools that consolidate and automate common management tasks such as software distribution, change and audit authorization, device inventory etc. They can also be used to monitor network performance and detect network problems. Certainly the software management tools analyzed offered a comprehensive, modular approach to network management but do not provide an integrated status model of a router.

There exist a number of different modelling methods that have been used to teach networking that include: OPNET (N. Davies, Ransbottom, & Hamilton, 1998), CLICK (Kohler, Morris, Chen, Jannotti, & Frans Kaashoek, 2000), NsClick (Neufeld, Jain, & Grunwald, 2002) etc. However none of these integrate router status into a single hierarchical model.

Effective device management may be enhanced by considering a router as an integrated collection of active states. Such a representation is not currently available. Accordingly a number of different modelling approaches to do this were analyzed.

MODELLING METHODS

There exist a wide range of modelling methods such as formal specifications (e.g. Z, VDM), Petri nets, simulation, deterministic and non-deterministic analytical modelling etc. Each modelling method has its own strengths and weakness. Modelling characteristics considered of particular importance in this application are: diagrammatic, ease of use, ability to control detail by means of hierarchical top-down decomposition and also the integration of the status of the different components of a router. According to Cooling (Cooling, 1991), there are two main types of diagram: high level and low level. High-level diagrams are task oriented and show the overall system structure with its major sub-units. Such diagrams describe the overall function of the design and interactions between both the sub-systems and the environment. The main emphasis is ‘what does the system do’ and the resultant design is therefore task oriented. By contrast, low-level diagrams are solution oriented and must be able to handle considerable detail. With low-level diagrams the main emphasis, applicable to this application, is ‘how does the system work’. A router was modeled using a number modelling techniques and the results evaluated. The techniques evaluated included: object-oriented modelling (G Booch, 1978), function-oriented modelling i.e. Data Flow Diagrams (Cooling, 1991), Process Specification (Gane & Sarson, 1979), Jackson System Development (M. Jackson, 1975, 1983), Specification and Description Language (Rockstrom & Saracco, 1982), Structured Analysis and Design (Marca & McGowan, 1988) and Design Approach for Real-Time Systems (Gomaa, 1993). Most of these methods could capture information flow, and device behavior but it was difficult to relate the model results to the data extracted from the CLI.

A state-oriented approach was also attempted. According to Peters (Peters & Pedrycz, 2000), '*In a state-oriented approach to specifying the behavior of software, an entity is modeled as a state machine. The behavior of an entity is described in terms of the passage of the entity through different states. The state of a software system is defined in terms of the value of the available information about the system at a particular instant in time*.' Furthermore, '*State-oriented specifications include variables that change each time a state change occurs.*' The standard approach to this type of specification is the finite-state machine.

A finite-state machine can be defined mathematically and also employs a graphical notation. The FSM is one technique used for modelling communication protocols. According to Halsall (Halsal, 1996), '*The three most common methods for specifying a communication protocol are state transition diagrams, extended event-state tables and high-level structured programs*.' He further states, '*Irrespective of the specification method, we model a protocol as a finite state machine or automaton. This means that the protocol – or, more accurately, the protocol entity – can be in just one a finite number of defined states at any instant. For example, it might be idle waiting for a message to send or waiting to receive and acknowledgment.'* A communication protocol can therefore be modeled as a protocol entity that can exist in one of a number of defined states. A router implements a number of communication protocols. Accordingly state modelling was used as the basis of modelling a router.

ROUTER STATE VARIABLE MODEL (RIP)

A router is an OSI layer 3 device. Accordingly it implements layer 3, layer 2 and layer 1 protocols. Interface status is typically determined by the CLI command 'Show interface' (Figure 1).

```
Device#show interface fastethernet 0/1
FastEthernet0/1 is up, line protocol is up
  Hardware is Fast Ethernet, address is 000c.30df.81c1 (bia 000c.30df.81c1)
  MTU 1500 bytes, BW 100000 Kbit, DLY 1000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Auto-duplex, Auto-speed
  input flow-control is off, output flow-control is off
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 23:41:43, output 23:41:30, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue :0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    536 packets input, 72501 bytes, 0 no buffer
    Received 276 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
    0 watchdog, 164 multicast, 0 pause input
```

Figure 1: CLI Show interface output

The output from this command may require interpretation. Furthermore, the status information is available only after issuing the CLI command. This output may be simplified to consider only state connectivity status.

The OSI physical layer line status is triggered by a carrier detect signal. Physical connectivity state is either up or down. The line protocol, triggered by keepalive frames, is an OSI data link layer function. The Layer 2 connectivity state can also be considered either up or down. However, the key both these protocols may be represented on a single router diagram (Figure 2).

Consider the layer 3 Address Resolution Protocol (ARP). This protocol, or protocol entity, can exist in one of a three defined states – free, pending or resolved. In the free state the time-to-live entry has expired; the pending state means a request for an IP entry has been sent but no reply received and the resolved state means the physical (MAC) to logical (IP) mapping is complete.

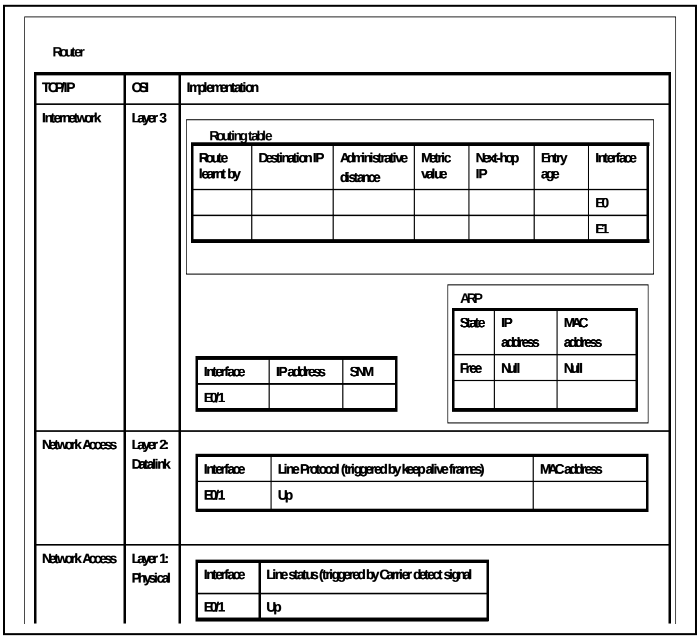


Figure 2: Router State Variable Model (RIP)

Each state has a state variable – Null, IP address and MAC address. All this information may be represented in an ARP State Variables Diagram (Figure 3). This diagram may be incorporated into a diagrammatic representation of a router (Figure 2). The router state variable information of this protocol is displayed in the ARP table with the logical (IP) and physical (MAC) address mapping. This information may be obtained by the CLI command 'show arp'.

Similarly a routing protocol such as Routing Information Protocol (RIP) can exist in a number of defined states. RIP is a simple distance vector routing protocol that sends updates every 30 seconds. When the router receives routing tables from other devices this triggers a routing table update and sets the 'Age Entry' timer event trigger. This data may also be represented in a single router diagram (Figure 2). The routing table details are obtained by the CLI command 'show ip route'.

The advantage of this approach is that all the important router status information is represented in a single diagram. This diagram links the communication state transitions and state variables of the communication protocols to data normally extracted from a number of complex hierarchical CLI commands. The model is diagrammatic, self-documenting and easy to use.

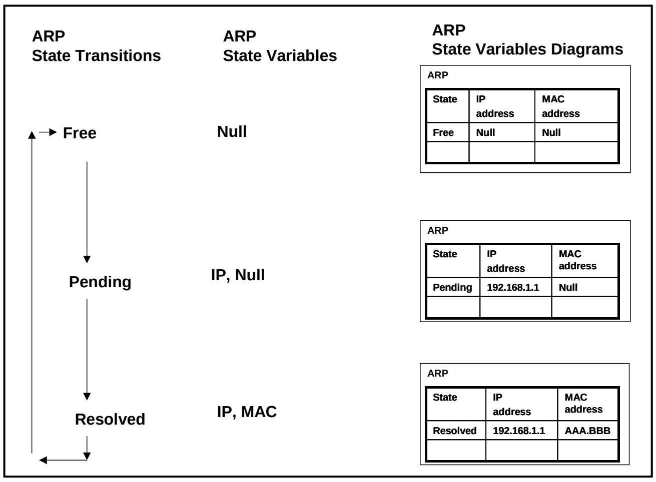


Figure 3: State transitions, variables and diagrams (STP)

State models such as this were used as the pedagogical foundation of two postgraduate units in network technology and the results evaluated. The results indicated a significant educational benefit (S.P & Kohli, 2004). Based on this very positive result the simple state model was developed further to include such as EIGRP.

ROUTER STATE VARIABLE MODEL (EIGRP)

The Routing Information Protocol (RIP) is a simple distance vector routing not suited to large networks due to problems with timing out, congestion and very slow convergence. Interior Gateway Routing Protocol (IGRP), a Cisco proprietary routing protocol was designed to address the limitations of RIP such as: hop count, composite metric, update timer etc. IGRP was superceded by the Cisco proprietary Extended IGRP (EIGRP) that is currently used for many large networks. EIGRP is defined by Cisco as an advanced distance vector routing protocol. However, it has some link-state characteristics.

In addition to the route table, EIGRP maintains two state variable tables - a neighbor table and a topology table. The neighbor table lists adjacent routers to ensure bi-directional communication between directly connected neighbors. The topology table contains all learned routes to all destinations. Both tables were simplified to include what was considered to be the key state information. The topology table state variables are: Route state, destination, successors, feasible distance (fd), advertised distance (ad) and route. The Neighbor table state variables are: process identification, neighbor address, handle and hold time. Both simplified state tables were incorporated into the Router State Variable model (Figure 4). A second level of diagram was also designed that provided more details about the topology and neighbor tables. The diagrams were then evaluated experimentally.

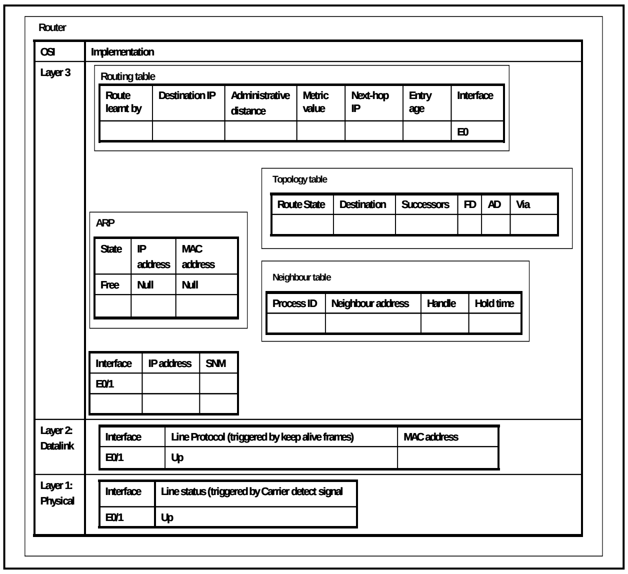


Figure 4: Router state variable model (EIGRP)

EVALUATION

A simple network was built (Figure 5). A hub was used to provide connectivity between both routers, which were running EIGRP. The packet sniffer NETMON was used to capture and monitor EIGRP packet traffic. A network such as this would normally be managed by six CLI commands – show ip eigrp neighbors, show ip eigrp topology, show ip route eigrp, show ip protocols, show interface e0/0 and show interface e0/1. EIGRP operation can also be verified by the following CLI commands – debug eigrp packets, debug eigrp neighbors, debug ip eigrp and debug ip eigrp summary. The output from these CLI commands can be complex.

The network was monitored using the router state variable model (EIGRP) and the results compared to the output from standard CLI commands. It was found that the diagram captured all the data directly relevant to monitoring and managing this network without the complexity of the output from different CLI commands. Significantly, it was possible to view all the key data on a single diagram. Furthermore, it was possible to obtain more detailed information, as needed, by expanding the either the topology or neighbor table.

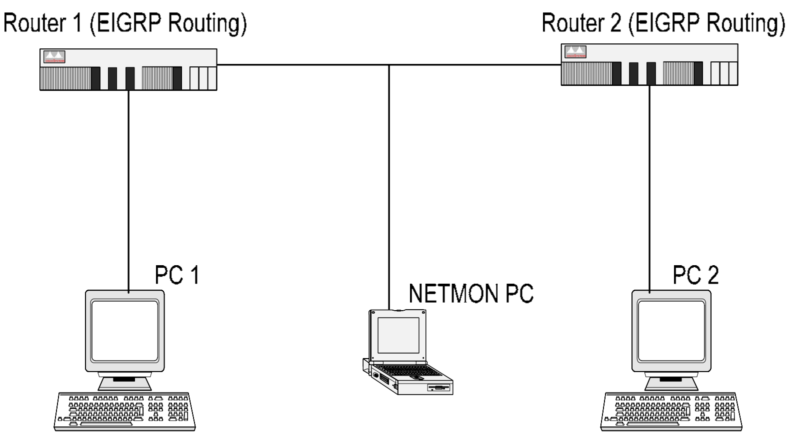


Figure 5: EIGRP Network.

CONCLUSIONS

Internetwork devices such as routers are typically managed by CLI commands. To determine the status of an active router a range of such commands must be used the output of which may be complex and require interpretation. No modelling method was found that specifically linked CLI command output with a diagrammatic representation. Accordingly, a hierarchical, top-down state variable model was developed. Using this model it was possible to capture all the important router status information in a single diagram. If needed, more detailed status information could be obtained by using the second level of diagram.

REFERENCES

1. Systems, C., *Cisco Network Academy Program: First Year Companion Guide*. 2nd ed. 2001, Indianapolis: Cisco Press.
2. Systems, C., *Internetworking Technologies Handbook*. 2001, Indianapolis: Cisco Press.
3. Heap, G. and L. Maynes, *CCNA Practical Studies*. 2002, Indianapolis: Cisco Press.
4. Odom, W., *Cisco CCNA Exam #640-507 Certification Guide*. 2000, Indianapolis: Cisco Press.
5. Shaughnessy, T. and T.J. Velte, *Cisco, A Beginner's Guide*. 2000, New York: Osbourne/McGraw-Hill.
6. Hudson, K., K. Caudle, and K. Cannon, *CCNA Guide to Cisco Networking*. 2nd ed. 2000: Thomson Course Technology.
7. Lammie, T., *Cisco Certified Network Associate Study Guide*. 2nd ed. 1999, San Francisco: Sybex.
8. Velte, T.J. and A.T. Velte, *Cisco, A Beginner's Guide*. 2nd ed. 2001, New York: Osbourne/McGraw-Hill.
9. McQuerry, S., *Interconnecting Cisco Network Devices*. 2000, Indianapolis: Cisco Press.
10. Paquet, C. and D. Teare, *Building Scalable Cisco Networks*. 2001, Indianapolis: Cisco Press.
11. McGregor, M., *Semester Five Companion Guide, Advanced Routing*. 2001, Indianapolis: Cisco Press.
12. Benjamin, H., *CCNP Practical Studies: Routing*. 2002, Indianapolis: Cisco Press.
13. Lammie, T., C. Timm, and S. Odom, *CCNP/CCIP BSCI Study Guide*. 2002, San Francisco: Sybex.
14. Zinin, A., *Cisco IP Routing*. 2002, Boston: Addison Wesley.
15. Grice, M., *CCNP Guide to Advanced Cisco Routing*. 2001: Thomson Learning.
16. Thomas II, T.M., et al., *Building Scalable Cisco Networks*. 2000, Indianapolis: Cisco Press.
17. Shinder, D.L., *Computer Network Essentials*. 2001, Indianapolis: Cisco Press.
18. Pignataro, C., R. Kazemi, and B. Dry, *Cisco Multiservice Swtiching Networks*. 2003, Indianapolis: Cisco Press.

19. Davies, N., S. Ransbottom, and D. Hamilton, *Teaching Networks through Modelling.* ACM SIGada Letters, 1998. **XVIII (5)**: p. 104-110.
20. Kohler, E., et al., *The Click Moduar Router.* ACM Transactions on Computer Systems, 2000. **18**(3): p. 263-297.
21. Neufeld, M., A. Jain, and D. Grunwald. *Nsclick: Bridging Network Simulation and Deployment*. in *International Workshop on Modelling Analysis and Simulation of Wireless and Mobile Systems*. 2002. Atlanta, Georgia: ACM Press, New York, NY, USA.
22. Cooling, J.E., *Software Design for Real-Time Systems*. 1991, Padstow, Cornwall: Chapman and Hall.
23. Booch, G., *Object-Oriented Analysis and Design*. 2nd ed. 1978, Redwood City, CA: Benjamin/Cummings.
24. Gane, C. and T. Sarson, *Structured Systems Analysis: Tools and Techniques*. 1979, Englewood Cliffs, NJ: Prentice-Hall.
25. Jackson, M., *Principles of Program Design*. 1975, New York: Academic Press.
26. Jackson, M., *System Development*. 1983, Englewood Cliffs, NJ: Prentice-Hall.
27. Rockstrom, A. and R. Saracco, *SDL-CCITT specification and description language.* IEEE Transaction on Communications, 1982. **30 (6)**: p. 1310-1318.
28. Marca, D. and C. McGowan, *SADT: Structured Analysis and Design Technique*. 1988, New York: McGraw-Hill.
29. Gomaa, H., *Software Design Methods for Concurrent and Real-Time Systems*. 1993, Reading, MA: Addisson-Wesley.
30. Peters, J.F. and W. Pedrycz, *Software Engineering - An Engineering Approach*. World Series in Computer Science, ed. D. Barron and W. P. 2000, New York: John Wiley &Sons.
31. Halsal, F., *Data Communications, Computer Networks and Open Systems*. 1996, Harlow, England: Addison-Wesley.
32. S.P, M. and G. Kohli, *New State Models for Internetwork Technology.* Issues in Informing Science and Information Technology Education, 2004. (Publication pending)

The following three papers were submitted, peer reviewed and accepted by the American Society of Engineering Education Conference 2005 held in Portland, Oregon and subsequently published in its proceedings.

The first paper titled ‘A Conceptual Model as an aid to student understanding of Network Security’ extends the work on state models to model security associated with internetworking devices e.g. switches and routers. Initial attempts have been made to model security issues related to layer 1, layer 2 and layer 3. The paper put forward a case for providing abstract models for managing network security and also for teaching students about network security.

The second paper titled ‘The Integration of State Diagrams with Competency-Based Assessment’ put forward a case for integrating state models with CBA. The use of CBA is widely used in computer and network technology. The details of the papers are as follows.

The third paper titled ‘A framework for a bandwidth based network performance model for CS students’. This paper outlines the use of a model that provides a conceptual framework for performance analysis based upon bandwidth. The bandwidth centric model uses high-level abstraction decoupled from the implementation details of the underlying technology. This paper represents an initial attempt to develop this theory further in the field of networking education and presents a description of some of our work to date. This paper also includes details of experiments undertaken to measure bandwidth and the formulae derived and applied to the investigation of converging data streams.

A Conceptual Model as an aid to student understanding of Network Security

G. Kohli, S. P. Maj, G. Murphy and D. Veal
Edith Cowan University, Perth, WA, Australia
g.kohli@ecu.edu.au

Abstract

Security is amongst the most widely discussed topics in today's world of high speed networking. Security broadly deals with problems that affect millions of computer users around the world either through the spread of viruses, or information theft from personal computers and network servers. Security issues can encompass large quantities of detailed information which can overwhelm network administrators. Security systems are traditionally often layered in a top-down manner. Abstract models could enable administrators to focus upon relevant details whilst filtering out non-essential details. Such models could also be used in a top-down fashion thus permitting the control of complexity via recursive decomposition. There are currently many security models used in industry and for teaching students about network security. These models are not only restricted to confidentiality, authentication, data integrity, non-repudiation, and access control, but also take into account physical and human aspects that can effect security. A model based upon Finite State Machines (FSM) and called a state model is proposed as an aid to device level management.

Introduction

The Internet is the driving force behind the rapid development of Computer and Networking technology. Whilst the Internet offers fast communication and ease of use, there are inherent problems. There has been a growing concern about information theft (Kargl, Maier, & Weber, 2001) and virus outbreaks on the Internet (Kocher, Lee, McGraw, & Rajhunathan, 2004). Furthermore Cisco notes with regard to corporate networks: "… *when you connect your network to the Internet, you are physically connecting your network to more than 50,000 unknown and all their users. Although such connections open the door to many useful applications and provide great opportunities for information sharing, most private networks contain information that should not be shared with outside users on the Internet*" (Systems, 1999). This gives the traditional administrator little choice but to protect and monitor the security of their networks. Security is one of the key tasks required of systems administrators.

The OSI seven layer model for networking was developed by ISO (International Standard Organisation) to define standardized methods for designing internetworks and their function. Its goal is to provide standards to which all computers hardware and software vendors will adhere, so that multiplicity of interconnection and interface practices could be reduced, thus reducing the costs of designing and producing both hardware and software. It is *A suite of protocols and standards sponsored by the ISO for data communications between otherwise incompatible* *computer systems* (*O'Reilly Search*, n.d.). The ISO code 7498-2 (*Security System Module*, 2004) defines the following:

Five types of security services
Eight security mechanisms that support the above services
Three required OSI security management methods.
The three dimensional graph put forward by ISO 7498-2 committee is shown in Figure 1.

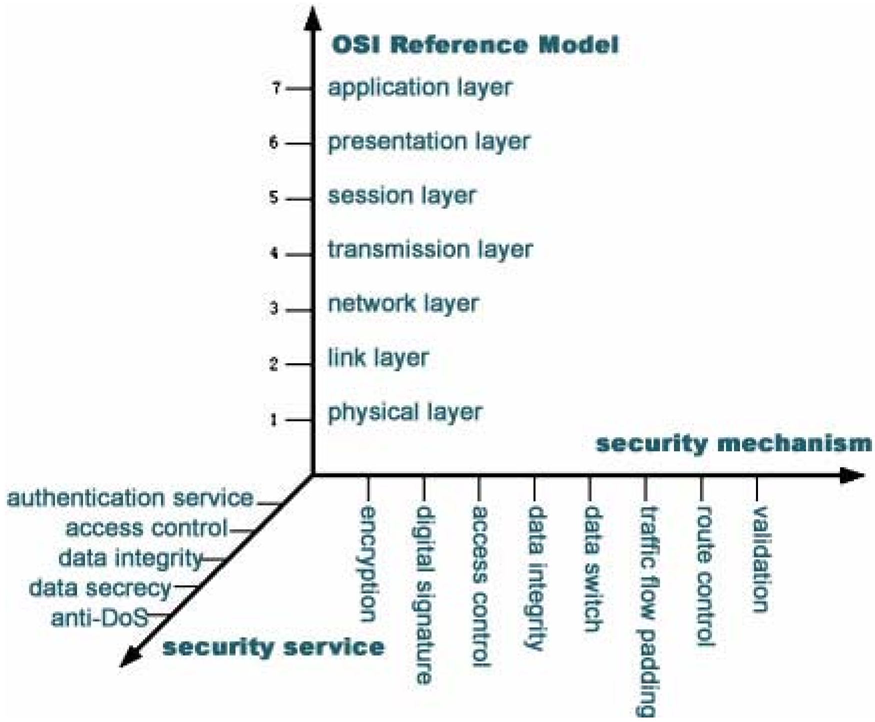


Figure 1: ISO 7498-2 3-dimentional graph (*Security System Module*, 2004)

There are currently many security models used in industry and for teaching students about network security (A. Dennis, 2002; Hung & Kamalakar, 2003). Some of these models are based upon the OSI model (Reed, 2003) and the IPsec Protocol Framework (Cisco, 2003). IPsec, in turn, relies on existing algorithms to implement the encryption, authentication, and key exchange (Systems, 2003). Other security models not based upon the OSI framework are premised upon role based models (Sandhu, 1992). However, Most of the security models developed to date are inadequate in the collaboration area (Aljareh & Rossiter, 2002). Whilst these models help administrators to understand security they may fail to provide an insight into security issues relevant to networking devices, i.e. the switches and routers that actually handle security. There is a need to develop a conceptual security model which can help networking administrators gain a clearer understanding of security issues on the networks they are managing.

At the device level network security deals with protocols, and all protocols can be expressed as finite state machines (FSM) (Halsal, 1996). Using a FSM, protocols can be modelled to exist in one of a number of defined states. In order to address implementation-specific details there is a need to consider protocols and the modelling of them to explain how internetworking devices e.g. switches and routers model security.

State Models

Models are a means of controlling detail and assisting communication. Among desirable characteristics are that any model is diagrammatic, self-documenting, and easy to use and permits hierarchical top-down decomposition to control detail. Levelling is the property in which complex systems can be progressively decomposed to provide completeness.

"*A model may be described as the body of information about a system gathered for the purpose of studying the system. It is not only an orderly collection of information, but is an orderly representation or structuring of the information. The characteristics should be representative of the characteristics of the real system* (Pooch & Wall, 1993). With respect to complex systems Burgess notes that "*System administration is full of intangibles; this restricts model building to those aspects of the problem which can be addresses in schematic terms. It is also sufficiently complex that it must be addressed at several different levels in an approximately hierarchical fashion* (Burgess, 2000).

According to Cooling there are two main diagram types of diagrams: high level and low level (Cooling, 1991). High level diagrams show the overall system structure with its major sub-units. By contrast, low level diagrams are solution oriented and must be able to handle considerable detail. Both high and low level systems may be represented by state models. One form of state model is an FSM. According to the National Institute of Science and Technology *A finite state machine is a model of computation consisting of a set of states, a start state, an input alphabet and a transition function that maps input symbols and current states to the next state. Computation begins in the start state with an input string. It changes to new states depending on the transition function* (Unknown).

At any given moment in time the system exists in a certain state. The set of all states is the state space. Significantly the state diagrams should show only details that are relevant to the current state. Two simple state models have been developed – one for a switch and one for a router (Kohli et al., 2004). However unlike typical state models these new models allow the introduction of progressively advanced conceptual features hence they provide scalability and complexity control (S. P Maj & G Kohli, 2004). Furthermore, Burgess, under a chapter heading "*Analytical Systems Administration*", notes that: "*… now days many computing systems are of comparable complexity to phenomena found in the natural world and our understanding of them is not always complete, in spite of the fact that they were designed to fulfil a specific task. In short, technology might not be completely predictable; hence there is an need for experimental verification"* (Burgess, 2000).

State models can be extended to model security starting with the physical layer. The physical layer is responsible for the physical communication between nodes. It is concerned with the actual encoding and transmission of data into electricity. The physical layer is critical as far security and delivery of communication data is concerned. Van Eck states the following with regards to eavesdropping on the physical layer *it is possible in some cases to obtain information on the signals used inside the equipment when the radiation is picked up and the received signals are decoded. Especially in the case of digital equipment this possibility constitutes problem, because remote reconstruction of signals inside the equipment may enable reconstruction of the data the equipment is processing* (Eck). Physical security is vital in any network (Beasley, 2004).

Switch Security Model

Maj proposed a diagrammatic state model of a switch (S. P Maj & G Kohli, 2004). Each physical port is represented on the switch model (e.g. Fastethernet 0/1 or Fa0/1). At the simplest level, connectivity can be represented by internal connections between the ports within the switch. At a more complex level switches perform three main tasks: address learning; address forwarding and filtering; loop avoidance. A simple table can be incorporated into this diagram to show how a switch learns and hence maps physical MAC addresses to ports i.e. address learning. This table can be then be used to show how a switch establishes one to one connectivity (micro-segmentation) and hence performs address filtering and data forwarding. Figure 2 provides an overview of the switch state model (S. P Maj & G Kohli, 2004). An advantage of the switch state model is that it includes in a single diagram capture the key features of the switch along with the relevant command line outputs (CLI). The model presents information in a hierarchical manner thereby controlling complexity. Furthermore, it provides scalability by expanding the basic switch model to cater for Spanning Tree Protocol (STP) and security (Maj et al., 2004).

Layer 2 vulnerability details with MAC Address, VLAN, VTP, etc (Reed, 2003). The switch state model can be modified to include security information. Various CLI outputs such as *show vlan*, *show vtp status*, etc can be used to gather the information about important security information at layer 2. Information gathered from these command can be integrated into a single state single switch security model (Figure 3). It should be noted that Figure 3 demonstrates the models.

<table>
<tr><th>TCP/IP</th><th>OSI</th><th>Implementation</th></tr>
<tr><td>Application</td><td>Layer7</td><td>NO</td></tr>
<tr><td>Application</td><td>Layer 6</td><td>NO</td></tr>
<tr><td>Application</td><td>Layer 5</td><td>NO</td></tr>
<tr><td>Transport</td><td>Layer 4</td><td>NO</td></tr>
<tr><td>Internetwork</td><td>Layer 3
Network</td><td>NO</td></tr>
<tr><td>Network
Access</td><td>Data Link</td><td>MAC-Address-Information
<table>
<tr><th>VLAN</th><th>MAC
Address</th><th>Type</th><th>Interfaces</th></tr>
<tr><td>1</td><td>AAAA AAAA AAAA</td><td>Dynamic</td><td>Fa0/1</td></tr>
<tr><td>2</td><td>BBBB BBBB BBBB</td><td>Dynamic</td><td>Fa0/2</td></tr>
</table>
Interface table
<table>
<tr><th>Interface</th><th>Line Protocol</th><th>Type</th></tr>
<tr><td>F0/1</td><td>Up</td><td>Trunk</td></tr>
</table></td></tr>
<tr><td>Network
Access</td><td>Physical
Layer</td><td>Carrier detect table
<table>
<tr><th>Inteface</th><th>Line Status</th></tr>
<tr><td>Fa0/1</td><td>up</td></tr>
</table></td></tr>
</table>

Figure 2 Switch Model

Network Access | Data Link

VLAN Information

Vlan	Name	Status
1	Default	Active
2	Security	Active

MAC-Address-Information

VLAN	MAC Address	Type	Interfaces	Age Time
1	AAAA AAAA AAAA	Dynamic	Fa0/1	300
2	BBBB BBBB BBBB	Dynamic	Fa0/2	300

Interface table

Interface	Line Protocol	Type
F0/1	Up	Trunk

VTP table

VTP Mode	Authenticaion/password	Domain
Server	MD5 / *******	test

Network Access | Physical Layer

Carrier detect table

Inteface	Line Status
Fa0/1	up

Figure 3 Switch Security Model

A Router Security Model

Similarly a router has been modelled using the ARP and routing table (Kohli et al., 2004). On a Cisco router the router commands "show arp" and "show ip route" can be used to in conjunction with the diagrams to show the state changes as networks are connected together (Figure 4).

TCP/IP	OSI	Implementation
Application	Layer 7	No
Application	Layer 6	No
Application	Layer 5	No
Transport	Layer 4	No

Network | Network

Routing Table

Route learnt by	Destination IP	Admin distance	Next-Hop IP	Interface
R	192.168.2.1	120	192.168.1.2	Fa0/0

ARP table

State	Mac Address	IP
Free	AAAA AAAA AAAA	192.168.1.1

Interface table

Interface	Mac Address	SNM
Fa0/1	AAAA AAAA AAAA	24

Network Access | Data Link

Interface table

Interface	Line Protocol
F0/1	Up

Network Access | Physical Layer

Carrier detect table

Inteface	Line Status
Fa0/1	up

Figure 4: Simple Router Model

Again this simple router model can be further modified to monitor security at layer 3 in the seven layer OSI model. Of major concern at layer 3 are the security of routing protocol updates, and preventing certain packets (which, the authors note, may or may not include routing protocol updates) from being passed to the next router (Reed, 2003).

The exchange of routing protocol updates is used to ensure that all routers in an administrative area, for example a university campus or a corporate headquarters, have a common view of the administrative area, and thus can establish paths, or routes, to every network in that administrative area. Route updates are sent and received by routing protocols, and are always sent in clear text. This means that a potential hacker can use packet sniffing software, examples of which are readily available on the Internet, to intercept and capture routing updates between two routers that are connected on a broadcast, multi-access network e.g. Ethernet. However, some routing protocols will only accept updates from, or send updates to, a neighbouring router if they have a shared authentication method. Some routing protocols do not offer authentication. For example RIP v1 does not provide authentication and broadcasts routing updates from all configured ports every 30 seconds (Cisco, 2002). The solution to this problem can be the use of another routing protocol such as RIP v2, OSPF, or EIGRP which does provide authentication. It should also be noted that most protocols offer both simple authentication and hashed authentication. In simple authentication, passwords are sent in clear text and so can be sniffed. For this reason, simple authentication should not be used to provide security in a production environment. In hashed authentication the key and route update are used to generate a hash, and the hash and route update are then sent to the receiver. The receiver accepts the update and hash, passes the update and its own key through the hash algorithm, and then compares the output with the received hash. If the hashes do not match, the update is rejected. This ensures that a router will only accept route update information from an identified partner, and should guarantee the integrity of the update. It does not, however, guarantee that the sending router has not been misconfigured, or has not passed on data that has otherwise been incorrectly or maliciously injected into the system. As noted previously, the route updates themselves continue to be transmitted in clear text and may be intercepted, thus providing an overview of the network to a sophisticated attacker. The choice of routing protocol depends upon network design and scalability of the routing protocol (Grice, 2001a).

It should also be noted that default operation of most routing protocols is to send updates out of all interfaces on a router, if that interfaces network is being advertised by the routing protocol. This means that even when no other router is attached to a broadcast multi access network, the router will send updates out to the network. Again, a sophisticated attacker can sniff the network, capture the updates, and reconstruct the topology from the information obtained. In Cisco routers this default behaviour is overcome by making the interface passive.

The passage of layer 3 packets, which can include routing protocol updates, through a router can be controlled by Access Control Lists (ACL), although Davies notes that ACLs can adversely affect router performance (P. T. Davies, 2002). In an article entitled "The Cost of Security on Cisco Routers" it is stated that *There are significant performance penalties once you enable ACLs, especially long ones that* *we used in our tests, because an access list cannot always take advantage of the fastest switching technique that might otherwise be available on the router* (Morris, 1999).

Furthermore, dramatic performance reductions after implementing 200-line ACLs have also been noted. Bandwidth degradation can be reduced by using hardware based Private Internet Exchange (PIX) firewalls and layer 3 switches (Bruno, 2003). From the performance perspective, ACLs can cause serious degradation in network performance but provide extra necessary security. Among other tasks, ACLs may be used as either a packet filter or as a route filter. When used as a packet filter, they can permit or deny transit traffic based on its source IP address, its destination IP address, its TCP or UDP source port, its TCP or UDP destination port, or any combination of these. Thus, for example, a router ACL could be used to permit a single host on a network to access the Internet, while preventing all other hosts on that network from doing the same. When used as route filter, they can be used to permit or deny transit route update traffic about a given destination network.

For the purposes of this paper, the authors are interested in security at layer 3 and how can it expressed using state models. The modified router security state model is shown in Figure 5.

Transport | Transport

ACLID	Source Port	Destination Port	Interface	Direction
25		80	F0	IN

Network | Network

Routing Table

Route learnt by	Destination IP	Admin distance	Next-Hop IP	Interface
R	192.168.2.1	120	192.168.1.2	Fa0/0

User table

Username	Password
fred	fred

Authentication

Authentication chain	Authenti cation mode
Test	MD5

Authentication key for Test

Key ID	Key
12345	Julie

Standard ACL

ACLID	Interface	Direction	Permit/Deny	SN
101	Fa0/1	IN	Deny	10.0.0.0

ARP table

State	Mac Address	IP
Free	AAAA AAAA A	192.168.1.1

Interface table

Interface	Mac Address	SNM
Fa0/1	AAAA AAAA AAAA	24

Network Access | Data Link

Interface table

Interface	Line Protocol
F0/1	Up

Network Access | Physical Layer

Carrier detect table

Inteface	Line Status
Fa0/1	up

Figure 5 Router Security Model

State Model as an Aid to Teaching Networking

The diagrams were used as the pedagogical foundation of non-vendor based curriculum in networking technology and the results evaluated (Maj, Kohli, & Fetherston, 2005; Maj et al., 2004). Students on two different units were given 20 and 40 hours instruction based on the new models. The results were compared with students from three other vendor based units who had received 100, 120 and 160 hours of instruction using the standard method of teaching based on the CLI. A networking expert was interviewed by means of a list of questions and the results recorded. The same questions were given to all groups. Despite the large difference in teaching time, the two groups taught using the new state model correctly used far more terms than the other three groups. Furthermore the answers provided by the two groups taught using the new models closely mapped the answers obtained from the expert.

Within education it is well documented that after successfully completing an examination it is not uncommon for the majority of students to demonstrate very poor retention of not only factual information but also concepts. One month after setting their examinations the students taught using the new models were again evaluated. The majority of the students clearly demonstrated that they had internalised the model. On questioning, they were able to reproduce a working model, although this was not an exact copy of the ones provided. Furthermore they demonstrated an understanding of concepts they had been taught. However further work is needed.

State models may provide advantages that network administrators may find useful. Network administrators need to search through various configuration scripts and screens output from the switch and router; this can involve huge amounts of data which can result in 'information overload'. In contrast by using the security state model all the relevant information can be trapped on the state diagram thus providing a more effective method of gleaning appropriate information. The following are some of the potential advantages of security state models:

- They provides a hierarchical view of the network;
- They make fault diagnosis easier to handle;
- A single diagram captures key security vulnerabilities; and
- Information can be hidden via abstraction.
- The uniformity and reproducibility of the model make it much easier to identify sought information.

Potential problems in the use of state models include:

The use of abstraction information could inadvertently be hidden that could prove useful in a particular situation. A limitation of the state model as developed to date is that it only captures information for the bottom four layers of the OSI model. Further work is in progress to model the security of the top three layers of the OSI model.

Conclusions

Understanding security is of crucial importance in today's world of high speed networking. As new vulnerabilities are constantly being discovered one has to provide an insight into security issues relevant to networking devices, e.g. the switches and routers that handle security. The use of a conceptual security model may help networking engineers and computer networking students gain a clearer understanding of relevant security issues on the networks that they manage or study. The use of a state model is proposed for the conceptual modelling of security on networking devices and covers some of the key issues in networking security. Although no extensive testing has been undertaken by the authors, initial investigation suggests that these models can be of use in aiding the understanding of complex systems handling security. Furthermore, the state models provide advantages of abstraction via the use of levelling and information hiding thereby controlling complexity. This method may also provide a hierarchical perspective of networking devices which may help in fault diagnostics. Further research on state models is currently being undertaken by the authors.

Bibliography

1. Kargl, F., J. Maier, and M. Weber. *Protecting web servers from distributed denail of service attacks*. in *tenth international conference on World Wide Web*. 2001. Hong Kong: ACM Press.
2. Kocher, P., et al. *Security as a new dimension in embedded system design*. in *41st annual conference on design automation*. 2004. San Diego, CA, USA.
3. Systems, C., *Internetworking Technologies Handbook*. 1999, Cisco Press: Upper Saddle River, NJ. p. Chap.51 p.1.
4. *O'Reilly Search*.
5. *Security System Module*. 2004.
6. Hung, C.K. and K. Kamalakar. *A secure workflow model*. in *Proceedings of the Australasian information security workshop confernce on ACSW frontiers 2003*. 2003. Adelaide, Australia: Australian Computer Soceity.
7. Dennis, A., *Networking in the Internet Age*. 2002, John Wiley & Sons, Inc: New York, NY, USA.
8. Reed, D., *Applying the OSI Seven Layer Network Model To Information Security*. 2003, SANS Institute. p. 1-29.
9. Cisco, *Securing Cisco IOS Networks*, C.L. Product, Editor. 2003, Cisco Systems, Inc.
10. Systems, C., *Securing Cisco IOS Networks*. 2003.
11. Sandhu, R. *A lattice interpretation of the chinese wall security policy*. in *Proc. 15th NIST-NCSE National Computer Security Conference*. 1992: US Govt. Printing Office.
12. Aljareh, S. and N. Rossiter. *A task-based security model to facilitate colloboration in trusted multi-agency networks*. in *Proceedings of the 2002 ACM symposium on Applied computing*. 2002. Madrid, Spain: ACM Press.
13. Halsal, F., *Data Communications, Computer Networks and Open Systems*. 1996, Harlow, England: Addison-Wesley.

14. Pooch, U.W. and J.A. Wall, *Discrete event simulation: A practical approach*. 1993, Boca Raton, FL: CRC Press.
15. Burgess, M., *Principles of Network and System Administration*. 2000, John Wiley & Sons Ltd: Chichester, England. p. 293.
16. Cooling, J.E., *Software Design for Real-Time Systems*. 1991, Padstow, Cornwall: Chapman and Hall.
17. Unknown, *National Institute of Science and Technology*.
18. Kohli, G., et al. *Abstraction in Computer Network Education: A model based approach*. in *ASEE*. 2004. Salt Lake City, Utah.
19. Maj, S.P. and G. Kohli, *A New State Model for Internetworks Technology.* Journal of Information Technology Education, 2004. **1**.
20. Eck, W.V., *Electromagnectic Radiation From Video Display Units: An Eavesdropping Risk?*
21. Beasley, J., *Networking*. 2004, Pearson Press: Upper Saddle River, NJ. p. 338.
22. Maj, S.P., G. Murphy, and G. Kohli. *State Models for Internetworking Technologies*. in *34th ASEE/IEEE Frontiers in Education Conference*. 2004. Savannah,GA.
23. Cisco, *Cisco Networking Academy Program: Second Year Companion Guide*. 2 ed. 2002, Indianaplois IN USA: Cisco Press. 718.
24. Grice, M., *CCNP Guide to Advanced Cisco Routing*. 2001, Boston MA: Thomson Course Technology. 424.
25. Davies, P.T., *Securing and Controlling Cisco Routers*. 2002, Boca Raton FL: Auerbach Publications.
26. Morris, C., *White Paper: The cost of security on Cisco Routers*. 1999.
27. Bruno, A.A., *CCIE Routing and Switching*. 2003, Indianaplois IN: Cisco Press.
28. Maj, S.P., G. Kohli, and T. Fetherston. *A Pedagogical evaluation of new state model diagram for teaching internetworking technologies*. in *28th Australasian Computer Science Conference*. 2005. Newcastle, Australia: Australian Computer Society.

The Integration of State Diagrams with Competency-Based Assessment

G. Kohli, D. Veal & S. P. Maj and G. Murphy
Edith Cowan University, Perth, Western Australia
g.kohli@ecu.edu.au

Abstract

Hands-on units in Computer Networking technologies are increasingly popular amongst Computer Science students. However, to test the hands-on component it has been found to be necessary to use Competency Based Assessment (CBA). The hands-on exercises can become outdated very quickly due to the rapid advancement of technology. To offset such effects the authors have developed an abstract high level model to aid students' conceptual understanding across a range of technologies. These models are capable of providing state information of different internet networking devices e.g. switch, routers, and also to model routing protocols. The authors provide a method of integrating state diagrams along with CBAs.

Introduction

As a reflection of the computer industry requirements there is an increasing emphasis on computer network and data communications in the Computer science curriculum. This has been supported by ACM / IEEE (ACM, 2001). Networking courses are often based one ore more of the following areas as noted by Davis et al(N. Davis et al., 1998):

- The OSI Model;
- Performance Analysis;
- Network Simulation

Within computer networking, data communication, Information Systems (IS), and management units the OSI model based instruction is very common (Gutierrez, 1998; R. Smith, 2001). Typically in this approach, the functionality of each layer of the OSI model is explained and then examples of this functionality in a specific network protocol are presented. A problem with the OSI Model is that it fails to provide new details with respect to protocols and functionality compared with the TCP/IP or Department of Defence Model except perhaps that the OSI model uses more layers. However, there are few, if any, implemented network protocols in which the actual architectural layers of a real system are strictly aligned with those of the OSI model. The OSI model provides the basis of layering networks and it also helps hardware and software designers to use a common framework for design and development of equipment. Most networking courses start by giving an overall overview of OSI model (Comer, 1999; R. Smith & Francia, 2000).

On the other hand courses based on performance analysis use analytical based models that are often specialized in their area of application. This may involve the use of complex mathematics which may not be suitable or relevant to an employer's expectation for many computer networking students. However, its advantages include the use by students of powerful mathematical tools. A major disadvantage is that these models may not be based upon real systems.

Simulation of computer networks provides students with the advantages of simulated observation of the operation of a network. Barrnet (Barnett III, 1993) proposes the use of the NetSim simulator to support both major project assignments and more focused homework assignments. Whatever the simulation tool, a prudent technique is to incorporate the tool into supervised lab/project assignments, individual homework assignments, and classroom demonstrations (Barnett III, 1993).

Although simulated environments have their benefits, real hands-on exercises give students a chance to apply the theory they learn from textbooks (Minch & Tabor, 2003). It has been noted that practical experience is necessary to fully understand network management problems, and that it is desirable to change the level of participation of students by increasing expectation and so making students more responsible for their own learning (Lankewicz, 1998; McDonald et al., 2001).

A possible approach to teaching computer networks topics is to allow students to learn using real networks. Traditionally, computer networks courses have not provided students with hands-on access to networking equipment and software; cost and implementations factors have made it difficult (Dixon et al., 1997; Pattinson & Dacre, 1998). However, due to increasing popularity of vendor-based courses as components of undergraduate curricula, students now have the opportunity to study a more practical approach and hence program networking devices (switches, routers). Furthermore, relatively inexpensive equipment, such as switches and routers, and associated on-line vendor based curricula, such as Cisco Certified Networking Associate (CCNA), and the Cisco Certified Networking Professional (CCNP) are now readily available (Cisco, 2004). Furthermore, many students are studying networking and internetworking from a non computing science and even a non-technical background (Maj et al., 1998; Maj & Veal, 2000). Students doing the Cisco Networking Academy Program (CNAP) need to comprehend a large number of new concepts within a short time span. Each unit of a CNAP generally requires more time than is available under the university course structure. For example, in one university in Western Australia, four hours per week for 12 teaching weeks, for a total of 48 student contact hours, are allocated to deliver a one unit of CNAP program that Cisco recommend should take 70 hours. The CNAP program is based on a hands-on approach and mandates the uses of physical equipment (switches and routers) as part of the learning.

The hands-on approach to network technology education requires an understanding of switch and router operation. However, an analysis of educational materials (Hudson et al., 2003; Lewis, 2003; Waters et al., 2000) in this area has indicated that these devices are typically treated as 'black boxes'. Such an approach may not be best suited to the promotion of learning as this force student to construct their own mental model of the internal operation of such devices. These students constructions may or may not, be correct and may not be able to encompass future ideas. There is a need for students to develop a good conceptual model of networking devices.

State Models

A high level abstract model has been developed to aid conceptual understanding. The state models implementation is based upon Finite State Machines (FSM) (Halsal, 1996). Using a FSM, network protocols can be modelled to exist in one of a number of defined states. These models are capable of providing state information of different internet networking devices e.g. switch, routers, and also to model routing protocols (Maj & Kohli, 2004; Maj et al., 2004).

The state models present students with appropriate information which can facilitate understanding. Typically internetworking students are presented with the large quantity of information on typical router and switch screens as the result of user interrogation of these devices. The information presented can be quite complex and may require advance understanding. This is contrary to education theory which supports the need of abstraction in order to construct good understanding. With regards to abstraction the ACM notes “Levels of Abstraction: the nature and use of abstraction in computing; the use of abstraction in managing complexity, structuring systems, hiding details, and capturing recurring patterns; the ability to represent an entity or system by abstractions having different levels of detail and specificity” (ACM/IEEE, 2001). Furthermore, Rumbaugh et al define abstraction as “Abstraction is the selective examination of certain aspects of a problem. The goal of abstraction is to isolate those aspects that are important for some purpose and to suppress those aspects that are unimportant" (Rumbaugh et al., 1991).

Abstractions can be applied to computer and networking technology to help provide students with common fundamental concepts regardless of the particular underlying technological implementation. This helps to avoid the rapid redundancy of a detailed knowledge of modern computer and networking technology implementation and hands-on skills acquisition.

Maj et al have developed two states models based upon the principle of abstraction (Maj et al., 2005; Maj et al., 2004). In case of state model a PC can be modelled as a simple state device with a logical (IP) to physical (MAC address) Address Resolution Protocol (ARP) table and a Network Interface Card (NIC) table (IP address, Subnet mask and MAC address). The PC command “*IPCONFIG*” output directly maps onto these simple PC state diagrams. A router can then be modelled as a state diagram using the ARP and NIC table (as found in the PC) plus a routing table (figure 1). Hence an incremental learning path is provided. The router commands “*show arp*” and “*show ip route*” can be used to in conjunction with the diagrams to show the state changes as different networks are connected together.

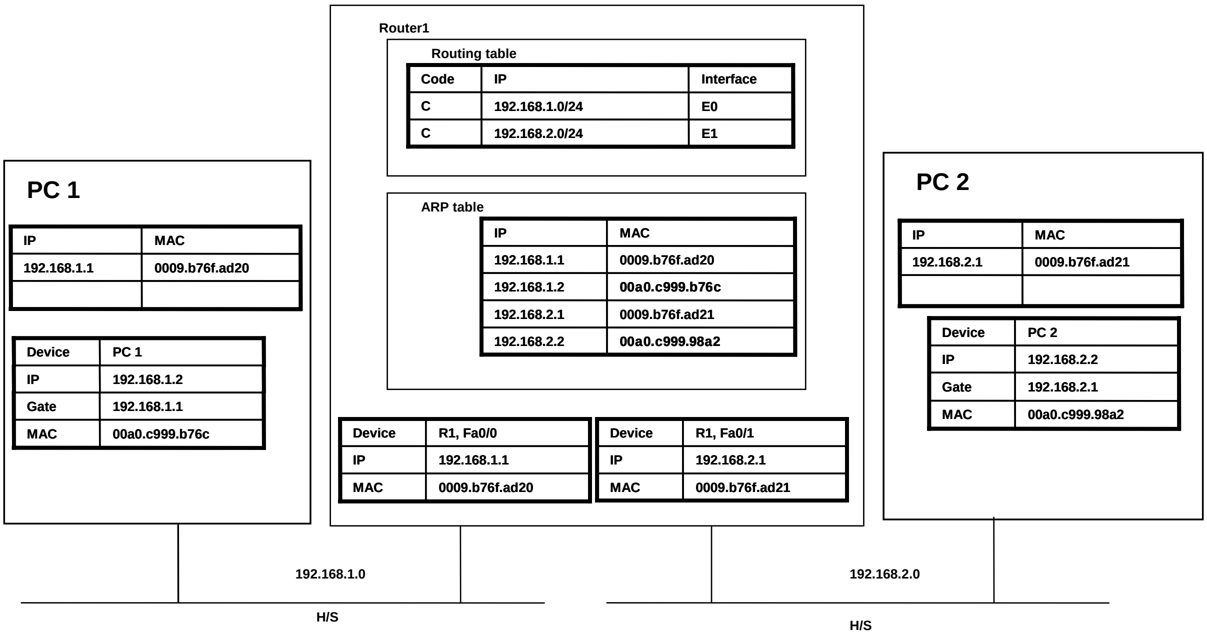


Figure 1: Router Model

The state models provide the basic understanding which the students can use to relate how internetworking system physically work.

Competency based assessment

Competency has been defined as: "*The ability to perform in the workplace*" (Goldsworthy, 1993, p. 113). Mirabile also defined competency as : "*A knowledge, skill, or characteristic associated with high performance on a job*" (Mirabile, 1997, p. 75).

Competency has been the cause of a great deal of controversy within higher educational institution in Australia. A 1995 Australian Government National Board of Employment Education and Training (NBEEB) report on the "Demand for and Dimensions of Education and Training" notes: "*Employers are increasingly emphasising a broad range of non-academic factors as more accurate indicators of a new employee's potential to succeed in the workplace. … They recruit on the assumption that graduates have satisfied the academic requirements of each institution, thus allowing them to focus on the particular skills and attributes they believe are most essential for the particular work environment. Generally, employers emphasise skills and attributes which are more difficult to evaluate than academic skills*" (NEEB, 1995, p. 21).

Another report by Australian Government report "*Learning for the knowledge society: An education and training action plan for the information economy*" noted the importance of universities ensuring that graduates enter the workforce with the required competencies (DETYA, 2000)

Within the computer networking courses the practical hands-on assessment is gaining in popularity and Murphy notes: *That practical 'hands on' skills and* *knowledge is fundamental to conceptual understanding. That knowledge is more transferable to different situations when acquired by a gradual process of conceptual understanding* (Murphy, 1998, p.246).

Furthermore, Tarrant favours the need of practical assessment: *In fact both practical skill and knowledge of certain principles are jointly sufficient for success. This should warn against any scheme in which practical knowledge is rigidly separated form theoretical knowledge* (Tarrant, 2000, p. 77).

The use of CBAs may place a large burden on the staff conducting them if the classes are large. It has been the authors' observation that in a class of 40 students it takes around 60 hours to conduct the actual CBA assessment. This was found to be the case during the current semester. The assessment time included setting up the laboratory between different assessments and ensuring that the equipment used during the assessments was working correctly. In regard to time constraints Osterich when discussing the Cisco Certified Internetworking Expert (CCIE) lab tests notes that *The lab exam takes place over two days. The first day the candidate has the task of setting up a complex internetwork using disparate technologies. During the evening of the first day, test administrators essentially sabotage the work you did in the morning, so the second day you spend troubleshooting and diagnosing those issues. The CCIE exam has a high failure rate: generally, more than 80% of first time candidates fail* (Osterich, 2000, p.7).

A further problem with the notion of competency is that of a common definition. Laver gives an example within Australian context (Laver, 1994), stating that *Semantic confusion has arisen because the term 'competency' is used in at least three different senses: in the training sector, it means the capability to perform certain designated tasks satisfactory so that defined outcomes can be met; in the Mayer Committee's Report it covers more generic skills, such as problem solving and planning; and in the university sector, though generally rejected, it is sometimes recognised as comparable to such concepts as graduate attributes* Cited in (Reid, 1996, pp. 133-134).

Secondly within the university sector the use of competency is not common as students are not often given credits based upon whether they can perform a task successfully or not. Understanding of the topic is regarded as more important, although some competency skills that can be offered as part of the unit are desirable. The later conditions should not be seen as mutually exclusive.

The authors decided to integrate the use of state diagrams as discussed above along with CBAs. The idea behind this assessment was to measure understanding of the topic covered during the semester. Furthermore it allowed the authors to measure the student's ability to configure a physical network. Each student was given a case study on internetworking. The case study involved building the physical network and filling the appropriate information on to the state diagram provided.

The information could be obtained from the various Command Line Interface (CLI) outputs from the internetworking devices such as switches and routers. This information can then be transferred to the state diagrams. This allowed the students to produce a diagram of the physical network and also capture relevant information regarding that particular case study as required for the assessment.

Evaluation

At the end of the assessment students were asked to fill in a questionnaire based upon their experience with this assessment. The results of the questionnaire are tabulated in Table 1.

Questions	Yes (%)	No (%)	Note Sure (%)	No Answer (%)
Are you satisfied with various assessment instruments used as part of this course?	86.36	4.55	4.55	4.55
Are you satisfied with the timing of the assessment during the semester?	77.27	22.73	0	0
Do you only want practical assessment as part of this unit?	68.18	22.73	9.09	0
Has the assessment in this unit helped you in doing other assessment as part of your degree?	86.36	4.55	9.09	
Has the assessment helped you to improve your understanding of the topics covered in this unit?	86.36	0	0	13.64
Has the state diagram helped in better understanding the case study used in the assessment	80	0	0	20

Table1: Student Responses

Clearly a huge proportion of the students (86.36%) are satisfied with the methods used in the assessment. Furthermore, the use of state diagrams along with the assessment proved to a good mix of integrating understanding and competency. However, further investigation is needed as to the time problem which are not been addressed by this research.

Conclusions

The hands-on units are becoming more popular in computer network courses and this necessitates the use of CBAs to assess hands-on skills. The use of CBA has both advantages and disadvantages. The authors proposed integrating state diagrams along with CBAs to measure student understanding. The initial finding have been promising and the authors are currently extending the scope of this research to factor out other issues associated with the use of CBAs, one of the most important of which is time management.

Bibliography

1. ACM, *Computing curricula 2001 ACM/IEEE Joint Task Force Computer Science Final Report*. 2001, ACM Press.
2. Davis, N., S. Ransbotton, and D. Hamilton, *Teaching Computer Networks Through Modelling.* Ada Letters, 1998. **XVII**(Sept/Oct).
3. Gutierrez, J.A., *A connectionless approach to integrated network management.* International Journal of Network Management, 1998. **8**(4).
4. Smith, R., *The computer networks course: will it all fit?* Journal of Computing Science in Colleges, 2001. **17**(2).
5. Comer, D., *Computer network and Internets*. 1999, Upper Saddle River, NJ: Prentice-Hall, Inc.
6. Smith, R. and G. Francia. *A Hybrid Computer Networks Course for CS and IS Majors*. in *Proceedings of ISECON 2000*. 2000. Philadelphia, PA.
7. Barnett III, B.L. *An Ethernet Performance Simulator for Undergraduate Networking*. in *Twenty-Four SIGCSE Technical Symposium on Computer Science Education*. 1993: Miller Education.
8. Paxson, V. and D. Floyd. *Wide Area Traffic: The Failure of Poisson Modelling*. in *IEEE/ACM Transaction on Networking*. June 1995.
9. Minch, R. and S. Tabor, *Networking Education for the New Economy.* Journal of Information Technology Education, 2003. **2**.
10. Lankewicz, L.B. *Resources for Teaching Computer Networks*. in *Twenty-ninth SICSE Technical Symposium of Computer Science Education*. 1998. Atlanta, Georgia: ACM Press.
11. McDonald, M., et al., *Practical experiences for undergraduates computer networking students.* JCSE, 2001. **16**(3): p. 261-270.
12. Dixon, M., T. McGill, and J. Karlsson. *Using a network simulation package to teach the client-server model*. in *Proceedings of the conference on Integrating Technology into Computer Science Education*. 1997.
13. Pattinson, C. and T. Dacre. *A network model for network managment teaching.* in *Proceedings of the Third Australasian Conference on Computer Science Education*. 1998.
14. Cisco, *Cisco Networking Academy Program*. 2004, Cisco Systems.
15. Maj, S.P., et al. *Computer & Network Infrastructure Design, Installation, Maintenance and Management - a proposed new competency based curriculum*. in *The Proceedings of the Third Australasian Conference on Computer Science Education*. 1998. The University of Queensland, Brisbane, Australia.
16. Maj, S.P. and D. Veal, *Computer Technology Curriculum - A New Paradigm for a New Century.* Journal of Research and Practice in Information Technology, 2000. **32**(August/September): p. 200-214.
17. Hudson, K., K. Caudle, and K. Cannon, *CCNA Guide to Cisco Networking*. 2 ed. 2003, Boston: Thomson Course Technology. 534.
18. Waters, J., M.J. Rees, and J.T. Coe, *CCNA Routing and Switching*. 2 ed. 2000, Scotsdale Arizona: Coriolis. 387.
19. Lewis, W., *CCNP Cisco Networking Academy Program: Multilayer Switching Guide*. 2003, Indianapolis IN: Cisco Press. 636.

20. Halsal, F., *Data Communications, Computer Networks and Open Systems*. 1996, Harlow, England: Addison-Wesley.
21. Maj, S.P. and G. Kohli. *A New State Model for Internetworks Technology*. in *Informing Science + Information Technology Education Joint Conference*. 2004. Rockhampton, OLD Australia.
22. Maj, S.P., G. Murphy, and G. Kohli. *State Models for Internetworking Technologies*. in *34th ASEE/IEEE Frontiers in Education Conference*. 2004. Savannah,GA.
23. ACM/IEEE, *Computing curricula 2001 ACM/IEEE Joint Task Force Computer Science Final Report*. 2001, ACM.
24. Rumbaugh, J., et al., *Object-Oriented Modelling and Design*. 1991, New Jersey: Prentice-Hall.
25. Maj, S.P., G. Kohli, and T. Fetherston. *A Pedagogical evaluation of new state model diagram for teaching internetworking technologies*. in *28th Australasian Computer Science Conference*. 2005. Newcastle, Australia: Australian Computer Society.
26. Goldsworthy, A.W., *IT and the Competency Debate - Skill vs Knowledge A Major Issue*. The Australian Computer Journal, 1993. **25**(3): p. 113-122.
27. Mirabile, R.J., *Everything you wanted to know about competency modelling.*, in *Training and Development*. 1997. p. 73 -78.
28. NEEB, *Demand for and Dimensions of Education and Training*. 1995, National Board of Employment Education and Training (NBEET): Canberra. p. 23.
29. DETYA, *Learning for the knowledge society: An education and training action plan for the information economy*. 2000, Department of Eduaction Training and Youth Affairs.
30. Murphy, K., *Western Australian Upper Secondary Schooling Scince the 1960s*, in *Education*. 1998, Murdoch: Perth, WA. p. 309.
31. Tarrant, J., *What is Wrong with Competence?* Journal of Further and Higher Education, 2000. **24**(1): p. 77 - 83.
32. Osterich, H., *CCNA 2.0 Prep Kit Routung and Switching*. 2000, Indianapolois IN: QUE. 570.
33. Laver, P. *Overview of values: An Industrial Perspective In Training Minds: How well do Employers and Educators align their values and responsibilities?* in *Proceedings of the Industry -Education Forum*. 1994. Perth Australia.
34. Reid, I., *Higher Education or Education for Hire? - Language and Values in Australian Universities*. 1996, Rockhampton, Queensland: Central Queensland University Press.

A Framework for a Bandwidth Based Network Performance Model for CS Students

D. Veal, G. Kohli, S. P. Maj

Edith Cowan University
Western Australia

J. Cooper

Curtin University
Western Australia

Abstract

There are currently various methods by which network and internetwork performance can be addressed. Examples include simulation modeling and analytical modeling which often results in models that are highly complex and often mathematically based (e.g. queuing theory). The authors have developed a new model which is based upon simple formulae derived from an investigation of computer networks. This model provides a conceptual framework for performance analysis based upon bandwidth considerations from a constructivist perspective. The bandwidth centric model uses high-level abstraction decoupled from the implementation details of the underlying technology. This paper represents an initial attempt to develop this theory further in the field of networking education and presents a description of some of our work to date. This paper also includes details of experiments undertaken to measure bandwidth and formulae derived and applied to the investigation of converging data streams.

Introduction

There are many ways of approaching computer networking education. Davies notes that: *"Network courses are often based on one or more of the following areas: The OSI model; Performance analysis; and Network simulation"* [1]. The OSI model is a popular approach that is used extensively in the Cisco Networking Academy Program (CNAP) [2] and in other Cisco learning materials. With respect to simulation Davis describes the Optimized Network Engineering Tools (OPNET) system that that can model networks and sub-networks, individual nodes and stations and state transition models that defines a node [1]. However, Davies gives no indication as to the accuracy of this simulation or of the limits of its application. The development and testing of networking simulations may depend upon student's possessing knowledge and having suitable experience in computer programming languages that many networking students might not possess.

Performance analysis can use analytical based models that are often specialized in their area of application and may involve the use of complex mathematics which could be problematic for many computer networking students. Its advantages include the use of powerful mathematical tools. However, these models may not be based upon real systems. Queuing is a form of analytical modeling [3]. Networks of queues can also be analyzed [4, 5]. However, queuing analysis often assumes a Poisson distribution, which is not the case with most networks [6,7]. Performance analysis in computer networking can be based upon various models; and bandwidth in MB/s is a common performance indicator. Students can perceive differences in performance with respect to bandwidth which can be used as a starting point for a constructivist based performance model.

Constructivism

The dominant theory of conceptual understanding in education is constructivism. This has been extensively tested in the field of science and mathematics education [8-11]. Although Constructivism is a foundation of many modern teaching practices it has not been influential within computer education [12]. The knowledge the learner has already constructed will affect how new knowledge is interpreted [13], *"Put in the simplest way, to understand what someone has said or written means no less but also no more than to have built up a conceptual structure that, in a given context, appears to be 'compatible' with the structure the speaker had in mind. – and this compatibility, as a rule, manifests itself in no other way than that the receiver says and does nothing that contravenes the speaker's expectations"*[14]. The importance of conceptual understanding in the field of computer science education has been highlighted by the ACM who note that: *"Concepts are the raw material for understanding new information technology as it evolves"* [15]. Hence there is a need to develop a performance-based model in computer networking curricula that has its foundation based upon students' common conceptual understanding. The B-Node model provides such a foundation based upon the students' own perceptions of computer network performance [16].

A Bandwidth Model

B-Node models are bandwidth centric high-level abstractions, which are independent of the underlying implementation details of a particular technology [16]. The rapid rate of technological change within the field of Computer and Networking echnology (CNT) has given rise to the need to future-proof student learning. This is achieved by omitting details of the underlying technological implementation, which may change rapidly as the technology progresses [17,18]. Furthermore additional advantages of a bandwidth centric approach are that bandwidths can be readily measured in computer networks and bandwidths are often included in network equipment specifications. As B-Nodes are high-level abstract models they allow the possibility of recursive decomposition into their lower level component parts [16].

Potential Problems with a Bandwidth Centric Approach

A bandwidth centric approach needs to address the problem of latency. Hennessy and Patterson, under a heading of *"Fallacies and Pitfalls"* mention the: *"… pitfall of using bandwidth as the only measure of network performance. … this may be true for some applications such as video, where there is little interaction between the sender and the receiver, but for many applications such as NFS, are of a request-response nature, and so for every large massage there must be one or more small messages … latency is as important as bandwidth"* [19].

In this paper latency is addressed by subsuming its effects under a heading of bandwidth. Bandwidth is defined as being equal to the size of the file transmitted from source to destination, divided by the time taken to send that file. Latency is simply included in the time to transfer the file. Networks and computers may produce more bandwidth and yet result in a lower performance when compared to another system that produces better performance with less bandwidth [20]. Yet any measure of performance has its drawbacks and may not yield meaningful results in all circumstances.

Deterministic B-Node Models

The authors chose to concentrate upon an initial deterministic model rather than a stochastic model. Lai and Baker note that: *"Deterministic models are typically easier to work with mathematically than stochastic models, enabling us to find an analytical solution rather than a numerical one. Unfortunately, a deterministic model implies modeling with absolute certainty and many things cannot be known with enough certainty to make this practical"* [21]. However, a deterministic model can allow the checking of the basic parameters upon which to base a more complex stochastically based model. These initial parameters are to be determined by experimental results.

The Experiments

As an approach to these difficulties of modeling network performance, the authors have designed a solution that has analytical components and yet is both relatively simple to use and quantitatively based. These investigations commenced with experiments using PING to determine a baseline for subsequent bandwidth measurements. Table 1 shows the shorthand used for describing a given chain of networked of devices.

Device chain	Abbreviations
PC to fiber link to PC	PFP
PC to PC	PP
PC to switch to PC	PSP
PC to Router to PC	PRP
PC to Switch to two Routers to Switch to PC	PS2RSP
PC to Switch to three Routers to Switch to PC	PS3RSP

Table 1: Device chain abbreviations

Extended PING is a program used to check connectivity at the first three layers of the OSI model. This was used under Windows XP to measure the latency between a source and a destination. The version of PING used under Windows XP only gives a resolution of 1 ms and is of unknown accuracy. To address this problem we used a baseline of PCs connected by a single crossover cable. The results are shown in Figure 1 which shows the size of the data transferred against the transfer times. This also provides a measure of the bandwidth as 5.48 MB/s. It should be noted that these results are based upon binary Megabytes where as 100 mbps is denary based. The actual bandwidth is twice this value, approximately 11 MB/s, due to Round Trip Time (RTT). Multiple values were taken to obtain a better approximation of bandwidth. The slope for respective devices gives bandwidths which are tabulated in table 2.

Devices	Line plotted on graph	Bandwidth (MB/s) (Slope x 2)
PP	A	11.1
PSP	B	11.4
PSRSP	C	7.9
PS2RSP	D	7.5
PS3RSP	E	7.3

Table 2: Legend for PING transfer time graph

It can be seen from Figure 1 that when a switch replaces a crossover cable as the link between two PCs this introduces a latency of much less then 1 ms and the link operates approximately at line speed line speed (approximately 12 MB/s) as the slopes of lines ‘a’ and ‘b’ are approximately equal. The latency introduced by the addition of a switch is negligible at this level, as shown by the horizontal differences of points on the lines of ‘a’ and ‘b’. The latency of adding extra routers and is shown by the time between lines ‘c’ and ‘d’ and lines ‘d’ and ‘e’ which is approximately 1ms.

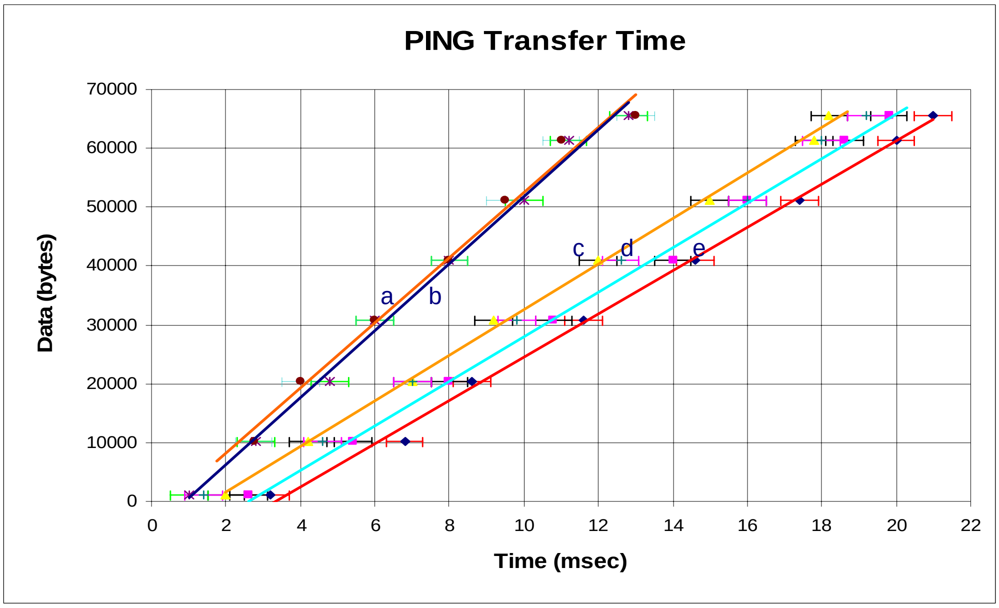


Figure 1: Data verses PING transfer time

The bandwidth drops from 11.4 MB/s to approximately 7.5 MB/s as shown by the difference in twice the slope of appropriate lines. It should be noted that fast switching was enabled by default on all routers and switches [25] .

FTP Transfer Time Experiments

Further experiments were conducted by transmitting across a series of devices using File Transfer Protocol (FTP). This was done to investigate larger file transfers, since PING can only send data up to 64KB. Figure 2 shows the bandwidths obtained using different configurations an example of which shown in Figure 3. This showed that the introduction of a router in a series of (B-Nodes) decreased the bandwidth to 7.5 MB/s.

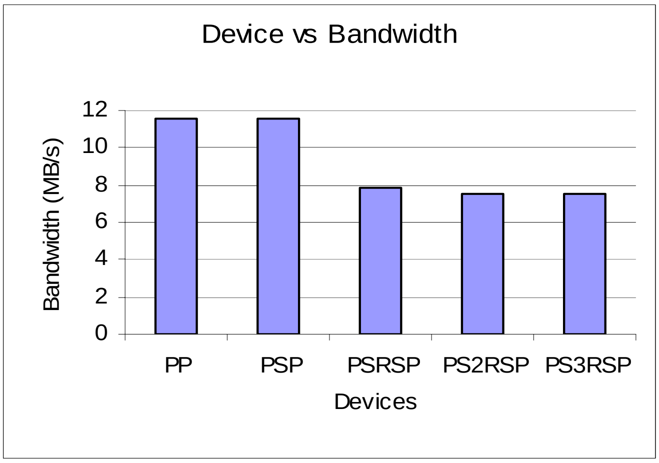


Figure 3: Device verses bandwidth for FTP transfers

Negative Bandwidth Experiments

In order to model the reduction in bandwidth due to the addition processing on devices (e.g. a router) the authors measured the decrease in observed bandwidth when processing is taking place compared to the bandwidth when this processing is not taking place, and they defined this as 'negative bandwidth'.

To investigate further the effect of processing in the router upon the bandwidth of traffic flowing through it, Access Control Lists (ACLs) were used to provide a measure of 'negative bandwidth'. ACLs can adversely affect router performance [22]. Significantly in product documentation the authors could not find where the effects of ACLs were quantified even though such a measure can be useful. To obtain such a measure experimentally, extended ACLs of varying numbers of Access Control Entries (ACEs) were applied to router Fast Ethernet interfaces and effects upon bandwidth noted. The results are shown in Figure 4. It can be concluded that, with the equipment used, the decrease in bandwidth is approximately 32kB/s per additional ACE as indicated by the slopes. (The dashed line is represents a file size of 150MB and the solid line indicates a file size of 15MB).

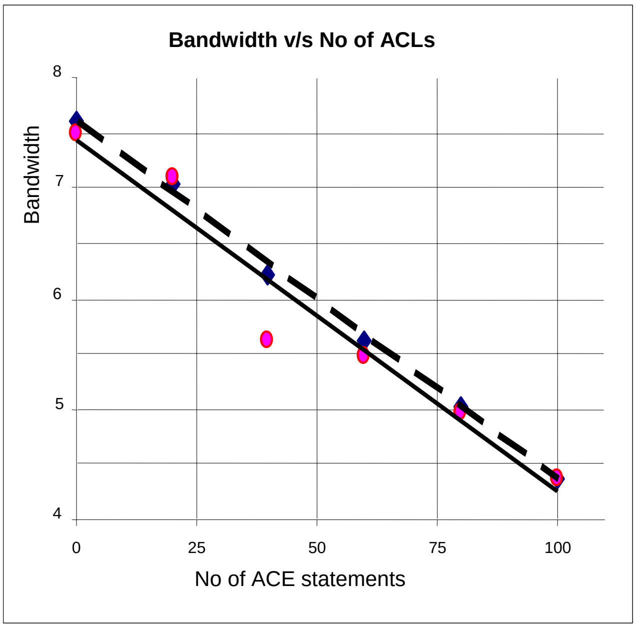


Figure 4: Bandwidth verses number of ACE statements

B-Node Model Requirements

The B-Node model would need to handle both series and parallel configurations. These are represented as shown in Figures 5 and 6. Given a chain of devices in series, the resultant bandwidth is the minimum individual bandwidth of given set of individual bandwidths of each of these devices:

B(Resultant) = Min $\{B_1 , B_2 , B_3 , \ldots\}$.

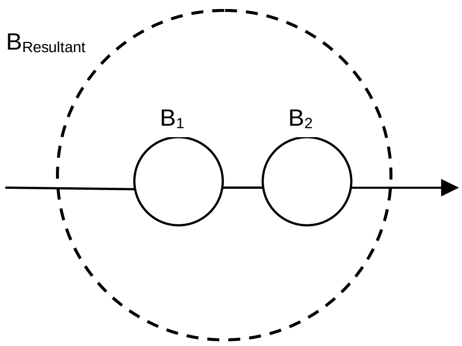


Figure 5: B-Nodes in series

This minimum result was also noted by Jain and Dovrolis [23].

ACL's were applied to a series of routers and the major effect on the overall source to destination bandwidth was dominated by the router with the maximum number of ACLs statements. Furthermore, this principle can also be seen to apply when routers were introduced into the chain the bandwidth dropped from 11.5 MB/s to 7.5 MB/s. As well as routers in series, data can also flow via parallel routes through an internetwork as show in Figure 6. The division of such flows can depend upon the particular routing protocol implemented [24, 25]. This requirement led to a resultant bandwidth which is the sum of given set of parallel bandwidths:

B (Resultant) = Sum {B1, B2, B3 , …}

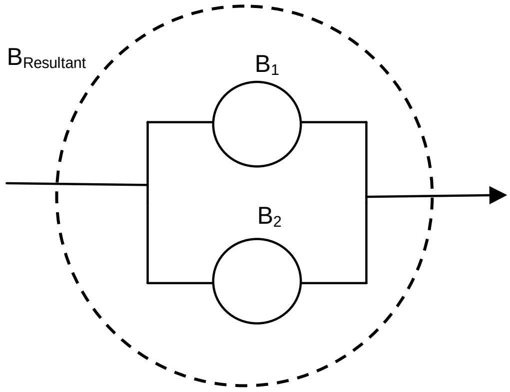


Figure 6: B-Nodes in parallel

Modeling the Flow of Data

To allow for more complex configuration the modeling of networks and internetworks via bandwidth based B-Nodes requires a consideration of converging, diverging, counter-flow and crossing data-flows. These have the representations shown in Figure 7.

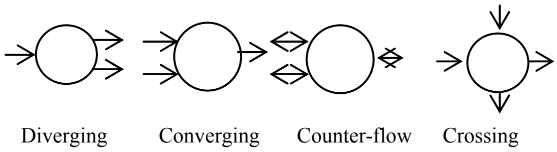


Figure 7: Data flow representations

Converging Data Flows

Files were sent across a network from a source PCs to a destination PCs that is from $P_1$to P_4, P_2 to P_5 and P_3 to P_6 as shown in Figure 8.

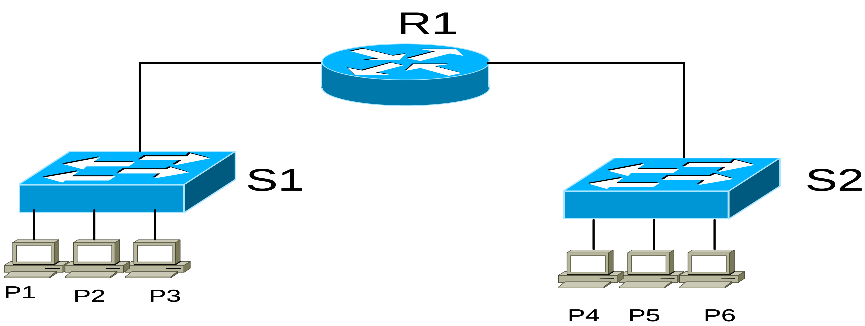


Figure 8: Experimental configuration

Files of different length, such that $L_3 > L_2 > L_1$, were first sent individually between PSRSP configuration. A value of channel capacity C, which is the bandwidth of the source to destination PC using FTP, 7.5 MB/s was used in calculations. This was the average obtained previously in over 100 readings. This meant that the minimum channel capacity 'C' did not occur in the PC but in the routers. Next the effect of multiple files sharing a link of capacity C of was investigated. The version of FTP used only gave transfer times therefore an experimental design technique used to reduce timing errors was used to send a series of files across the network. Whereby each subsequent file sent was smaller than the previous file and also arrived ahead of it, as shown in Figure 9.

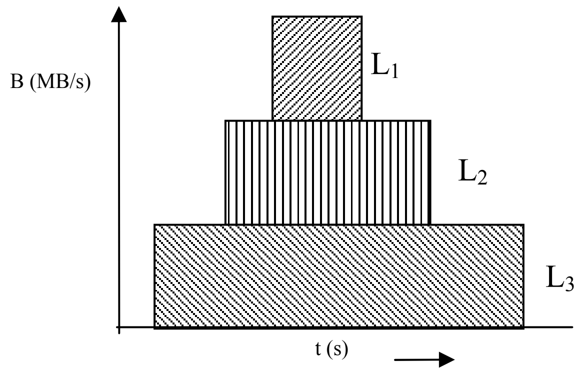


Figure 9: File transmission sequences

As, the initial bandwidths of the file transmission without any other data sharing the channel was 7.5MB/s = C.

The Convergence Formula

A hypothesis was that the files would "squeeze" into the channel and so become elongated with respect to time. This can be seen by considering the diagrams in Figure 10. Lie presents a similar argument for such elongation [21]. This is in keeping with what was expected from equal priority packet switching.

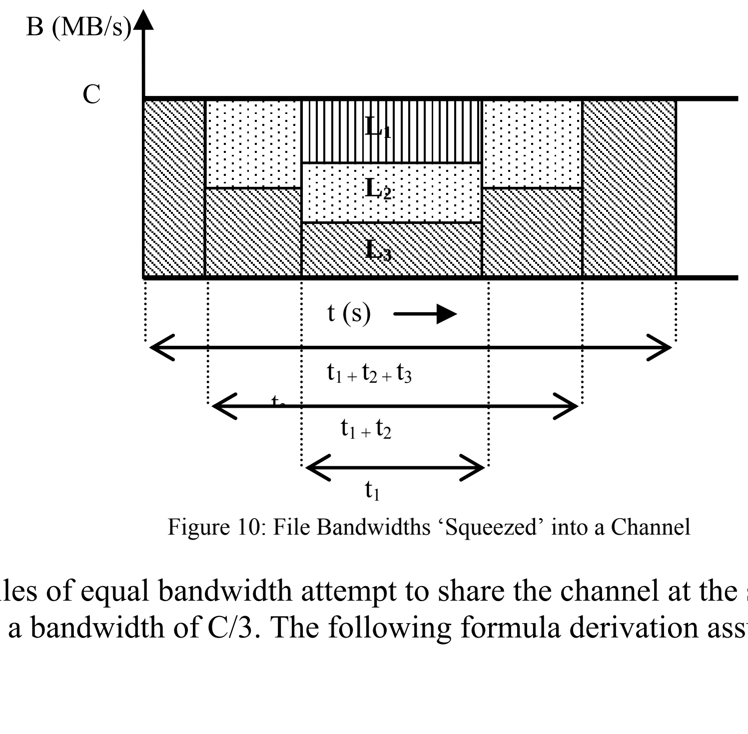


Figure 10: File Bandwidths 'Squeezed' into a Channel

If three files of equal bandwidth attempt to share the channel at the same time then each gets a bandwidth of C/3. The following formula derivation assumes equal bandwidths on each data flow of data sharing a channel, equal priority packet switching and also that the channel capacity C ≤ sum of the individual bandwidths of the data flows. This gives the following series for the times to transfer the files and results in the formula shown for determining total transfer time. Assuming nested file transmission times as in Figure 10 and that N represents the number of files to be transmitted.

Where $L_0 = 0$ and $B_0 = 0$.

It can be seen by considering Figure 9 that time $t1 = L_1/(C/3) = 3L_1/C$

Similarly t_2 = (remaining data in L2)/ bandwidth

$$= (L_2-L_1)/(C/2) = 2(L_2-L_1)/C$$

$$t_3 = (L_3-L_2)/C$$

$$t_{total} = t1 + t2 + t3 \ \dots \ tN$$

$$= N\,L1/C + (N-1)(L2 - L1)/C \dots + (L_n - L_{n-1})/C$$

This series results in the following formula for the nth term:

$$T_n = \frac{(N - n + 1)(L_n - L_{n-1})}{C}$$

File L_1 will arrive with a time of t_1 and file L_2 will arrive with a time $t_1 + t_2$ and so on. These results were confirmed by experiment within an accuracy of 5% or better which confirms how the FTP files converged. These experiments were needed to form a foundation upon which to build more complex models to form networks and internetworks. Such internetworking models can then be made more complex as show in Figure 12.

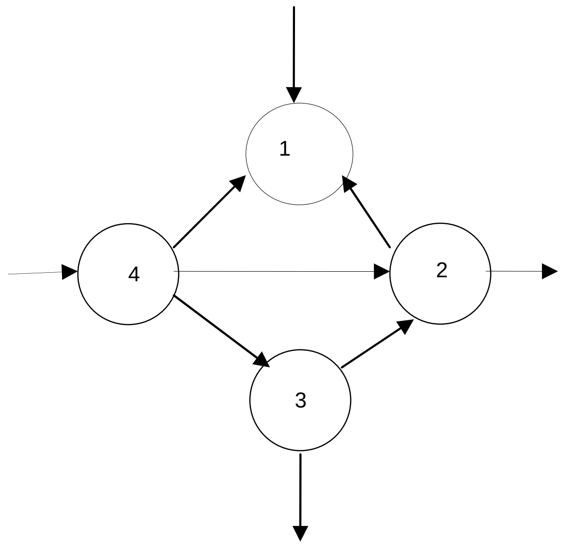


Figure 11: B-Nodes with Recursive Decomposition

The B-Node model needs to undergo further development and testing and its breakdown points need to be determined.

Conclusions

Constructivism offers the possibility of meaningful conceptually-based measures of performance. B-Node models are high-level abstractions that are decoupled from the underlying technological detail. From the work to date, under the constraints of the equipment used, it may be concluded that the switches operated at line near to a line speed. Furthermore, introducing routers into the path of the data flow reduced bandwidths.

The authors have quantitatively evaluated the negative bandwidth introduced per extra-extended ACE statement. Further investigations are being undertaken to determine the effects of counter-flows and cross-flows. The B-Node model needs further investigation and its breakdown points need to be determined. Further work is planned to repeat these experiments with a range of routers switches and PCs and to use a range of other protocols such as HTTP as the transfer program. Work to date suggests that this simple quantitative model may support student learning.

Bibliography

1. N, Davies, S. Ransbottom, & D. Hamilton, Teaching computer networks through modeling. *ACM SIGada ada Letters*, XVIII(5): 104-110. (1998).
2. Cisco, Cisco networking academy program: First year companion guide. 2 ed. (Indianapolis, IN: Cisco Press. 2001).
3. L. Kleinrock, Queuing systems vol I: Theory. New York, John Wiley and Sons. (1975).
4. T.G. Robertazzi, Computer networks and systems: Queuing theory and performance evaluation. Telecommunication networks and computer systems, ed. M. Gerla, P. Kuehn, and A. Lazar., New York: Springer-Verlag. (1990).
5. D. A. Menasce, & A.F. Alemeida, Capacity planning for web services: Metrics, models, and methods Upper Saddle River, NJ: Prentice Hall. (2002).
6. D.J. Lilja, Measuring Computer Performance: A practitioner's guide Cambridge: Cambridge University Press. (2000).
7. V.Paxton,.& S. Floyd, Wide area traffic: The failure of Poisson modeling. *IEEE/ACM Transactions on Networking*, 3(3): 226-244. (1995).
8. Cobern, W.W. Contextual constructivism: The impact of culture on the learning and teaching of science. In *Annual Meeting of the National Association for Research in Science Teaching*. Lake Geneva, WI: NARCT. (19910.
9. J. Confrey, Learning to listen: A student's understanding of powers of ten., in radical Constructivism in Mathematics Education, E. von Glasersfeld, Editor. Kluwer Academic Publishers: Dordrecht, Holland. 111-138. (1991).
10. R. Driver, & B. Bell, Students' thinking and the learning of science: A constructivist view Leeds, UK: Centre for Studies in Science Education University of Leeds. (1985)
11. G.H. Wheatley, Constructivist perspectives on science and mathematics learning. Science Education, 75(1): 9-21. (1991).

12. M. Ben-Ari, Constructivism in Computer Science Education. In *Twenty Ninth SIGCSE Technical Symposium on Computer Science Educ*. Atlanta, GA: ACM Press. (1998).
13. J. P. Mestre, Cognitive Aspects of Learning and Teaching Science, in Teacher Enhancement for Elementary and Secondary Science and Mathematics: Status, Issues, and Problems, S.J. Fitzsimmons & L.C. Kerpelman, Eds. National Science Foundation: Washington, D.C. 3-54. (2000).
14. von Glasersfeld, E., Cognition, Construction of Knowledge, and Teaching. Synthese, (80), 121-140. (1989).
15. ACM, Computing Curricula 2001 ACM/IEEE Joint Task Force Computer Science Final Report. ACM. (2001).
16. S.P.Maj, & D. Veal. B-Nodes: A Proposed New Method for Modelling information System Technology. In the I*nternational Conf on Computing and Information Technologies ICCIT'2001*. Montclair State University, NJ. (2001).
17. S. P.Maj, & D. Veal, Computer technology curriculum - A new Paradigm for a New Century. *Journal of Research and Practice in Information Technology November (Formerly Australian Computer Journal)*, 32(3/4) 200-214. (2000).
18. S. P. Maj, D. Veal, & R. Duley. A Proposed New High Level Abstraction for Computer Technology. In *SIGCSE ACM 2nd Technical Symposium in Computer Science Educ*. Charlotte, NC. (2001).
19. Hennessy, J.L. & D.A. Patterson, Computer Architecture: A quantitative approach. San Francisco, CA: Morgan Kaufmann. (1996).
20. McComas, B., Can the Pentium 4 recover? 2001, InQuest Market Research.
21. Lai, K & M. Baker. Measuring link bandwidths using a deterministic model of packet delay. In *ACM SIGCOMM 2000 Conf*. Stockholm, Sweden: ACM Press. (2000).
22. Davies, P.T., Securing and Controlling Cisco Routers., Boca Raton, FL: Auerbach Publications, (2002).
23. Jain, M. & C. Dovrolis. End-to-End Available Bandwidth: Measurement Methodology, Dynamics, and Relation with TCP Throughput. in *ACM SIGCOMM 2002 Conf.* Atlanta, GA: ACM Press. (2002).
24. Lammie, T. & A. Pfund, CCNP Support Study Guide. 2 ed., San Francisco, CA: Sybex. (2002).

McGregor, M., CCNP Cisco Networking Academy Program: Advanced Routing. 2 ed., Indianapolis, IN: Cisco Press. (2001).

11.LIST OF ACRONYMS

ACJ.............................. Australian Computer Journal

ACLs .... Access Control Lists

ACM .... Association for Computing Machinery

ACS .... Australian Computer Society

AMCIS .... Americas Conference on Information Systems

APC .... Australian Personal Computer

ATM............................. Asynchronous Transfer Mode

AITP............................. Association for Information Technology Professionals

AMCIS .... Americas Conference on Information Systems

ARP .... Address Resolution Protocol

ASEE............................ American Society for Engineering Education

BCS .... British Computer Society

BGP .... Border Gateway Protocol

BIOS............................. Basic Input Output System

bps .... bits per second

CBA .... Competency-Based Assessment

CBMG's .... Customer Behaviour Model Graphs

CCIE............................. Cisco Certified Internet Expert

CCNA........................... Cisco Certified Network Associate

CCNP .... Cisco Certified Network Professional

CDROM .... Compact Disc Read Only Memory

CDP .... Cisco Discovery Protocol

CLDD .... Composite Logical Data Design

CLI……………………Command Line Interface

CIM .... Computer Installation and Maintenance

CMBGs .... Customer Model Behaviour Graphs

CMOS .... Complementary Metal Oxide Semiconductors

CNA .... Certified Novell Administrator

CNAP .... Cisco Network Academy Program

CNE.............................. Certified Novell Engineer

CNS.............................. Computer and Network Support

CNT.............................. Computer and Network Technology

CPU.............................. Central Processor Unit

CS................................. Computer Science

CSE .... Computer Science Education

CSIDs .... Client/Server Interaction Diagrams

CSM .... Computer Systems Management

DDR SDRAM.............. Double Data Rate Synchronous Dynamic Random Access Memory

DEET .... Department of Employment, Education and Training (Australia)

DFD .... Data Flow Diagram

DNS.............................. Domain Name Server

DRAM.......................... Dynamic Random Access Memory

DS................................. Digital Signalling

ECC.............................. Error Correcting Code

ECU .... Edith Cowan University

EDO .... Extended Data Out

EIGRP .... Enhanced Interior Gateway Routing Gateway Protocol

ENDEC .... Encoder/Decoder

ERD.............................. Entity Relationship Diagrams

FCS.............................. Frame Check Sequence

FCHS............................ Faculty of Computing Health and Science

FIE................................ Frontiers in Education

FSM.............................. Finite State Machine

fps................................. frames per second

FTP............................... File Transfer Protocol

GUI............................... Graphical User Interface

GPSS .... General Purpose System Simulation

HDD .... Hard Disk Drive

HTML .... Hyper Text Markup Language

HTTP............................ Hypertext Transfer Protocol

HTTPS .... Secure Hypertext Transfer Protocol

IASTED .... International Association of Science and Technology for Development

IBM .... International Business Machines

ICs .... Integrated Circuits

iCOMP .... Intel Comparative Microprocessor Performance Index

ICT .... Information and Communication Technologies

IEEE .... Institute for Electrical and Electronic Engineers

I/O .... Input/Output

IOS .... Internet Operating System

InSITE.......................... Information Science + Information Education

IP .... Internet Protocol

IS .... Information Systems

ISA .... Industry Standard Architecture

ISDN .... Integrated Services Digital Network

ISO .... International Standards Institute

IT.................................. Information Technology

ITiCSE .... Innovation and Technology in Computer Science Education

JRPIT .... Journal of Research into Information Technology

LAN .... Local Area Network

LDV .... Logical Design Volumes

MBs.............................. Mega Bytes

MB/s............................. Mega Bytes per second

mps............................... Megabits per second

MAC............................. Media Access Control

MCA............................. Micro Channel Architecture

MCQs .... Multiple Choice Questions

MCSE........................... Microsoft Certified Systems Engineer

MIMD .... Multiple instruction stream, multiple data stream

MIPS .... Millions of Instructions per Second

MIS............................... Management Information Systems

MISD............................ Multiple instruction single data stream

MFLOPS .... Millions of Floating point Operations per Second

MPP.............................. Massively Parallel Processing Systems

MPUs .... Microprocessor Units

NDM .... Network Design & Management

NFS .... Network File System

NIC................................ Network Interface Card

NIM .... Network Installation and Maintenance

NOS.............................. Network Operating System

NSF .... National Science Foundation (USA)

NUMA .... Non-uniform Memory Access Computers

OPNET.......................... Optimized Network Engineering Tools

OS................................. Operating System

OSI .... Open Systems Interconnection Model

OSPF .... Open Shortest Path First

PC................................. Personal Computer

PCI .... Peripheral Component Interconnect

PDUs .... Protocol Data Units

PING .... Packet Internet Groper

PIR .... Peak Information Rate

PR................................. Performance Rating

QoS............................... Quality of Service

RAID............................ Redundant Array of Independent Disks

RAM............................. Random Access Memory

RAS .... Remote Access Server

RF................................. Radio Frequency

ROM .... Read Only Memory

RIP .... Routing Information Protocol

RSVP............................ Resource Reservation Protocol

RTT .... Round Trip Time

rps................................. revolutions per second

SCIS .... School of Computer and Information Science

SSAD .... Structured Systems Analysis and Design Method

SCSI .... Small Systems Computer Interface

SDRAM .... Synchronous Random Access Memory

SI .... Systems Internationale

SIGada.......................... ACM Special Interest Group on Ada Programming Language

SIGCOMM................... ACM Special Interest Group on Data Communication

SIGCSE........................ ACM Special Interest Group on Computer Science Education

SIGCPR........................ ACM Special Interest Group on Computer Personnel Research

SIGMETRICS.............. ACM Special Interest Group on Measurement and Evaluation

SIMD............................ Single instruction stream, multiple data stream

SISD............................. Single instruction stream, single data stream

SMEs............................ Small and Medium Enterprises

SMP.............................. Symmetric Multiprocessor Systems

SPEC............................ System Performance Evaluation Co-operative

SSADM........................ Structured Systems Analysis and Design Method

STDM........................... Statistical Time Division Multiplexing

STP............................... Spanning Tree Protocol

TAFE............................ Tertiary and Further Education (Australia)

TCP .... Transmission Control Protocol

TFTP .... Trivial File Transmission Protocol

TPC .... Transaction Processing Performance Council

TPC-W .... Transaction Processing Performance Council (Ecommerce workload)

tps .... transactions per second

TTL .... Time To Live

TTL .... Transistor Transistor Logic

UDP.............................. Universal Datagram Protocol

UICEE........................... UNESCO International Centre for Engineering Education

UNESCO....................... United Nations Educational Scientific and Cultural Organisation

UPT .... Unshielded Twisted Pair

VBC .... Vendor Based Curricula

VLAN........................... Virtual Local Area Network

VDM .... Vienna Development Method

VTP……………………VLAN Trunking Protocol

VoIP .... Voice over Internet Protocol

WAN .... Wide Area Network

WWW .... World Wide Web

ZDT.............................. Zero Down Time