

State Model Diagrams for Teaching Networking – A SOLO Based Evaluation

Stanislaw Paul Maj¹, Kazu Ohtsuki^{1,2}, Toru Akamatsu³ & Steve MacKay¹

¹ Engineering Institute of Technology, Perth, Western Australia

² Graduate Schools of Intercultural Studies, Kobe University, Japan

³ Graduate School of Information Technology, Kobe Institute of Computing, Japan

Correspondence: Dr. Maj, S. P. Engineering Institute of Technology, 1031, Wellington St, West Perth, Western Australia, 6005. Tel: 13-0013-8522. E-mail: paulm@eit.edu.au

Received: April 20, 2016

Accepted: June 16, 2016

Online Published: July 27, 2016

doi:10.5539/mas.v10n10p231

URL: <http://dx.doi.org/10.5539/mas.v10n10p231>

Abstract

The Cisco Network Academy Program is the world's largest network technology curriculum with over 500,000 students' worldwide. An analysis of this curriculum found that it did not provide students with a conceptual model that is consistent throughout the curriculum. Without such a pedagogical model students develop their own which is likely to be incomplete, inconsistent and incorrect. State Model Diagrams (SMDs) were designed to address this problem. SMDs allow networking concepts and technical detail to be taught using a single common template. Research has clearly demonstrated that SMDs considerably improve student learning at both introductory and advanced levels. As a diagrammatic technique SMDs are potentially language independent – an important aspect of any global curriculum. This paper investigates this hypothesis and using the SOLO taxonomy further evaluates SMDs as a pedagogical method for teaching network technology.

Keywords: state model diagrams, Cisco, network technology education

1. Introduction

1.1 The Cisco Network Academy Program

In the late 1990's Cisco introduced, at a cost of US\$75 million, the Cisco Network Academy Program (CNAP). The CNAP curriculum is designed for schools, colleges of further and advanced education, universities and also practicing professionals (Veal, Kohli, & Maj, 2005). The CNAP curriculum consists of a number of awards. The Cisco Certified Network Associate (CCNA) assumes no knowledge of networking and is the prerequisite to the more advanced Cisco Certified Network Professional (CCNP). Within the school sector typically only the first two semesters of the CNAA are taught. Vendor based curriculum within the university sector has both advocates and critics (Abelman, 2000; Maj, S. P., & Dharukeshwari, 2003). The CNAP website provides the entire curriculum along with other training material (Murphy, Kohli, Veal, & Maj, 2004). It is currently the world's largest network technology curriculum.

However an extensive analysis by Maj found that the CNAP curriculum is based primarily on the hierarchical text-based Command Line Interface (CLI) (Maj, S. P. & Kohli, 2004). To obtain operational data from a network device several CLI commands may be needed, the output of which may be complex. Furthermore the CNAP curriculum does not provide a single coherent conceptual model of the many different network devices and associated protocols. This is problematic because if students are not taught using a conceptual framework they may develop their own which is likely to be incomplete, inconsistent and incorrect. All of these problems are compounded because there are different types of network devices and many associated protocols. According to Von Glasserfeld teachers need adequate models as the basis of instruction (von Glasersfeld, 1982).

1.2 Modelling Complexity

Even the simplest network is complex as each device is concurrently running a number of different protocols – all interacting. Abstraction is used to control complexity. The ACM/IEEE Computing Curriculum 2001 lists abstraction as one of twelve recurring concepts fundamental to computing,

“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. Examples include levels of hardware description, levels of specificity within an object hierarchy...” (Tucker et al., 1991)

Models are based on abstraction. The importance of a diagrammatic model as an aid to student learning is illustrated by Thomas, “The use of diagrammatic representation provides an alternative to just offering more words, which may only compound their difficulties” (Thomas, 2000). Furthermore, 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).

In order to determine the operation of a single router specific key information is needed: interface Internet Protocol (IP) and Medium Access Control (MAC) addresses; interface line status (triggered by carrier detect signal); interface line protocol status (triggered by Keep Alive Frames); Address Resolution Protocol (ARP) details and routing table entries. This information is typically obtained from four CLI commands – show interface fa0/1, show interface fa0/0, show Arp and show IP route. Actual output, from an operational router, for such commands is as follow:

```
Router1#show interface fa0/1
```

```
FastEthernet0/1 is up, line protocol is up
```

```
Hardware is AmdFE, address is 000c.30e2.e501 (bia 000c.30e2.e501),
```

```
Internet address is 192.168.1.1/24
```

```
MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec, reliability 255/255, txload 1/255, rxload 1/255
```

```
Encapsulation ARPA, loopback not set, Keepalive set (10 sec), Full-duplex, 100Mb/s, 100BaseTX/FX
```

```
ARP type: ARPA, ARP Timeout 04:00:00
```

```
Last input never, output 00:00:09, 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
```

```
0 packets input, 0 bytes, Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
```

```
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignore, 0 watchdog,
```

```
0 input packets with dribble condition detecte, 68 packets output, 8770 bytes, 0 underruns
```

```
0 output errors, 0 collisions, 2 interface resets, 0 babbles, 0 late collision, 0 deferred
```

```
3 lost carrier; 0 no carrier, 0 output buffer failures, 0 output buffers swapped out
```

```
Router1#show interface fa0/0
```

```
FastEthernet0/0 is up, line protocol is up
```

```
Hardware is AmdFE, address is 000c.30e2.e500 (bia 000c.30e2.e500)
```

```
Internet address is 192.168.2.1/24
```

```
MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec, reliability 255/255, txload 1/255, rxload 1/255
```

```
Encapsulation ARPA, loopback not set, Keepalive set (10 sec), Full-duplex, 100Mb/s, 100BaseTX/FX
```

```
ARP type: ARPA, ARP Timeout 04:00:00
```

```
Last input 00:00:23, output 00:00:05, 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
```

```
21 packets input, 5214 bytes, Received 20 broadcasts, 0 runts, 0 giants, 0 throttles
```

```
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignore, 0 watchdog
```

```
0 input packets with dribble condition detecte, 68 packets output, 8751 bytes, 0 underruns
```

```
0 output errors, 0 collisions, 2 interface resets, 0 babbles, 0 late collision, 0 deferred
```

```
3 lost carrier; 0 no carrier, 0 output buffer failures, 0 output buffers swapped out
```

```
Router1#show arp
```

Protocol	Address	Age (min)	Hardware Addr	Type	Interface
Internet	192.168.2.2	1	0012.01a1.cda1	ARPA	FastEthernet0/0
Internet	192.168.1.1	-	000c.30e2.e501	ARPA	FastEthernet0/1
Internet	192.168.2.1	-	000c.30e2.e500	ARPA	FastEthernet0/0

```
Router1#show ip route
```

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
 D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
 N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
 E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
 i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
 ia - IS-IS inter area, * - candidate default, U - per-user static route
 o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C 192.168.1.0/24 is directly connected, FastEthernet0/1
 C 192.168.2.0/24 is directly connected, FastEthernet0/0
 R 192.168.3.0/24 [120/1] via 192.168.2.2, 00:00:03, FastEthernet0/0

In order to help control this complexity and hence better understand what is happening within the device, Maj proposed a diagrammatic abstraction called State Model Diagrams (SMDs) (S.P Maj & Kohli, 2004). State Model Diagrams are based on a state-oriented approach to specifying system behavior (Peters & Pedrycz, 2000). The standard approach to this type of specification is the finite-state machine which can be used for modeling communications protocols (Halsal, 1996)]. A single SMD can be used to represent the main data extracted from these four separate CLI commands and link this information to the appropriate OSI level (Figure 1). Significantly it is possible to apply the same basic principles to other routing protocols.

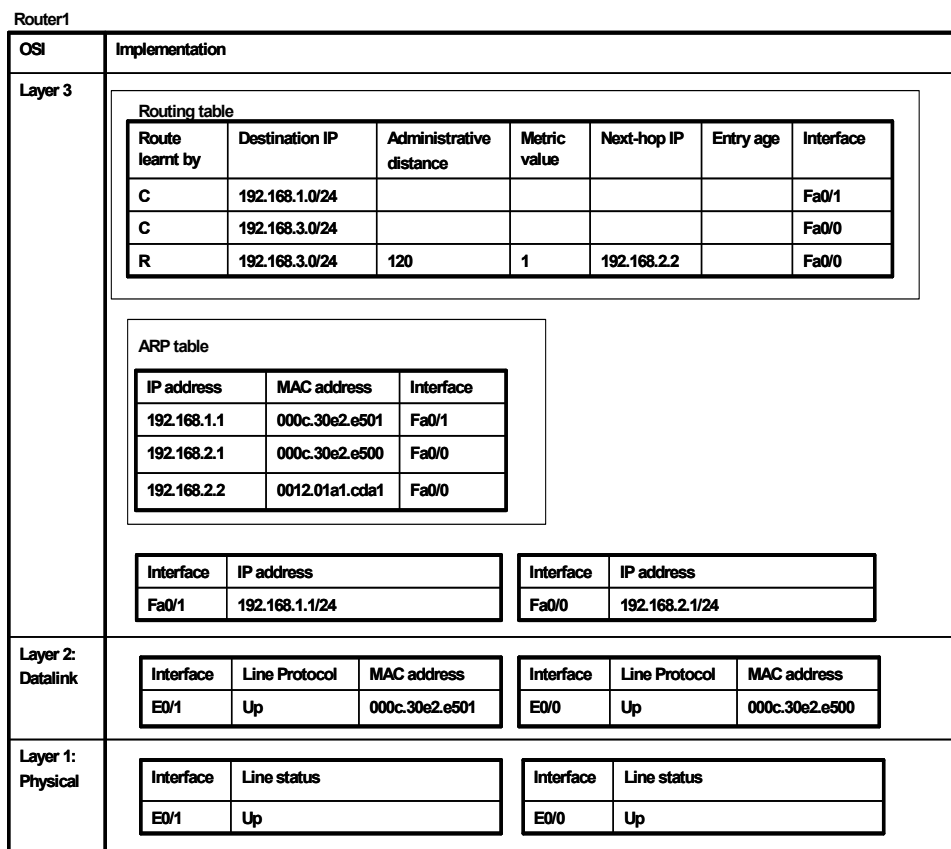


Figure 1. State Model Diagram of a Router (RIP)

“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 (Interior Gateway Routing Protocol – IGRP), link state (Open Shortest Path First–OSPF) and balanced hybrid (Enhanced Interior Gateway Routing Protocol – 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)”. (Maj, S. P., Kohli, & Murphy, 2004).

Furthermore network security devices and protocols can also be modeled using State Model Diagrams (Nuangjamnong, 2007), (Kohli, Maj, Murphy, & Veal, 2005), (Maj, S. P., Makasiranondh, W., & Veal, D., 2010), (Maj, S. P. & Veal, D., 2010). Significantly SMDs may be also used for teaching network management (Maj, S. P. & Tran, 2006b). Furthermore it has been demonstrated that there are potentially considerable pedagogical benefits to their use (S. P. Maj & Tran, 2006a), (Kohli, Veal, Maj, & Murphy, 2005; Maj, S. P. & Tran, 2007). According to Maj,

“SMDs allow networking concepts and technical detail to be taught using a single common template. Technical details may be progressively included whilst maintaining conceptual integrity by means of hierarchical leveling. SMDs may therefore support student learning at both introductory and advanced levels. In effect students do not have to learn a new conceptual model; rather they build upon and extend existing knowledge. In this context new knowledge reinforces existing knowledge.” (Maj, S. P. & Veal, 2007).

Initial learning is typically based on real world objects – what does it look like? Novice students are assisted in the conceptualization process by including in a single SMD both physical details (e.g. connectors) along with the conceptual information. This is an important initial step. A network diagrammatically modeled using SMDs can directly link the concrete (the devices and connectors) to the abstract (protocols etc) SMDs are a diagrammatic means of teaching as opposed to the sequential text based CLI. Intrinsically therefore it is possible using SMDs to demonstrate the relationships and interactions between network devices and their protocols. This is important because deep learning requires students to think not only about multiple things but also how those objects interrelate. A single SMD can represent the output from nine separate CLI commands and hence show concurrent interactions between different devices and associated protocols. In effect SMDs are designed to assist students make the transition to deeper learning e.g. from multistructural to relational. In a study by Maj, students taught using SMDs were evaluated six weeks after the final examination, it was found that:

“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.” (Maj, S. P., Kohli, & Fetherston, 2005).

Cognitive taxonomies such as Blooms may be used to evaluate student performance (Bloom, Engelhart, Furst, Hill, & Krathwohl, 1956). A limited study has demonstrated that curriculum, based on SMDs, results in student learning up to Bloom’s fourth level (analysis), (Maj, S. P. & Tran, 2007). One study suggests that SMDs may be modified or extended (XSMD) and support higher order learning for postgraduate Japanese students (Akamatsu, 2007). The Bloom taxonomy has both its advocates and its critics – a debate beyond the scope of this paper. An alternative approach to classifying learning is the Structure of Observed Learning Outcomes (SOLO) Taxonomy with four category definitions (Biggs & Collis, 1989). The SOLO taxonomy was used in this research. As a diagrammatic technique SMDs are potentially language independent – an important aspect of any global curriculum. In order to evaluate this hypothesis SMDs were used as the basis of instruction in Japanese.

2. Method

2.1 Participating Population

An introductory lecture on routing based entirely on SMDs was translated to Japanese. This lecture was then given remotely, on-line by a native Japanese speaking student to twenty students in Japan. The student presenter was assisted by an academic. All twenty students in this group were enrolled on a postgraduate course. All these students were graduates but more than half of them did not have an undergraduate qualification in computer science. Prior to the on-line lecture all these students had received instruction (theory and practice) in computer network technology all based on the Linux operating system.

2.2 Method

After the on-line presentation all the participants were given a questionnaire (in Japanese) that included a brief explanation of the SOLO Taxonomy method and the associated definitions of the unistructural, multistructural and relational learning categories. The questionnaire was based on a Likert scale (Strongly Agree; Agree; Neutral; Disagree; Strongly Disagree) along with a request for any comments. All recorded comments were written in Japanese and then translated to English. The participants were asked to implement a simple network consisting of two PC Linux routers connecting together two end user PCs and complete an associated SMD.

2.3 Results

Of the twenty students nineteen completed the questionnaire and one student gave a null response to all

questions and was excluded from the results. The data along with all verbatim comments were as follows:

Table 1. State Model Diagrams assist in relational learning

SA	A	N	D	D	SD	Null
39%	56%	5%				

Comments: “I agree that the model could be useful in relational learning. Yet basic knowledge should be required to achieve the level of relational learning. If the state model diagram assist in relational learning depends on how that basic knowledge was obtained.”

“Is the state model the one shown on-line presentation from Australia? I will answer the questions assuming that the state model is the diagrams shown on the presentation.”

Table 2. State Model Diagrams are language independent

SA	A	N	D	D	SD	Null
33%	50%	17%				

Table 3. State Model Diagrams explicitly link network concepts in a single diagram

SA	A	N	D	D	SD	Null
39%	44%	11%				6%

Comments: “The State Model is a good table that lists state of devices and it help student link network concepts. However, a diagram which has image with explanation should also be required to facilitate clearer understanding.”

“It helps to understand the difference between the 2nd layer and 3rd layer.”

Table 4. State Model Diagrams control complexity

SA	A	N	D	D	SD	Null
61%	17%	11%	6%			6%

Comments: “The state model shows the outcomes of certain commands and relationships of devices. In practice, however, we need to know the commands and file manipulations to get those outcomes. Since the model don’t show the latter, it is not necessary that the model control complexity.”

“I think the state model helps to understand and control networking since it changes the text results to a diagram.”

Table 5. State Model Diagrams can be used to model different network devices and protocols

SA	A	N	D	D	SD	Null
56%	33%	11%				

Comments: “It should be used to model various network devices and protocols.”

Table 6. Using State Model Diagrams it is possible to see the interaction between different devices and protocols in a single diagram

SA	A	N	D	D	SD	Null
44%	39%	11%				6%

Comments: “It is possible although there may be limitation.”

“It might be hard to understand TCP/IP and network devices without some explanation by someone who understand them even though the diagram is presented. However, it is possible to say that it is easier to understand networking using the state model diagrams than using only text.”

Table 7. State Model Diagrams make the principles of networking clearer

SA	A	N	D	D	SD	Null
56%	39%		6%			

Comments: “The model alone cannot help. However, it could be very useful with the effective use of figures and explanations.”

Table 8. Using State Model Diagrams may be used to teach networking using Linux

SA	A	N	D	D	SD	Null
50%	33%	11%				6%

Comments: “It is possible since there is no big difference between Linux and Windows.”

“I think the state model diagrams works well to understand networking using Linux.”

Table 9. State Model Diagrams represent a template that can be used to teach different network technologies and protocols

SA	A	N	D	D	SD	Null
44%	50%	6%				

Comments: “The model works well as a template.”

Table 10. State Model Diagrams link the physical device to a conceptual model

SA	A	N	D	D	SD	Null
44%	50%	6%				

Comments: “I am not sure if the model can link the physical device to a conceptual model perfectly.”

2.4 Data Analysis

As a global curriculum the CNAP is offered in a number of different languages. However, it is not uncommon for students whose first language is not English to study using the English based CNAP materials. Language, as an instructional vehicle is important, but depends on a good understanding of grammatical rules. Subtleties of the language used in instruction and assessment may lead to students not understanding some material. For example, the following CNAP multi-choice question assumes readers are familiar with the use of a “double negative”;

Which of the following is not a feature of switches?

- A. They operate at Layer 3 of the OSI model
- B. They are more intelligent than hubs
- C. They do not make any forwarding decisions
- D. They build and maintain address tables

By contrast SMDs are based on tables and numbers. Hence these diagrams are substantially language independent and hence less reliant upon the understanding of finer points of the English language. Students are required to extract data from CLI output commands and populate the SMD. In order to assess deeper relational learning students may be presented with five connected routers but the CLI output for only two of them. In order to complete this exercise students would need to understand not only how routing works, but have sufficient understanding of routing in order to be able extrapolate and hence predict the data entries for the other routers.

One of the objectives of this research was to determine if SMDs are substantially language independent (question 2). In response this question 83% of the participants either strongly agreed or agreed. Three students (17%) were neutral and no comments were recorded. However, in Japan technical English terms are referred to as ‘technical expressions’ and accepted as different to Japanese and as such may not represent a major linguistic problem. The term ‘language independent’ was not explained in the questionnaire given to the participants. An alternative, but equally valid, interpretation of this expression is that the diagrams are independent of the operating system language e.g. Linux or the Command Line Interface. The results are therefore inconclusive and further work is needed.

This research also sought to determine if SMDs support higher order learning as defined by the SOLO taxonomy. Significantly 95% of respondents either strongly agreed or agreed that SMDs assist in relational learning (question 1). This finding is supported by similar results from other related questions. Questions 3, 6, 7 and 10, in effect, ask the same question but in different terms. For questions 3 and 6, 83% of the participants either strongly agreed or agreed. Positive responses from questions 7 and 10 were 95% and 94% respectively. Previous research suggests that SMDs may be used as a common template for teaching different network devices and protocols. The responses to question 5 and 9 (89% and 94%) further support this hypothesis.

There are different but equally valid ways of teaching network technology. The CNAP curriculum is based on configuring networks based on devices such as routers, switches etc. An alternative, approach is to use the Linux operating system – as used by all participants in this survey. SMDs are designed to be an abstraction that can be used regardless of the networking technologies being used. The majority of participants (83%) indicated that SMDs could also be used to teach networking using Linux.

As a further test of this hypothesis, all participants implemented a Linux based network and attempted to complete the associated SMD. All twenty participants completed the SMD details for IP and MAC addresses which demonstrates SOLO taxonomy multistructural learning. All twenty participants completed the SMD details for the ARP and routing tables. This suggests that they can integrate the different parts of a network into a coherent whole with structure and meaning i.e. SOLO taxonomy relational learning. Significantly these students had been given only one lecture on SMDs.

The diagrams produced by the participants consisted of a mixture of Japanese language annotations and associated numerical data such as device addresses and interface numbers (Figure 2, Figure3). Despite the use of Japanese, even a non-Japanese speaker can read these SMDs and examine how a student has extracted the relevant data and correctly incorporated it into a conceptual scheme. A few students made a few simple errors which may be attributed to simple transcription errors.

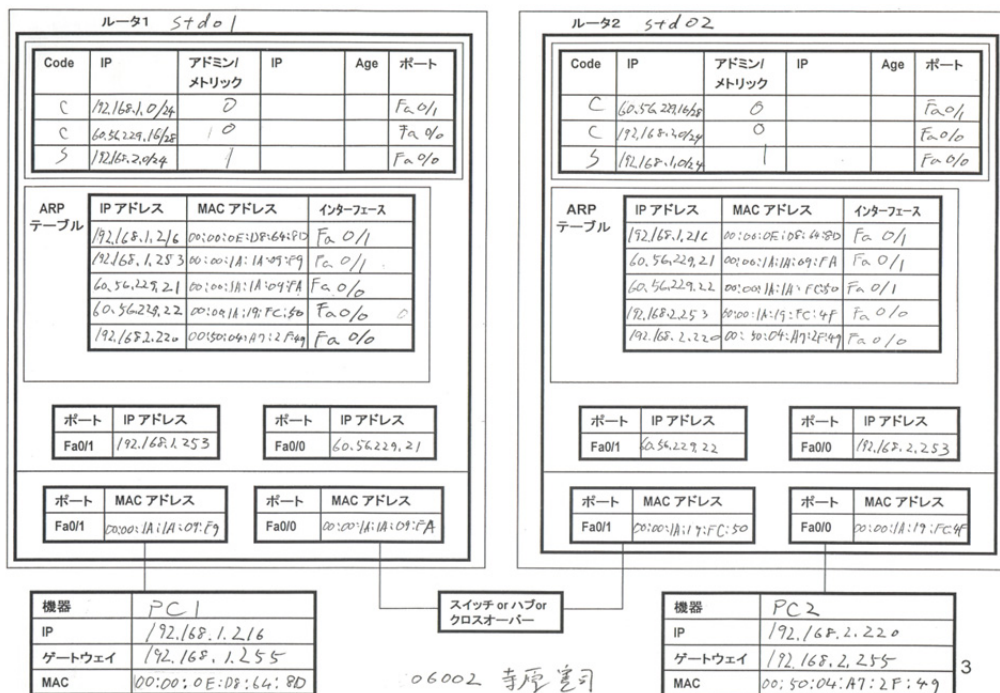


Figure 2. Completed SMD by Japanese student #06002

Student #06002 (Figure 2) successfully built an operational network. This is evident because the SMD can only be completed from data extracted from the network devices. There are however some mistakes. For example PC has been given a gateway IP address of 192.168.1.255 which is incorrect as this is a broadcast address. Even though the SMD is substantially correct there are some mistakes that would suggest a significant conceptual misunderstanding. The ARP table in routing device std01 incorrectly holds a mapping for the IP address and associated MAC address of PC2. Successful completion of the SMD indicates evidence of deep SOLO relational learning. Relational learning indicates that several aspects are integrated to that the whole has a coherent structure and meaning.

By contrast student #506001 (Figure 3) only partially completed the SMD. From the SMD it is evident that this student either did not have an operational network or they did not understand how a network works. This student displays, at best, only very limited surface SOLO unistructural learning. Unistructural learning indicates that one aspect of a task is understood serially, but there is no relationship of facts or ideas. This student has correctly assigned an IP address to PC1 and identified the MAC address of this device. However the PC gateway IP address is incorrect. For PC1 to form a connection to the adjacent router the Address Resolution Protocol (ARP) table must resolve the IP to MAC address mappings. This student has only partially completed the router ARP table and the entries are incorrect. These errors would suggest this student almost certainly has a very limited understanding of some of the key concepts of networking.

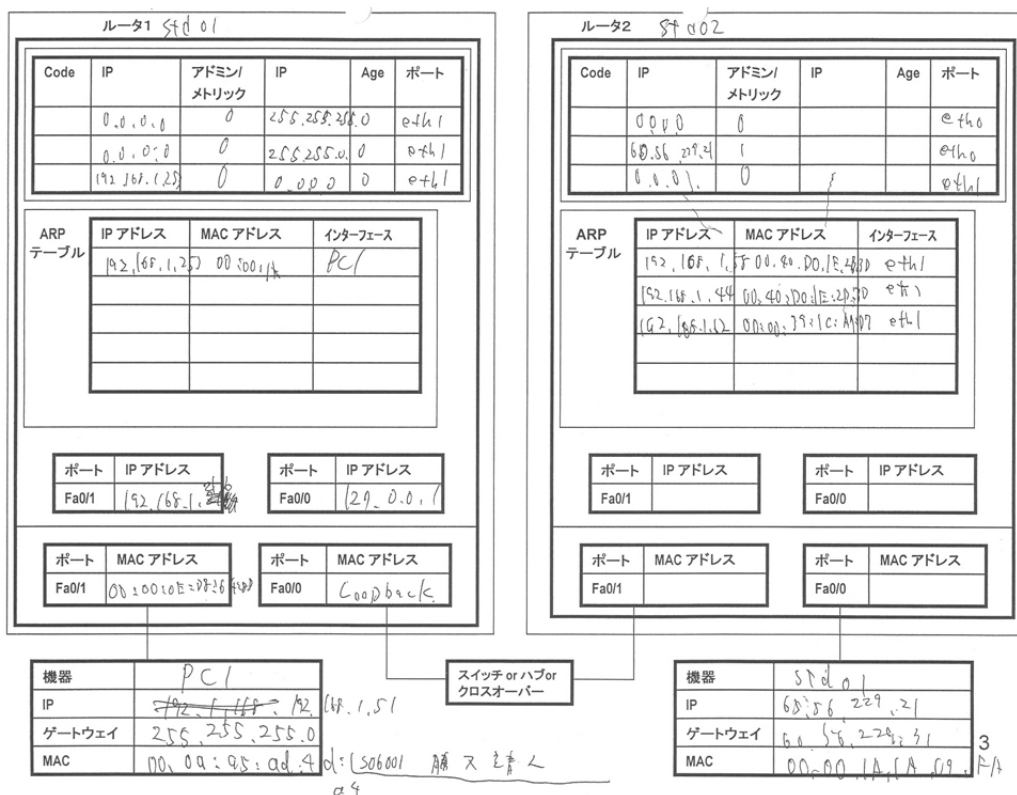


Figure 3. Completed SMD by Japanese student #06002

The Method section describes in detail how the study was conducted, including conceptual and operational definitions of the variables used in the study, Different types of studies will rely on different methodologies; however, a complete description of the methods used enables the reader to evaluate the appropriateness of your methods and the reliability and the validity of your results, It also permits experienced investigators to replicate the study, If your manuscript is an update of an ongoing or earlier study and the method has been published in detail elsewhere, you may refer the reader to that source and simply give a brief synopsis of the method in this section.

4. Discussion

SMDs have been successfully used as the basis of teaching network technology. Significantly SMDs are a diagrammatic method for controlling the complexity associated with teaching networking. Using SMDs it is possible to extract data from the complex CLI commands and hence clearly demonstrate concurrent interactions between different devices and their associated protocols. In effect SMDs are designed to assist students make the transition to deeper learning e.g. from multistructural to relational. This limited study indicated that, as a diagrammatic method SMDs are substantially independent of the language of instruction. However, there may be ambiguity in the use of this term. Hence further work is needed. However, all the participants in this experiment successfully implemented a Linux based network and completed the associated SMD. Completion of this SMD is an indication of deep SOLO relational learning. Using the completed SMDs it was possible to determine that one participant had not completed all the entries indicative of possible conceptual understandings. Despite the fact that the completed SMDs were written in a mixture of Japanese and numerical values it was possible for a non-Japanese speaker to interpret the results. A further, more extensive investigation is needed.

References

- Abelman, C. (2000). A parallel universe. *Change*, 32(3), 20-29.
- Akamatsu, T., Ohtsuki, K., & Maj, S. P. (2007). *A teaching model for computer network technology in professional graduate schools*. Paper presented at the 8th International Conference on Information Technology Based Higher Education and Training, Kumamoto, Japan.
- Biggs, J., & Collis, K. (1989). Towards a Model of School-based Curriculum Development and Assessment Using the SOLO Taxonomy. *Australian Journal of Education*, 33(2), 151-163.
- Bloom, B. S., Engelhart, M. D., Furst, E. J., Hill, W. H., & Krathwohl, D. R. (1956). *Taxonomy of Educational Objectives - The Classification of Educational Goals*. New York: David McKay Company Inc.
- Gilbert, J. (1995). *The role of models and modelling in some narrative science learning*. Paper presented at the Annual meeting of the American Educational Research Association, San Francisco, CA.
- Halsal, F. (1996). *Data Communications, Computer Networks and Open Systems*. Harlow, England: Addison-Wesley.
- Kohli, G., Maj, S. P., Murphy, G., & Veal, D. (2005). *A Conceptual Model as an aid to understanding Network Security*. Paper presented at the 2005 American Society for Engineering Education Annual Conference & Exposition (ASEE 2005), Portland, Oregon.
- Kohli, G., Veal, D., Maj, S. P., & Murphy, G. (2005). *The integration of state diagrams with competency based assessment*. Paper presented at the 2005 American Society for Engineering Education Annual Conference & Exposition (ASEE 2005), Portland, Oregon.
- Maj, S. P., & Dharukeshwari, J. (2003). Vendor Based Network Engineering Education - An International Comparison. *World Transactions on Engineering and Technology Education*, 2, 313-316.
- Maj, S. P., & Kohli, G. (2004). A New State Models for Internetworks Technology. *Journal of Issues in Informing Science and Information Technology*, 1, 385-392.
- Maj, S. P., Kohli, G., & Fetherston, T. (2005). *A pedagogical evaluation of new state model diagrams for teaching Internetwork technologies*. Paper presented at the 28th Australasian Computer Science Conference (ACSC2005), Newcastle, Australia.
- Maj, S. P., Kohli, G., & Murphy, G. (2004). *State Models for Internetworking Technologies*. Paper presented at the IEEE, Frontiers in Education, 34th Annual Conference, Savannah, Georgia, USA.
- Maj, S. P., Makasiranondh, W., & Veal, D. (2010). An Evaluation of Firewall Configuration Methods. *IJCSNS International Journal of Computer Science and Network Security*, 10(8), 1-7.
- Maj, S. P., & Tran, B. (2006a). *Global curriculum on a budget*. Paper presented at the EDU-COM (2006), Engagement and Empowerment: New Opportunities for Growth in Higher Education, Nong Khai, Thailand.
- Maj, S. P., & Tran, B. (2006b, 2007). *State Model Diagrams - a Systems Tool for Teaching Network Technologies and Network Management*. Paper presented at the International Joint Conferences on Computer, Information and Systems Sciences, and Engineering, University of Bridgeport.
- Maj, S. P., & Tran, B. (2007). *Network Technology Education: A Novel pedagogical model for novices and practicing professionals*. Paper presented at the Teaching and Learning Forum (2007), UWA, Perth, Western

Australia.

- Maj, S. P., & Veal, D. (2007). State Model Diagrams as a Pedagogical Tool - An International Evaluation. *IEEE Transactions on Education*, 50(3), 204-207.
- Maj, S. P., Veal, D. (2010). An Evaluation of State Model Diagrams for Secure Network Configuration and Management. *IJCSNS International Journal of Computer Science and Network Security*, 19(9).
- Murphy, G., Kohli, G., Veal, D., & Maj, S. P. (2004). *An Examination of Vendor-Based Curricula in Higher and Further Education*. Paper presented at the 2004 American Society for Engineering Education Annual Conference & Exposition (ASEE 2004), Salt Lake City, Utah.
- Nuangjammong, C., Maj, S. P., & Veal, D. (2007). *Network Security Devices and Protocols Using State Model Diagrams*. Paper presented at the 5th Australian Information Security Management Edith Cowan University, Perth, Western Australia.
- Peters, J. F., & Pedrycz, W. (2000). *Software Engineering - An Engineering Approach*. New York: John Wiley & Sons.
- Thomas, R. (2000). *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.
- Tucker, A. B., Barnes, B. H., Aiken, R. M., Barker, K., Bruce, K. B., Cain, J. T. ... Mulder, M. C. (1991). A Summary of the ACM/IEEE-CS Joint Curriculum Task Force Report, Computing Curricula (1991). *Communications of the ACM*, 34(6).
- Veal, D., Kohli, G., & Maj, S. P. (2005). *Cisco Based Curricula in University Units*. Paper presented at the 2005 American Society for Engineering Education/Australasian Association for Engineering Education (ASEE/AAEE) 4th Global Colloquium on Engineering Education, Sydney, NSW, Australia.
- von Glasersfeld, E. (1982). An interpretation of Piaget's constructivism. *Revue Internationale de Philosophie*, 36(4), 612-635.

Copyrights

Copyright for this article is retained by the author(s), with first publication rights granted to the journal.

This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).