Renumbering: Threat or Menace?

Eliot Lear, Jennifer Katinsky,
Jeff Coffin, and Diane Tharp
Silicon Graphics, Inc

Renumbering: Threat or Menace?

Eliot Lear, Jennifer Katinsky, Jeff Coffin, & Diane Tharp
– Silicon Graphics, Inc.

ABSTRACT

Remember when wide area networks consisted of private lines between a few offices routed by a good old fashion classful distance vector protocol? Ever take one of those networks and grow it by several orders of magnitude? Guess what happens? Things begin to break, and network analysts run scared. Herein, we tell the story of how Silicon Graphics, Inc. replaced our old creaking network design for 26 sites with a new bells and whistles design for 200+ sites. The astute reader can savor the irony of some of the (sometimes painful) lessons we learned along the way.

Environment

Silicon Graphics, Inc. (SGI) is a leading computer manufacturer which employs over 10,000 people world wide. It's the "world wide" part that causes us to have a wide area network (WAN) connecting some 200 sites, including our headquarters in Mountain View, CA.

In its early days (circa 1991), SGI did networks just like most IP-savvy companies. We strung a bunch of 56kbs leased lines and connected them to Cisco routers configured to use IGRP, Cisco's proprietary routing protocol. We expected to grow at a reasonable pace of 20% per year. Conventional wisdom held that we should use a Class B network subnetted for the maximum number of hosts any site would have, in SGI's case 126 hosts per network (/25 in CIDR parlance)[1].

Too bad for the network, the market decided it liked what SGI was doing. The network grew just a bit faster than planned, at an annual rate of 100% during 1992 and 1993. By the middle of 1995 we had over 150 sites connected through a Frame Relay fabric. During this growth spurt, we paid little attention to the routing architecture or addressing scheme.

Problems We Addressed

By the beginning of 1995, several problems were glaringly apparent:

- Routing overhead was quite high. A flapping ethernet interface on a router brought down the entire WAN. Even if everything was stable, there was still a fair amount of routing information transmitted to every router to indicate that nothing had changed.
- We were nearly out of network numbers. Because we had allocated networks based on 126 hosts per network, as opposed to the number of hosts they actually had (closer to 20 on average), we were using less than 10% of available address space spread across over 300 networks. At a time when Powers That Be were concerned about depleting free address space, this made it difficult for us to ask for more.
- It was nearly impossible for us to isolate Frame Relay errors. In the initial implementation, Cisco made the Frame Relay fabric appear as a multipoint network, such as an ethernet. Because the fabric was represented as a multipoint network, it was difficult to determine which sources and destinations were experiencing problems when they arose.
- We were running obsolete router code. Some routers were running code as old as 3 years, which is an eternity in wide area networking, leading to inadvertent interoperability experiments. Worse yet, one couldn't count on a particular feature being on any given router.
- We had no configuration management scheme. None of the router configuration files were kept in a central location, and people routinely tripped over each others' work.
- We did not have a reliable way to connect our wide area routers to our headquarters network. We did not even make a distinction between WAN and LAN routers. This proves to make routing quite difficult when the two are using different routing methods.

Fixing Frame Relay

SGI went to a Frame Relay network for several reasons – line cost, ability to have multiple circuits through a single local loop, and reduced lead time for new circuits at existing sites. Because we entered the frame game relatively early, we had to use the early implementation supplied by Cisco. The "multipoint" model made the WAN sites appear as if they were all on a single network. This had the advantage of

[1] network-number/X indicates that there are X number of significant bits in a route. Legal values of X are 1 through 32, excluding 31, where 32 is a host route. For example: Class A = /8, Class B = /16, Class C = /24.

allowing us to use a single network number (like 155.11.201.0/25) for all sites.

There's just one small problem. Frame Relay does not behave like an ethernet, it is not a broadcast media. This pseudo-network consists of many point to point links, usually lacking the full mesh that an ethernet provides. This limitation required us to commit a form of routing heresy, as we had to turn off split horizons, a safety feature that prevents routing information from being transmitted on the interface on which it was received. Worse yet, ethernet snooping tools could not be used to locate problems since packets aren't really broadcast.

The simplest solution would be to find a way to get back to point to point interfaces, a model that works well with IP. If you can have a virtual circuit, why not a virtual interface? With virtual interfaces, if a specific PVC[2] is down, the interface appears down, there are interface errors associated with the PVC, and the split horizons rule can be used. In other words, it looks and smells a lot more like the old serial interface model.

In order to move to virtual interfaces, however, we either had to run the virtual interfaces unnumbered on each end, or we had to assign an entire /25 network. In the long term neither solution was acceptable. Given our address shortage, and since we were attempting to move closer to a "classical" IP network model we were well advised to use the unnumbered interfaces as a transition tool to a more permanent solution. The permanent solution, in which all interfaces are numbered, would require classless IP addressing.

Moving to New Routing

With old fashioned IGRP, each WAN site would see a full routing update every 30 seconds or whenever link states changed. If a SLIP connection came up in Denver, Paris saw an update. Although this was flexible, we nearly broke our backs on overhead. Things could get really bad when an interface flapped, since every site in the world would constantly see routing updates[3].

Newer routing protocols associate each network with a network mask, allowing aggregation of a bunch of contiguous networks. For instance, outside an administrative area, one could represent the 64 network routes 169.238.0.0/24 through 169.238.63.0/24 with a single route of 169.238.0.0/18. A router outside of an administrative area would only see a small number of aggregated routes. In our example, if the Denver SLIP connection comes up, Paris no longer notices, since the SLIP network is part of a larger aggregated route [1]. Clearly we needed to move to an aggregate routing system. This change alone, however, would require every site on the WAN to renumber to new IP addresses.

Additionally, through the use of a modern routing protocol, we would be able to reduce ambient overhead by moving to incremental updates, so that changes are transmitted only when they occur, and stable information is kept off the wire.

The protocol we chose, OSPF, supports classless IP addressing, route aggregation, and incremental updates. We could have chosen Cisco's EIGRP, but we did not want to be tied to a specific vendor's protocol. At the time RIP version 2 was not yet available.

OSPF groups sites into areas. Only sites within the same area should be directly connected to each other. Only routers connected to area 0 (the backbone) may be connected to other areas. This encourages strong meshed connectivity between sites within each area (including the area 0 backbone), but discourages connections between random sites in different areas. Each area within our routing system had a hub router which connects to other hubs and to our headquarters. In addition, each site connects to our headquarters in Mountain View on a separate router from the area hub. We thus extend every OSPF area back to Mountain View. For North America, we implemented this plan using eight non-backbone areas.

Doling Out the Bits

Having decided to move to OSPF, we would have a routing protocol that supports classless IP addressing. This means that we could allocated smaller subnets for sites that had small numbers of hosts. It also meant that we could assign networks with room for only two hosts for our PVCs.

If everyone were able to understand classless IP addressing, then each subnet at each site could be perfectly tailored for the number of hosts it would contain. Of course, we don't live in a perfect world. We live in a world in which there are many different types of hardware running many different types of software, most of which do not understand classless addressing. We had two ways to address our imperfections – buy all new hardware and upgrade to all new software or use fixed subnet masks and a default route within each WAN site. The method chosen by our budget department is an exercise left to the reader.

The fixed subnet mask within the site required us to allocate each network for a site based on the largest network within that site, a far shot better than the largest subnet mask within the entire network, since most of our sites have less than 32 hosts. To determine the size of each subnet at a site we applied the following equation:

[2] Permanent Virtual Circuit. Generally, a virtual circuit that is permanently established. PVCs save bandwidth associated with circuit establishment and tear down in those situations where certain virtual circuits must exist all the time.

[3] IGRP can be configured in many ways. We diverged from its default configuration because we needed higher convergence times for transient networks, such as demand SLIP or PPP.

$$netsize = 2^{int(\log(host \times G))+1}$$

where G is a growth factor. This method allows for backward compatibility at the cost of efficiency. Since efficiency was not our prime motivation we found this to be an acceptable cost. We calculated the number of networks for each site as the maximum of the number of nets multiplied by the growth factor or the number of networks needed to fit the number of hosts.

The annual rate at which new field sites opened was leveling off from 100% to a mere 50%. We assumed that as SGI grew in popularity, the number of people required to satisfy sales demands within a region would increase. However, different regions grow at different rates. Therefore, we needed a flexible allocation plan. OSPF's aggregation between areas gave us such a plan rather naturally.

Once we decided to use classless addressing, we had to choose between one class B or a block of 256 class C addresses. With over 3000 hosts and an entirely separate routing system, our Mountain View headquarters would not use classless routing for the foreseeable future. Since each SGI workstation maintains a unaggregated routing table, the effects of 256 routes would be aesthetically unappealing at the very least, and a performance problem in our older operating systems. But use of a class B would risk incompatibilities within the WAN for those sites that could not take a default route. Fortunately, we were able to use default in every location, and thus the class B.

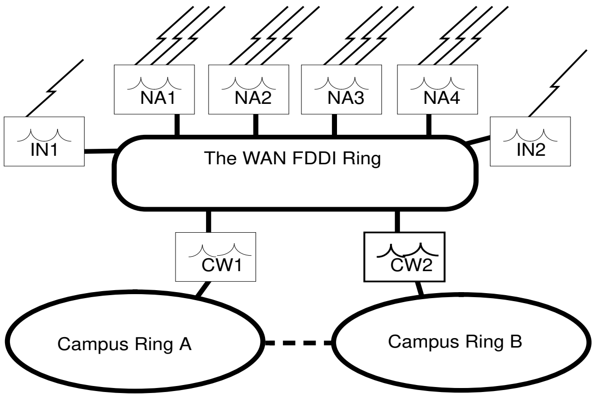

Figure 1: Getting to and from the WAN

Rationalizing Headquarters Access

When we started, we had two WAN routers that came into the campus, each had configuration files over 400 lines. Interaction between campus and the WAN was poorly defined, and interaction between the two routers themselves was not well handled. In order to keep the old routing system in one piece we ran a Frame Relay PVC between the two routers. When that failed, routing would still work, but one might end up going through Boston only to come back to Mountain View on the way to Switzerland. In order to remedy the situation we employed some technology that had matured since the last time we reviewed WAN design-FDDI. Figure 1 shows how our WAN campus interface appears.

Each router is a Cisco 4700. The ring to ring routers (CW1 and CW2) do not attach to any serial lines. The serial routers (NA1, NA2, NA3, NA4, IN1, and IN2) do not participate in any campus routing (a separate routing system). These two constraints keep the configuration files and the routing far more manageable. Each ring to ring router introduces a default route into the WAN. Thus, if one fails, the other will within a short time provide full redundancy.

Because we were moving toward an OSPF area topology we re-aligned connectivity such that all PVCs within an area came back to one of the serial routers in Mountain View, as well as a regional hub. The hub itself would also come back to a different serial router in Mountain View. This topology was designed for redundancy. In case a link directly connected to the Mountain View serial router failed, the regional hub would provide and alternate path.

Standardizing Router Software Releases

By virtue of their growing complexity, IP routing advances occur on virtually a daily basis. Cisco follows a model where they attempt to get those advances to the customer as rapidly as possible. We call this "The Release of the Week Club". There are usually two major software releases per year, along with numerous subreleases, and bug fixes to boot. Although having short release cycles can be helpful and reflects the pace of the market, different versions behave differently and have different features. With Cisco this is particularly true for OSPF and Frame Relay. When we started the project we had routers running over a dozen versions of the software ranging from version 8.2(3), released in 1990, all the way to 10.3(2), released in 1995. To top everything off we were running brand new hardware.

We wanted to pick with Cisco a release of the software that would both support our hardware platform and also meet our software stability needs.

	10.0(8)	10.2(8)	10.3(7)
Frame Relay	X	X	X
Improved OSPF		X	X
4700 support			X

Figure 2: Determining Releases

The Costs of Renumbering

It's one thing to plan up a storm. It's quite another to have it actually rain. The importance of getting management commitment in a large organization should not be underestimated. We found that the cost of renumbering our WAN was a half day of down time at each site. To mitigate the inconvenience we attempted to align downtime with other events, after hours, or on weekends.

The costs of moving to the new architecture were not limited to site downtime. Indeed classless addressing is a new paradigm, and every network support person needs to understand those changes. Earlier at SGI it was considered unacceptable for even a fixed subnet mask to appear outside of small areas of our network with any mask that did not land on an octet boundary. Our network operations staff needed to be trained to process calls where the local network mask within a particular WAN site might be set incorrectly, or not at all. We supplied everyone with a conversion chart for network masks; bit boundaries; and a table listing each site, assigned network range, netmask, and OSPF area. Even so, it takes time and training for people to understand the new paradigm.

Tools

To accomplish the task of renumbering over 7000 hosts at 83 sites, the system administrator's principle of least work was foremost on our minds. This held especially true when applied to each site administrator whom we would ask to do work. Therefore, we developed a set of tools to assist us.

Doling Out The Bits (Reprise)

Practically speaking no one in their right mind would assign 83 sites classless network address space without some sort of automation. Although usually reserved for MBAs and other budgetary masters, a spreadsheet comes in very handy. Given the number of hosts and networks at each site and a growth factor, a small number of equations can determine the address space needed for each site. If that list is sorted (a common spread sheet function) based on demand within areas, the actual address assignments can then be determined. This also requires that the areas be sorted based on demand.

The resultant spreadsheet can be used to generate tables of sites, their address assignments, and their network masks for both computer and human consumption. Put simply, without the darn thing you're probably dead in the water.

Renumbering

The script used at each site did as much as it reasonably could to automate the task of renumbering. Mondo, as we called it, took as input the old /etc/hosts file for each site, and produced as output a new hosts file that used the new assignments. In essence, Mondo is a poor man's version of DHCP [2]. Renumbering needs, such as those described in this document or others as related to Internet connectivity, present a strong argument for the use of DHCP. While later versions of SGI's operating systems support DHCP, few if any workstations within the company used it. Still, this was only part of Mondo's responsibility.

Once the hosts file was generated, the script would then push a reconfiguration script to each host within the domain. This script would run when a host was shutdown[4]. It created a new hosts file, created a new resolv.conf, modified the NVRAM configuration, set the appropriate subnet mask for the site, and installed a new SOCKS proxy configuration file that includes the new network [3][5].

Not only did we have to document everything we were doing at each site, but we needed to leave

[4] Our extremely explicit instructions did not include the warning that machines should not simply be unplugged or otherwise reset without going through the normal shutdown procedure. This nailed us in at least one instance.

[5] SOCKS is used for Internet access. SGI runs an older version that requires enumeration of every internal network. While one might think that such a file shouldn't change over time, when it does it has to be updated everywhere. The newer mechanisms for such services that do not use network numbers, such as the one used by the Netscape Navigator, are a welcome relief to this problem.

```
mondo now running on momserv.denver in denver.sgi.com.
WAN Site: Denver
New IP Range: 169.238.64.0 to 169.238.72.255
Subnet mask: 255.255.255.128
Found the following networks:
        155.11.155.0
        155.11.155.128
You have 18 new networks.
Replacing 155.11.155.0 with 169.238.64.0
Replacing 155.11.155.128 with 169.238.64.128
writing out new hosts file into /etc/hosts.new.
building package
pushing packages
cannot rcp to napili.denver.sgi.com
You will need to renumber the following hosts by hand:
         napili.denver.sgi.com
```

Figure 3: Mondo

reminders in prominent places, both for our network operations staff and for the site support staff, who in some cases might be a sales person or a graphics engineer (i.e., someone whose expertise is not system administration). We listed in each domain host file the subnet mask for each network used, the first and last valid host addresses, the reserved and broadcast addresses. In addition, we prominently placed a site's subnet mask at the top of each host file.

Mondo could only renumber SGI computers which do make up the vast majority of our network. Even so, Mondo was always billed as a system administration tool and not a replacement for the system administrator. Every site that was renumbered had some sort of small quirk, such as non-SGI hardware, where renumbering needed to take place by hand.

Within a network the size of SGI's the benefit of corporate standards quickly becomes clear. We renumbered sites ranging from two to over two hundred hosts. The time it took to renumber small sites was usually close to, and sometimes exceeded, the time it took to renumber large sites. Large sites tended to have a full time system administrator who understood the value of sticking to corporate procedures. Conversely, small sites did not see the value, and in some cases didn't even know of the existence, of those standards. When attempting to write the procedures and the scripts to be run by WAN sites, we aimed for a 99% solution. In a network of 7,000 hosts that meant we would have 70 exceptions that would need to be manually configured. We probably came in closer to 97%, handling around 200 exceptions.

DNS

Each site is responsible for generating DNS records for its computers. SGI uses an custom-written tool which takes as input the /etc/hosts file and a configuration file. The tool generates both forward and reverse DNS maps, as well as a named.boot file, with properly updated SOA records. We modified this tool so that it would output proper reverse maps for classless addresses [4].

We also wrote a program that would take as input a configuration file and generate the records necessary to delegate domains and networks in our root DNS. For instance, the octet triple 169.238.100 could have addresses 0-127 allocated to one domain, 128 through 191 to another, and 192 through 255 to yet a third.

The Instructions Generator

At different times we needed to generate specific instructions for each site. We wrote a mail merge program which took spreadsheet output as inpu, and mailed to each site administrator a set of instructions with nifty customizations such as the site router name, address, and its model. This tool helped us upgrade 83 routers in five days with minimal downtime.

The Router Configuration Checker

At some point the complexity of a routing system exceeds the understanding of enough people such that people begin to feel really nervous. This is made plain when one moves to a new routing system. NETSYS Technologies has created a product that simulates a WAN. One hands it router configuration files and it points out inconsistencies and errors within the entire routing system. This thing even knows about Cisco bugs within various releases[6].

[6] Sort of reminds one of most VT100 emulators.

```
# This hosts file was generated on Sun Mar 10 18:26:42 EST 1996 by
# mondo version 1.1.
#
#
# This is the hosts database for clubfed.sgi.com. It should
# only be used in its entirety on the relay system
# beyond.clubfed.sgi.com.
#
#
# The subnet mask for this domain is 255.255.255.128.
#
# The following networks are used in this domain:
#
#    Network      First Address   Last Address    Broadcast
# 169.238.1.0     169.238.1.1     169.238.1.126   169.238.1.127
# 169.238.1.128   169.238.1.129   169.238.1.254   169.238.1.255
# 169.238.2.0    169.238.2.1      169.238.2.126   169.238.2.127
# 169.238.2.128  169.238.2.129    169.238.2.254   169.238.2.255
```

Figure 4: An Example /etc/hosts file

Lessons Learned

We renumbered every SGI WAN site in North America in a period of six weeks. It took us six months to plan the activity, and cost us a large amount of people's time. The task would have been made easier had we previously deployed DHCP. We forgive ourselves in this case as in others, as technology such as DHCP is still emerging. The same logic holds true for the old routing and addressing schemes we used way back when. In 1991, OSPF and classless addressing were not available to us.

We also must emphasize that the reason we renumbered was that we wanted to protect our routing system from excessive storms, while still quickly converging. We did not renumber, as many sites do, today, because our Internet provider asked us to. Similarly, we could have chosen to use private address space. Because SGI communicates with many companies and tends to buy others, we chose to not use number that could possibly be used elsewhere, to reduce future complications [5].

Our choice of routing protocol is a continuing point of discussion. As mentioned earlier, we could have used EIGRP to largely accomplish the same ends. We could also have used BGP in certain places. The main reason we did not consider RIP version 2 was that it was not available on the Cisco routers at the time of this project. No matter what routing protocol, the requirement to renumber persists, and the principles used to renumber persist. We chose to address our routing problems through route aggregation, and we merely used OSPF as the means to that end.

In fact, OSPF has some limitations involving its interaction with other routing processes on Ciscos. Until the protocol is extended to handle "not so stubby areas" some routes advertised into OSPF are not aggregatable. Which means routing tables are not as manageable as we like them to be.

The importance of documentation cannot be overstated. The process we used to renumber North America was used to renumber our European network during the months of May and June. Where we documented things properly (especially the parts that caused us problems) our European counterparts were able to sail through.

The use of a test lab was critical, as were pilot implementations for each stage of our network upgrade. Before attempting to monkey with 7,000 workstations, the reader is encouraged to try a smaller number, like twelve.

Acknowledgments

This project was accomplished by a team, of which each of us were merely members. Along side the authors, the team consisted of John Parisi, Doug Herfurth, Wayne Silver, Gabriel Villalobos, Greg McGrath, Mark Mellis, Oliver Enzmann, and Chan Wilson. Many thanks to our network operations staff for their support and understanding. Numerous people at Cisco and Sprint made each transition as painless as possible. Finally, without the vast experience and boundless energy of Rod Scott this project would not have been possible. In addition, Ed Cole's and Mel Pleasant's superb project management skills as well as Rob Cowan's patience, it's not clear that we ever would have gotten the project off the ground.

Author Information

The authors are members of the Corporate Information Network Services of Silicon Graphics, Inc. and can be reached via US Mail at Silicon Graphics, Inc., 2011 N. Shoreline Blvd., MailStop 730, Mountain View, CA 94043-1389. Jennifer Katinsky can be reached via electronic mail at ariana@sgi.com. Eliot Lear can be reached via electronic mail at lear@sgi.com.

References

[1] Fuller, V., Li, T., Yu, J., and K., BARRNET Varadhan, "Supernetting: an Address Assignment and Aggregation Strategy", *RFC 1338*, BARRNET, Cisco, Merit, OARnet, June 1992.

[2] Droms, R., "Dynamic Host Configuration Protocol", *RFC 1531*, Bucknell University, October 1993.

[3] Koblas, D., Koblas, M., "SOCKS", USENIX, *UNIX Security III Symposium Proceedings*, pp. 77-83, September,1992.

[4] H. Eidnes, G. de Groot, "Classless in-addr.arpa delegation", Internet Draft, SINTEF RUNIT/RIPE NCC, January, 1996.

[5] Y. Rekhter, R. Moskowitz, D. Karrenberg, G. de Groot, E. Lear, "Address Allocation for Private Internets", 2/29/1996.