The
Adventure

369 plans for its new program!

Calendar of Events:

2/6/00	Scout Sunday
2/12/00	Awards Banquet
2/26/00	Maple Sugar Festival
3/4/00	Maple Sugar Festival
3/7/00	Bring a 6' section of Natural Rope
3/14/00	Perl Classes Begin
3/52/00	Articles are due for The Adventure
3/28/00	Bring two 6' sections of Natural Rope
5/13-14/00	Flower Planing at Murfield
5/20-21/00	Flower Planting at Murfiled
6/24/00	Wyandotte Lake, Scout Family Fun Day
6/9/00	Sleepover
6/10/00	Garage Sale
6/24/00	Wyandot Lake
7/2-8/00	Summer Camp
7/4/00	No Meeting
9/19/00	Open House [First Nighter]
10/13-15/00	Book Binding Campout [Exchange Lodge]
10/31/00	Pizza Party
12/15/00	Silver Beaver Apps Due
12/19/00	Christmas Party
12/26/00	No Meeting
12/28/00	Leave for New Zealand

Venturing into Perl
James D. Corder

Venturing Crew 369 will be offering a new course in February for its "Red" students: Introduction to Perl for Programmers." This new 356 page book goes to the printer in January.

I am proud of 369 for its ability to produce a book/course a year. This influx of new material keeps our graduates on the cutting edge.

I look forward to sitting in on some of Mr. Potters Classes next year!

369 prepares for 2001
James D. Corder

Due to the unsurpassed growth of 369 the courses and meeting structure must change. Oh, what a good problem to have!

In the past 369 has taken a group of youth [age 16] and stayed with them until they got a job [age 20]. Then 369 would open membership and start the ball rolling again. Due to the increase size of both the youth waiting list and the ever growing number of corporations asking to use our members as interns we need to begin teaching multiple classes simultaneously.

Venturing Crew 369's meetings will be divided into a 5 week rotation depending on what Tuesday of the week it is:

- 1st   Munchie Night - and a leadership module
- 2nd   Technical module
- 3rd   Social Night
- 4th   Outdoor Training
- 5th   Guest Speaker

Web Status, For January

82511	*6787*	*Bible*
104512	16685	Venturing
194638	26497	Scouting
124589	14778	ExpNews
78928	6078	Links
97700	9413	Calendar
142800	10527	Members
79236	7431	Adults
5497	1094	Toadies
34958	2800	Youth
47520	4786	Program
1461	564	Project

Venture Code

I believe that America's Strength lies in her trust in God and in the courage and strength of her people.

I will, therefore, be faithful in my religious duties and will maintain a personal sense of honor in my own life.

I will treasure my American heritage and will do all I can to preserve and enrich it.

I will recognize the dignity and worth of my fellowmen and will use fair play and goodwill in dealing with them.

I will acquire the venturing attitude that seeks the truth in all things and adventure on the frontiers of our changing world.

■

Crew Finances

Topic	Need	On-Hand
The Adventure	$95,000	$375.00
General Fund	$3,000	$1,002.65
Floor Fund	$2,500	$2,500.00
Electrical Fund	$2,500	$2,500.00
Flag Fund	$1,000	$0.00
Room Fund	$3,800	$0.00
Grand Total		**$6,452.65**
In the Bank		**$6,000.00**
Cash on Hand		**$2.65**

Up-an-Coming Crew Expenses

12/01/00	Crew Charter	$30.00
12/01/00	Crew Insurance	$250.00
12/31/00	Registration	$675.00
Monthly	The Adventure	$75.00

Up-an-Coming Member Expenses

09/05/00	Registration	$25.00
09/26/00	Book	$25.00
06/06/00	Summer Camp	$175.00

Quote of the Month

Any fool can risk his Life.

But a hero risks his hart.

exp369@post369.columbus.oh.us

Our Principals:

1) Honor before all else.
2) The difference between a winner and a loser is that the winner tried one more time.
3) K.I.S.M.I.F.
4) Y.C.D.B.S.O.Y.A.

Our Creed:

Exploring: Enthusiasm, Energy, & Excellence.

Venture Crew 369:

Venturing Crew369 was chartered on December 31, 1994 to the Reformation Lutheran Church.

Venturing Crew369 specializes in UNIX for Programmers while emphasizing a deep theme of Engineering Computer Information & Science through an Entrepreneurial Spirit. Membership in Venturing Crew 369 is open to young men and women between the ages of 14 [and in high school] and not yet 20. Annual Membership fees are $25

Our Web Page:

http://post369.columbus.oh.us

Our E-Mail Addresses

Adults Members:

James D. Corder[E]	www.corder .com
Don Corder Sr.	Committee Member
David J. Alden	Consultant
Herb Docken	Institutional Representative
Andy Drake	drakea@nationwide.com
Rick Gauger	Committee Member
Terry Jones	Committee Member
Ralph Maurer[E]	Committee Chairman
Roy Niedzielski[E]	Associate Advisor
Tom Niedzielski[E]	Committee Member
Erika Orrick	Consultant
Lee Orrick	Associate Advisor
Stephen Potter	spp@colltech.com
Joe Harvey[E]	joharve4@vt.edu

Youth Members:

Suhasini Aswath	Aswath.2@osu.edu
Nicholas R. Carter	gamespymonkey@copper.net
VP-Neil A. Coplin	coplin.7@osu.edu
Aaron M. Croyle	Croyle.6@osu.edu
Ian B. Cunnyngham	kriegman@columbus.rr.com
Jason B. Cunnyngham	toten@columbus.rr.com
Bryan P. Fierst	-
Nathaniel P. Graham	ngraham@iwaynet.com
Mark C. Hamilton[E]	Hamilton.385@osu.edu
S/T-Ho-Sheng Hsiao	hsiao.39@osu.edu
Nathan C. Lee	Lee.1570@osu.edu
Jesse Kass	jeskas@apexmail.com
Stepan K. Kazakov	kazakov_s@yahoo.com
Nathan C. Lee	NateReflects@hotmail.com
Thomas R. Lowers	tom@wclcorp.com
David McIlroy[E]	davidmac@mit.edu
Ethan E. Metsger	Metsger.2@osu.edu
Daniel Morris	-
James J. Scherer[E]	Skyshark44@aol.com
Jon Schlegel	jschlege@vt.edu
Bill Schwanitz	type_O-@columbus.rr.com
Dave Scruby	-
Jack Trout	witmore@netzero.com
Nabeel A. Yousef	-

Toadies:

Justin Harris	Plebe
Aaron Morris	Toadie
Justin	Plebe

(E)	Eagle Scout
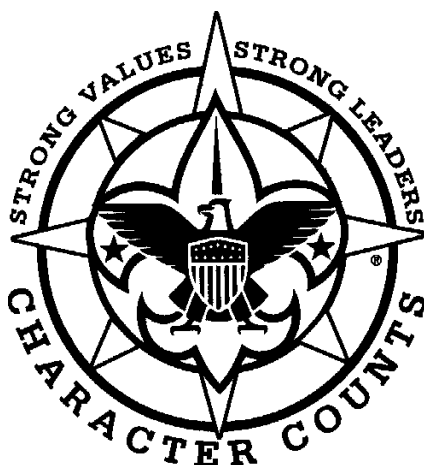
Venturering Crew 369
P.O. Box 307218
Gahanna, Ohio 43230
United States of America

We are looking for a new sponsor for The Adventure, Can you help?

My Thanks for Crew 369

Aaron M. Croyle   croyle.6@osu.edu

Wow, for the first time in my life I actually feel as though I have direction. Up until this point in my life I had just been cruising along, but now that I've found this fine group of gentlemen, I finally see myself going somewhere. Special thanks are in order for the crew's advisor, Mr. James D. Corder. For the first time in my life I feel I have a role model. Until now, I hadn't found anyone worth being like. I didn't want to be a doctor, or a sports star, or even like my parents who were a great influence on my life. I just rolled on through life doing what I though right and being the best I could. Well, in order to really do your best or try your hardest you have to have and end in mind, and finally I have one. Great thanks to the guys (and now ladies) of Crew 369, and to Mr. Corder.

Fun with ed

Bill Schwanitz

Often I have found that people forget about the simple yet powerful utilities. I am sure that if I asked a Unix admin about the ed editer, they would respond by saying something along the line of "why do you use that editer, use vi..." or something like that.

Well, I found an interesting and potentially powerful use for ed. Recently, I created a package of WU-ftpd 2.6.0 on my Sun box. I wanted to be able to install this package and not have to mess with modifing things like the hostname variable in the /etc/ftpaccess file. I came up with the idea of inserting hooks in a generic config file for this purpose. Next, I decided that I did not want to run something like sed on the file because it took extra steps, redirection. I figgured, lets use ed, I need the practice anyway.

I also noted many people forget the power of redirection. In my finnish script for the package, I have it grab information about the machine. In the finnish script, I have some variables which are obtained, specificly the hostname of the machine (possible dns in the future.)

I next went into ed and fiddled around with making my modifications noting the commands used. I then made a shell script which will insert this information into a file, which it creates. The next step after that is to invoke ed on the specified file, redirecting in the 'command' file. Take a look at the script below.

```
HOSTNAME=`/usr/bin/hostname`
echo "configuring /etc/ftpaccess for
    host ${HOSTNAME}"
echo "1,\$s/CHANGEME/${HOSTNAME}" >
    ed.commands
echo "w" >> ed.commands
echo "q" >> ed.commands
/usr/bin/ed /etc/ftpaccess < ed.com-
    mands || echo "failed to modify
    ftpaccess file, do it your-
    self!"
```

This will make the modification I want to the hostname variable in the ftpaccess file.

This was kind of a neat exploration of ed. I have not used it much lately, kinda good to know in a pinch.

Y2k and we are all are still here

Jack Trout Witmore@netzero.com

The "Y2K bug" Feared by all. Yet almost nothing truely failed. There was a stop light that blew out in austrialia. wait that wasn't y2k. There was those power outages in ohio. Never Mind that was line failure.

Why did people make it so big? I think they fear the unknown. most of the people throwing stones to at the computer world were the computer illiterate. These people listen to the news who want to sell thier papers and shows. Saying that computers will fail and the world will. Not because they have any basis but because it sells.

The basic consumer sees some big words and the front page articles and begins to worry. They ask people about it. They are assured that nothing will happen by people they trust. They continue to worry and believe it will be a disaster. Why? because the paper said so. When the clock struck 12 on 1-1-2000 nothing happens. The radios were trying to recover themselves by saying that the effects might come in february. Lets just wait and see. I still stand on the side that knows nothing major will fail..

Bob's Kingdom
By: Daniel Morris

Hi, Bob. What half-baked scheme have you come up with now?

I'm goign on a world tour.
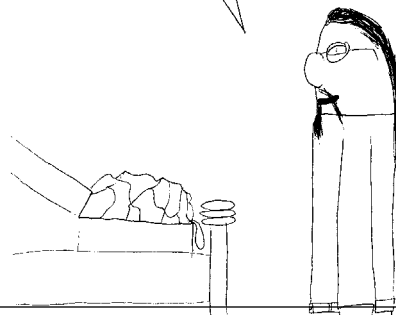
What!

To Be Continued

SGML, HTML, and XML: The History, Present, and Future of Markup Languages on the Web Part 1

Erika Orrick
erika_orrick@fitch.com

The World Wide Web has been one of the hottest trends ever to emerge in US society, technological or not. Millions of people, including quite a few high school and college students, have made money building web pages for others. But, how many of these people have stopped to think about the "language" they are using to build these pages? Do you, yourself, realize that HTML is not something someone thought up recently, but instead an application of a larger markup language (SGML) that was in its infancy in the 1960s?

A markup language is not a programming language in the traditional sense that it is compiled and run on a computer. Instead, a markup language is a set of instructions used to describe something about the content of a document. In SGML, HTML, and XML, these instructions are called tags. Most people who have written or even just looked at HTML source will recognize the <HTML>...</HTML> that (should) begin and end each document. This is an example of an HTML tag. Other examples include <H1>...</H1>, <STRONG>...</STRONG>, and <DIV>...</DIV>. What you will notice about each of these tags is that even though they surround bits of the content of the document, they do not explicitly tell how that content should be displayed. (We will cover tags such as <B> for bold and <I> for italics in the next article.) This is known as "descriptive markup." Descriptive markup tells you what each element is, but not how to display it.

- Example: Descriptive Markup v. Procedural Markup
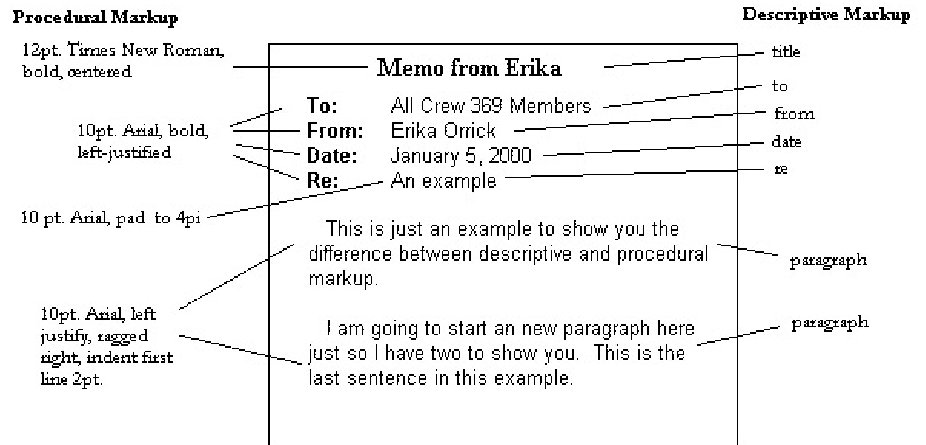
- Note: None of the markup is exact, it is just to give you an idea.

See Grapics

Contrast this with "procedural markup," which is what most word processing systems use. Procedural markup tells the program exactly how to make each element look (usually on paper). Procedural markup lets you tell the program exactly what you want your document to look like, but it comes at a price. The document must be marked up again and again as you add or change things. Once you write a document in Microsoft Word for 8.5 x 11" paper, that document is tied to either Microsoft Word or that 8.5 x 11" paper (word processors conversion utilities notwithstanding). This may not be a problem if you are an American college student where you and all your professors have easy access to Microsoft Word, but imagine if you were a international law firm that publishes its documents to both 8.5 x 11" paper and A4 paper. (A4 is a paper used elsewhere in the world that is slightly longer and narrower than what people in the U.S. are used to.)

This is where our story of SGML starts. Many years ago, an international law firm was tired of reformatting its documents every time it needed to send them to a different country with a different paper standard. They approached IBM to create a document program that would automatically reformat the data to the appropriate size for each country with losing the intent or the style of the document. IBM went to work on it and in 1969 Dr. Charles Goldfarb along with two colleagues developed Generalized Markup Language. In the years between 1969 and 1980, GML was further developed. The first working draft of SGML (Standard Generalized Markup Language) was published in 1980. In 1986, the International Organization for Standardization (ISO) published SGML as a worldwide standard known as ISO 8879:1986.

Before I close this part of our discussion, I will attempt to answer a question many of you might be asking yourself right now: why is SGML so important? SGML in and of itself is not a markup language used to markup documents. Instead it is a "meta-language," or a language that describes other languages. SGML is used to write "Document Type Definitions," or DTDs. DTDs describe in very explicit detail what tags can be used to markup a document. It also tells how they can be used. For example, imagine you are the U.S. military. You have hundreds of contracts out to companies all over the nation. Do you really want each company to pick their favorite word processor and send you the documentation in that form? Even if you made them all use the same version of WordPerfect, imagine going through and piecing all those documents together and putting them in the right order. Okay, so you managed to get them all together and formatted and printed out into hundreds of pages of books. That's great until someone needs a CD-ROM or two instead of 300 pounds of books. Now you have to reformat everything so that it can be

browsed from a CD-ROM. Or you can save yourself a lot of trouble by coming up with a DTD and having all of your contractors markup their documentation using it. Now, it doesn't even matter if you put all the documentation in order because SGML picks out the parts of the documentation you need based on ids and names you give to parts of the content. And by making a style sheet describing each format you want, you can sit back and let the computer put together your documentation in any format you choose. It saves a lot of work.

Next time: a tour of DTDs in general and the HTML DTD in specific.

Linux has matured, but not its users!
James D. Corder

Just as most technical neophytes lump all 80+ UNIXes[1] into one category, most Linux neophytes lump all Linuxes under one umbrella. First and foremost, Linux is an idealogy not an operating system. Its "original" author stated he would have never written Linux if he had known that Free BSD had already been written. Linux, as Free BSD, was an attempt to take the "open architecture" that UNIX was based around, unlike Microsoft, and make an affordable [free] OS. Both Linux and Free BSD are available free of charge off the InterNet and can be purchased in most computer stores for around $50.00US[2].

In its infantsy, UNIX was freely given away. Hence the split between BSD and SYSV. In the '80s, after the divestiture of the phone company, AT&T began to license UNIX. quickly the cost of UNIX rose. A single copy of UNIX could be upwards of $2,500.00. The hardware, worth the money for the performance, was astronomically expensive.

College students wanting the power of a workstation in their dorms had to come up with a cost savings alternative. There-

fore, both Free BSD and Linux were written for the cheaply and readily available personal computer hardware.

Sun Microsystems attempted to port their "SunOS" UNIX to the "Intel 386" platform. But the system didn't have enough power and Sun quickly took the Sun386i off the market. Though UNIX/RISC chips have been over the 600Mhz clock speeds for almost 5 years now, and last year past the 1Ghz mark, "Pentiums" recently past the 500Mhz speed. There-fore, making the personal computer once again marketable as a UNIX system.

Apple Computer's, in the '80s created a wonderful marketing strategy. Give their computers away at phenomenally discounted prices to high schools. Hence, when it was time to purchase a home computer parents would buy the ones that junior had already been using in school. Sun Microsystems adopted a similar relationships with the colleges. Sun sold its computers at more than 41% off list price to universities. Then when students graduated and began to purchases servers for corporate America it would be Sun.

Of the 80+ UNIXes only three are truly viable for the "High-End to Medium Server" market: #1 Solaris [Sun], #2 HPux [HP], and #3 AIX, [IBM]. For me, in the past few years DecUNIX [Digital] and Irix [SGI] have mad far to many bad decisions to be considered reliable in the support arena. For the low end, though I hate to admit it, a fourth is coming up strong: Linux.

Linux has began to captivate the low end ISP market. Moreover, Linux is a soupier product, to all but Solaris, for a firewall server. There are two things keeping Linux from obtaining true respect in the UNIX world.

1. Support

There are many UNIXes. But there is only one Solaris, one HPux, and one AIX. Many support it. But there is one final destination for ultimate support.

Since there are more than one Linux [Red Hat etc.] there is more than one flavor, There is more than one place to look for support. None of them are any good, so far! When the different flavors of Linux begin to be perceived as sperate identities, as sperate as Solaris and AIX they will stand a chance.

2. Its Users

Linux's strongest point is its biggest failing!

What set UNIX apart for Apple and Microsoft was its user community! UNIX users were more professional from the start. The high cost of entry into the UNIX world made one serious about the endeavor. My first personal UNIX station [1986] set me back $37,000.00US. To own a UNIX system said you were serious. Even in the early 90s if you had a "cheep" UNIX system you still had to put it together, therefore, you were instantly perceived as a professional.

The star-up costs of the Microsoft world was under $4,000.00US. Therefore, just about anyone with an interest or a hobby got into the computer market. The laws of supply and demand drove the cost of a Microsoft system administrator down. While the cost of a UNIX system administrator [SA] sky rocketed.

Now, with the entry level of a UNIX system being cheaper and more powerful than that of the personal computer, and the used UNIX systems being given away for free, just about anyone can get into the UNIX market place[3]. Unfortunately, the personal computer mentality is beginning to creep into the UNIX world as well.

Most personal computer owners buy on cost not on performance. They would rather save $100.00 and reboot daily. Moreover, one of the largest problems I have with the 369 is that they youth can not perceive a large

---

1.Linux is one of the UNIXes

2.Solaris for both RISC and Intell can be purchased off Sin's web site for under $30.00US.

3.I have no problem with competition, I still make more than 90% of the UNIX people.

scale professional data center. They truly believe that since they have a network of 4 or 5 personal computers running Linux as a firewall so that all the computers in their house can get access to the one cable modem that they understand networking... Moreover, it is hard to educate the low end SA about the true cost of downtime.

I can recall a client who had a server that lost $1,500,000.00US an hour when it went down. The client wanted to save $50,000.00 on cheaper disk arrays. It was a no-win-battle. The perceived bottom line won. He was a personal computer person:-(I have no way to calculate how many millions of dollars that $50,000.00 savings cost them.

Linux users, as Microsoft users, tend to view the world as small systems and small networks. Though I would prefer a Linux system over ANY Microsoft system, I would not currently bet mission critical systems on Linux.

My First Month
Suhasini Aswath
aswaths@hotmail.com

I suppose it is uncustomary to write an article about ones first month with the Venture Crew. However, I feel it is necessary. What I have learned of computers and myself in one month would have taken me at least a couple of years to learn. In one month I learned how to design a web page, tar it, and upload it. I am sure I would not have had the motivation to do this on my own! I have also learned a little bit of UNIX and got to use it when I logged onto Dave?s computer. In addition, I was elected as Patrol Leader of the Hawk Patrol. What I am most proud of myself for is that I got into the Gentleman and Ladies Club within a month, which for some, took several months! I am very happy that I joined the Venture Crew. I was a lit-

tle apprehensive about joining because of the lack of women on the Crew. Then I realized that no matter what I do, I would have to face this at some point or another since the computer industry does not have many women in it. And since I would be in the computer industry in the next four or five years, I thought I might as well get used to it! At first it felt like one of the neighborhood girls trying to get into the boy's only tree house even though Venturing is coed. However, once I came to the first meeting I realized that I had nothing to worry about. Overall, the Venture Crew has given me a new perspective to view everything and I am enjoying it immensely. I feel like I am finally being taught how to be a success in life.

Networking and the OSI Model
Lee Orrick   orrick@surfree.com

Many of you are familiar with IP addresses and why computers have them. Anyone who has a home network should know what a network cable is. You probably have used a computer application at some point in your life. In fact, if you are reading this article on-line, you are probably using an application that formats and presents HTML. All of these things can be related to a model proposed by the International Standards Organization (ISO). This model, known as Open Systems Interconnection (OSI) Reference Model, outlines an open standard for enabling computer systems to exchange information.

The model consists of seven layers stacked one on top of the other. The layers are numbered from one to seven and the numbering goes from bottom to top. The International Standards Organization envisioned several goals for this model. Layers would be created when a new layer of abstraction was needed. The functionality abstracted into a layer would be well defined with an eye towards defining internationally standardized protocols. The model needed to have enough layers to provide the levels of abstraction, but it could not

become so large that it was unwieldy and unusable by the computing community.

Think of a stack of seven drink coasters. All the coasters are different colors to signify the different data and functionality found at each layer. Each layer has its own name, and we start from the bottom of the stack, "layer 1".
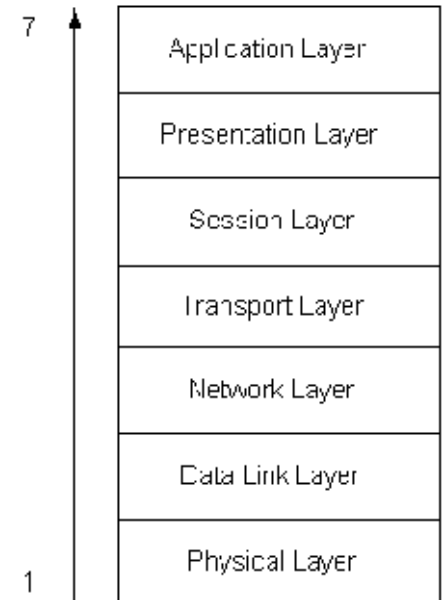
In the OSI model, communication only occurs between adjacent layers. The layers below are used to transport information from one machine to another, but they cannot and do not use information destined for another layer of the model. In other words, information from a layer is ôpushedö down the stack until it ends up at the physical layer which provides the physical means to transport the data from machine to machine. The different layers use different protocols to communicate. Layers 4 through 7 deal with end to end communication. Diagram 2 represents this with solid lines showing communication between the layers. Layers 1 through 3 also provide end to end communication, but they also negotiate communications between intervening devices. Machine A might be three router ôhopsö from Machine B and pass through 4 Ethernet switches. The physical, data link, and network layers of each of the devices participate to carry the traffic correctly from Machine A to Machine B. You notice that in place of a

solid line between the layers we see multiple boxes. These boxes represent devices such as routers, hubs and switches that the traffic might pass through on its way from Machine A to Machine B.

See Diagram 2

Let's look at each layer and some of the functions of each and the protocols used.

The physical layer is concerned with moving the bits from point A to point B. It includes the different types of cable and fiber. When you hold a Category 5 (Cat 5) cable in your hand, you are working with layer 1 of the OSI model. The Institute of Electrical and Electronics Engineers (IEEE) sets standards for physical layer medium and devices.

The data link layer, layer 2, is the first layer in the model that actually handles data. The data link layer breaks information into data frames to send information across the physical layer. Machines at layer 2 address each other at using Machine Access Control (MAC) numbers. A MAC address consists of six pairs of hexadecimal numbers. A typical representation of a MAC address has the format XX:XX:XX:XX:XX:XX. Each major hardware vendor has a distinguishing MAC prefix. For example, Sun Microsystems network interface cards

all have a MAC address with the prefix 08:00:20. So, a Sun machine on a network might have the MAC address 08:00:20:a0:04:ce.

Layer 3, the network layer, determines how packets are routed from source to destination. IP is a well-known protocol used by this layer. Machines can communicate at this layer via IP addresses. A machine uses an IP address to uniquely identify itself at a network layer. IP addresses have the form X.X.X.X where X is a number between 1 and 255.

The transport layer accepts data takes data from the session layer and splits it into smaller units and passes it on to the network layer. It also insures that information sent is received correctly at the other end. This layer uses a mechanism called flow control to regulate how fast information is sent to a remote machine so a fast machine does not overrun a slower machine with data. This is the first layer that we see true end to end.

The session layer synchronizes communications. It controls who sends data when and keeps track of whose turn it is in a conversation. It also inserts checkpoints into the data being transferred that allows a connection to resume where it left off if it is abnormally terminated.

The presentation layer deals with the syntax and semantics of the information transferred. It provides standard ways to present encoded information to the end user. Some computers use ASCII to represent strings while others use Uncoiled. The presentation layer handles the conversion of these formats into user readable form and back to machine-readable format.

The application layer contains protocols to provide software for users and handles tasks such as file transfers and file access. This is the interface that most users see.

The OSI reference model defines the function of each layer of the model. It does not, however, define the protocols used by each. Each layer has a multitude of protocols and mechanisms in place to accomplish one objective: allow machines to communicate over a network. Almost any course on computer networking will begin with the OSI model and build from there. Remember some of the concepts above, and you will be on your way to understanding computer networking.

369 prepares for 2001 (cont)
James D. Corder

The course material will be broken into three areas: Scouting/Outdoor, Technical, and Leadership. Though most youth join our program for the Technical courses [and career development], it is the leadership modules and outdoor experiences that truly make them successful in the job arena.

The Scouting aspect of our program is what developed "team" building skills. If a group doesn't work as a team on a campout, they don't eat:-)

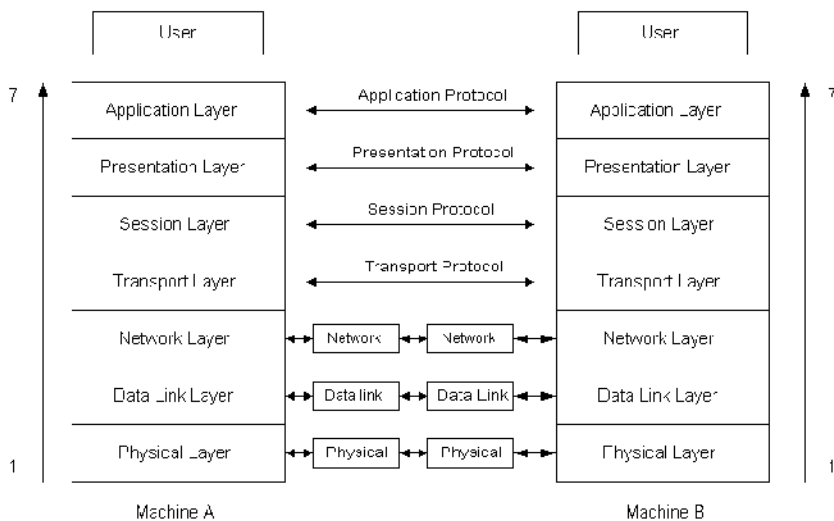The Leadership modules help our members to see more than the individual.

Diagram 2

Though employers are looking for the technical skills our youth have, they keep our youth because they can fit "their" team and need less guidance and overhead to get started.

The course materials will be rewritten next year so that a new member with "Red" level UNIX skills can take the beginning Leadership and Scouting Courses. In a way, 369 will have course offerings like a university. There will be both prerequisites in rank, attendance, and skill.

Last year 369 started the "Gentleman's Club." Though 369 has always be registered as coeducational, in 2000 the first females joined the program. Therefore, the "Gentlemen's Club" is now called "The Scouts of the Round Table", [SORT]. Membership requirements remain the same.

All members will be required to sign up for three courses: Technical, Leadership, and Scouting. All "Red" and above members are required to be an active member of SORT Those not achieving this honor will not be permitted to take the upper level courses.

Internships and career counseling will only be offered to active members of SORT. Moreover, special activities and trips will be offered to SORT members. Youth in Green Bar can loose their positions if they are not a qualified member of SORT.

By breaking the course materials into modules, 369 will be able to diversify its program. Now, the technical modules can be on anything from Veritas and Auspex to CAD, Cooking, or even diesel mechanics. Each book will be bound uniquely to the Scout and the modules s/he is taking.

Thank you God for your continual blessings!

Our Ranks, By scores on the Engineering Test.

None	White	Red	Blue	Green	Gold
-	0-79	80-179	180-279	280-299	300

Our Technical Courses

White	Venturing into UNIX For Programers	*
Red	Venturing into Shell	
	Venturing into Perl	*
	Venturing into Networking / Cisco	
Blue	Venturing into UNIX Systems Administration	
	Venturing into Oracle	
Green	Venturing into Auspex	
	Venturing into Veritas	

Part of the 2000-2001 program *

Oh C! Oh C! Where for out though Oh Schedule C?
James D. Corder

As many of you know, the last time I had a "job" was over three years ago. Last week I got a call asking me to come back "as an employee" for a company I contracted with before. I said no. They said: "Name your price", I did and they said OK.

I am a short timer. They didn't want me as a 1099:-(but a W2:-(It is only for a few weeks so I didn't care. Big mistake. I got my first pay check and I paid 48.7% in taxes. Then I owe the city I live in another 1.8% that was not with held. Therefore, I paid 50.48% in taxes. UCK.

FED Withholding	32.00%
FED FICA - MHI	1.45%
FED OASDI/DIS	6.20%
OH Withholding	7.03%
Columbus	2.00%

I agree that we need taxes for many of the services that our country provides. However, I feel ill that I have to give over half of my income so some fat welfare moms can sit on their cans sucking down bon-bons watching Opra while I work my fanny off! Some day I will tell you how I really feel:-)

Last weekend, the first "rough draft" of my new book came out. There is a chapter in it on finances. Its goal is to open the eyes of the reader about how much they are paying in taxes and to begin thinking about their retirement. It is "**NOT**" financial and/or legal advice.

One of the proof readers land based me about the tax section. It went something like: You don't pay 25% taxes for after deductions you only pay 16% and that you don't pay the entire 7.51% Social Security tax you only pay it on the first ~$5,000.00 and you don't pay sales tax on everything you buy so that my tax information is not correct.

The point of the section is to say that you need to think about putting away for retirement and long term savings. The problem with most people that argue the above point is that they are the ones that tend to be house poor and save less than 4% annually.

As I have always sated, I don't care what your tax numbers are: Simply calculate your income / retirement needs and plan for the future!

You are the master of your destiny. Only you can determine if you are going to be in a lower or higher tax bracket when you retire. I am aiming for a higher one!

■