

Networking Vendor Strategy and Competition and Their Impact on Enterprise Network Design and Implementation

By

Ray Fung

B.S. Chemical Engineering, *UC-Berkeley*

B.S. Petroleum Engineering, *UC-Berkeley*

Submitted to the Sloan School of Management and the MIT Department of
Electrical Engineering and Computer Science in partial fulfillment of the
requirements for the degrees of

Master of Business Administration

and

Master of Science in Electrical Engineering and Computer Science

In conjunction with the Leaders for Manufacturing Program at the

Massachusetts Institute of Technology

June 2006

© Massachusetts Institute of Technology, 2006. All rights reserved.

Signature of
Author _____

Sloan School of Management
Electrical Engineering and Computer Science

Certified
by _____

David Simchi-Levi, Thesis Supervisor
Professor of Civil and Environmental Engineering

Certified
by _____

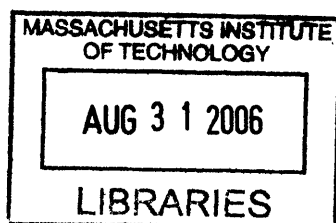
Michael Cusumano, Thesis Supervisor
Sloan Management Review Distinguished Professor of Management
Sloan School of Management

Accepted
by _____

Art Smith, Department Committee on Graduate Theses
Department of Electrical Engineering and Computer Science

Accepted
by _____

Debbie Berechman, Executive Director of Masters Program
Sloan School of Management



BARKER

Networking Vendor Strategy and Competition and Their Impact on Enterprise Network Design and Implementation

By

Ray Fung

B.S. Chemical Engineering, *UC-Berkeley*

B.S. Petroleum Engineering, *UC-Berkeley*

Submitted to the Sloan School of Management and the MIT Department of Electrical Engineering and Computer Science in partial fulfillment of the Requirements for the degrees of Master of Business Administration and Master of Science in Electrical Engineering and Computer Science In conjunction with the Leaders for Manufacturing Program at the Massachusetts Institute of Technology

Abstract

While a significant amount of literature exists that discuss platform strategies used by general IT vendors, less of it has to do with corporate networking technology vendors specifically. However, many of the same strategic principles that are used to analyze general IT vendors can also be used to analyze networking vendors. This paper extends the platform model that was developed by Michael Cusumano and Annabel Gawer to networking vendors, outlining the unique strategic aspects that the networking market possesses. The paper then reviews the strategy of the first dominant corporate datacom vendor, IBM, how it achieved its dominance, and how it lost it. The paper then discusses the strategies of various vendors who attempted to replace IBM as the dominant networking platform vendor and how they failed to do so. Finally, the paper discusses Cisco Systems, a vendor who did manage to achieve a level of dominance that parallels IBM's, and how that company has utilized its strategy to achieve and maintain its current dominance. Finally, Cisco's current strategic challenges are discussed. The impact of the strategies of the various vendors on the evolution of corporate networking is also discussed.

Thesis Supervisor: David Simchi-Levi
Title: Professor of Civil Engineering

Thesis Supervisor: Michael Cusumano
Title: Professor of Management

Abstract.....	2
1.Introduction.....	4
2. SNA – The first enterprise networking platform.....	7
2.1 – The initial SNA platform.....	7
2.2 – The rise of SNA compatible vendors.....	9
2.3 Token Ring – the LAN enhancement to the SNA platform	10
2.4 The Defeat of Token Ring by Ethernet.....	12
2.5 – SNA WAN Technology Developments.....	15
2.5.1 X.25 and QLLC.....	15
2.5.2 Frame Relay and the FRAD.....	16
2.5.3 Whither ATM?.....	17
2.6 IP and the Beginning of the End.....	18
3. IBM SNA and the Platform Levers	24
4 The Pretenders to the Throne.....	27
4.1 Xerox	27
4.2 Banyan	28
4.3 DEC	30
4.4 Apple	31
4.5 Novell.....	36
4.6 UNIX	45
5. Cisco Systems – the King of the Networking Platform Vendors	50
5.1 In the beginning.....	51
5.2 The internetworking wars.....	53
5.3 Cisco’s foray into the SNA world	56
5.4 Integration of voice and video.....	59
6. The Cisco Networking Platform and its evolution.....	62
6.1 Hardware componentization.....	63
6.2 Hardware Systems that are sold to Customers	66
6.3 The Cisco Software Platform.....	69
6.4 Cisco Business Tactics	77
7. Summary and Final Thoughts	81
8. Conclusion	88
9. Bibliography and References.....	93

1. Introduction

Enterprise data networks have evolved from being mere subordinate adjuncts to servers and clients to becoming a strategic component of an enterprise's information technology infrastructure in their own right. As networks have increased in importance, vendors that sell networking technology have likewise increased in prominence. These vendors have utilized a number of strategies to strengthen their grip on the market and to weaken the grip of their competitors. Successful networking vendors blend a number of technologies and strategies together to weave together a *networking platform* that allows for various levels of interconnection and interoperability with other vendors, but maintains a grip on various high-margin markets that can be harvested.

However, first things first. One has to be clear about what is meant by a *networking platform*. While there is no hard defining line between a networking platform and other information technology platforms, in general, a networking platform is a suite of technologies that provide primarily connectivity services between servers and clients (either dumb or smart), or that allow remotely located servers to share capabilities of some kind. A networking platform is therefore related to other information technology platforms, but also has a number of distinct characteristics. For example, unlike client-oriented platforms such as Microsoft Windows and Microsoft Office, network platforms are rarely seen or directly utilized by users. Rather, users will directly interact with clients, which in turn may invoke the network. Furthermore, unlike server-oriented platforms like the IBM mainframe or the Oracle database, the network by itself provides no computing services to clients. The network provides secure and reliable connectivity between clients and servers to allow them to pass data between each other, but interacts with the data only to the extent that doing so is required to securely and safely transmit that data. Vendors who hope to be successful in providing a networking platform must be cognizant of the similarities and differences between networking technology and other components of the information technology infrastructure, and how that impacts the firm's strategy.

Because networking platforms are related to other information technology platforms, they can be examined by the same strategic analysis that those other platforms are. In particular, networking platform vendors utilize the Four Levers of Platform Leadership strategic framework which is summarized as follows:

- Lever one relates to the scope of the firm. This lever deals with what the firm does inside and what it encourages others to do outside.
- Lever two relates to product technology, such as architecture, interfaces, and intellectual property. This lever deals with decisions that platform leaders and contenders must make with regard to the architecture of their product,

especially as regards to the degree of modularity, openness, and how much information to disclose about the platform to outside parties.

- Lever three has to do with the relationships with external complementors. This lever centers on determining how collaborative versus competitive should the relationship be between the vendor and its complementors.
- Lever four has to do with the firm's internal organization, and in particular, how the firm uses its internal organization structure to manage external and internal conflicts of interest.¹

However, because of networking platforms' unique nature, a fifth and sixth lever can be identified.

- Lever Five has to do with how the firm stokes demand for networking products in general, and for the firms' products in particular. Networks, more so than most other information technology products, display strong 'tipping characteristics' or (no pun intended) 'network effects', in that the larger the network becomes, the more valuable it is to its users. This stands in stark contrast to other information technology systems that were highly useful as stand-alone systems. For example, most computer systems built before the 1960's, such as the ENIAC and the UNIVAC, were designed to be standalone systems and were used as such. Prior to the development of the SNA networking protocol, most IBM mainframes were sold as standalone systems with no networking capability. Even during the early 1980's, most PC's were purchased without any networking capability and were to be used as standalone systems. In contrast, a network is completely useless if it has only one node, and the value of the network increases exponentially via Metcalfe's Law as more nodes are activated within the network.² Hence, a networking vendor, more so than vendors of most other IT markets, must constantly stoke customer demand for networks if the vendor hopes to succeed.
- Lever Six has to do with the competitive nature of the networking industry. Because the networking industry exhibits such strong 'network effects', the natural tendency of the industry is to tip to a single winner or perhaps an oligopoly of winners. Once a networking vendor loses share, the tipping nature of the market means that recovery is difficult. A firm that loses to a leading competitor might ultimately find itself caught in an accelerating death spiral of lost share that ejects that firm from the market completely. Hence, networking vendors have to constantly gauge how to attack and defend against other vendors. This has implications with all of the other levers, and in particular, levers 2 and 3. How do you maintain partnerships with other firms while competing against them at the same time? Do you even bother to strike partnerships at all?

¹ Cusumano, Michael and Gawer, Annabelle.. "Platform Leadership: How Intel, Microsoft, and Cisco Drive Industry Innovation". Harvard Business School Press. 2002. p. 8-9.

² Wikipedia Encyclopedia Entry on Metcalfe's Law: http://en.wikipedia.org/wiki/Metcalfe's_law

With this new strategic framework with which to analyze networking firms and networking platforms has been established, we can now apply this framework to the networking industry and examine the strategies of the various vendors.

The thesis proceeds as follows:

Chapter 2: SNA – The first enterprise networking platform discusses IBM's use of SNA and related mainframe networking technologies to create the first true networking platform and which resulted in IBM not only becoming the first true corporate datacom networking vendor, but would lead the industry for almost two decades. The various technical aspects of IBM's networking technology are explored.

Chapter 3: IBM SNA and the Platform Levers discusses the details of the IBM SNA strategy in the context of the utilization of platform levers, as well as the strategy's strengths and weaknesses. A discussion of what ultimately defeated IBM as a network vendor as well as what IBM could have done differently ends the chapter.

Chapter 4: Pretenders to the Throne discusses several other vendors that challenged for leadership in the network platform space. Such vendors include Novell, Xerox, Banyan, Apple, DEC, and the UNIX vendors. An analysis of how each vendor used or misused the platform levers follows, as well as how each vendor's attempt to claim the mantle of leadership was denied and why.

Chapter 5: The King of the Networking Platform Vendors discusses the rise of Cisco Systems to become the strongest networking platform vendor of modern times. The chapter traces Cisco's history from a humble startup to the victor of the multiprotocol wars, to the capture of IBM's SNA business and Cisco's forays into voice and video.

Chapter 6: The Cisco Networking Platform and its Evolution analyzes Cisco's use of the networking platform levers to reduce the power of its competitions while strengthening its own. Cisco's "partially open" hardware and software strategy, a business culture that relies on mergers and acquisitions as tools to be used liberally, and the leveraging of its extensive web of resellers and training/certification companies as force-multipliers are all analyzed within the context of Cisco's corporate strategy.

Chapter 7: Summary and Further Topics analyzes what new developments one might expect within the networking industry, and in particular where Cisco might be suffering from strategic weaknesses. New competitors such as Juniper and potentially Microsoft are examined.

2. SNA – The first enterprise networking platform

We first turn our attention first to the SNA (Systems Network Architecture) networking system that was launched by IBM in 1974 to provide networking capabilities to its then dominant System/370 mainframe computers, which were themselves the successors to the supremely successful System/360 mainframes that catapulted IBM into complete dominance of the industry³. Instead of offering only directly-connected terminal access to the mainframe, IBM could now provide remote terminal access to a mainframe located anywhere on the globe and therefore greatly enhanced the usability and penetration of IBM's mainframe technology. Individual departments in an enterprise that required computing capabilities no longer needed to run their own mainframes, but rather could access, through an SNA network, a centralized mainframe datacenter providing time-shared computing power. The rapid uptake of SNA cemented IBM's dominance in the enterprise information technology space for the next two decades to such an extent that to this day, the term 'legacy networks' is still taken to mean a network running SNA technology.⁴

2.1 – The initial SNA platform

While SNA, strictly speaking, is solely a networking architectural framework that defines the various ways that software protocols encapsulate and de-encapsulate data, for practical purposes, SNA is more properly treated as a complete networking platform that consists of a suite of interdependent hardware and software technologies. By leveraging its chokehold on the S/3x0 mainframe architecture, IBM dictated the implementations of SNA networks for a number of years after SNA was first launched, and many of the architectural principles that were developed during those years still hold today. The format of the actual SNA communications signals were dictated by Data Link Control, or DLC, specifications, which governed how SNA-compliant software and hardware components were to interact. SNA DLC-implemented networks in their initial flavor featured the following 4 components⁵.

- The host, which held the data that was to be remotely accessed. In all real-world SNA implementations, the host consisted of the mainframe itself, as it was the mainframe that held the data.
- The communications controller, which was originally another system external to the mainframe that managed the SNA network, and governed information exchanges between the rest of the network and the host. In IBM parlance, the communications controller was dubbed the IBM model 3745 controller and was colloquially known as the Front End Processor (the FEP). Later on, the

³ Hermes Group. "SNA: A Technological Overview". March 1, 2006.
<http://www.hermesgroup.com/whitepapers/SNA/sna.html>

⁴ Natal, Frank. Editorial comment within "Innovation Insights, A Look Back",. Oct. 18, 2004.
<http://www.informationweek.com/showArticle.jhtml?articleID=50500311&pigno=5>

⁵ Cisco Systems documentation. "Internetworking Technology: IBM Systems Network Architecture (SNA) Protocols". March 15, 2006.
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ibmsna.htm

- functionality of the communications controller was subsumed by software that was run on the host itself as an optional feature to the S/3x0 operating system.
- The establishment controller, which was another system located near the remote terminals which essentially acted as remote hubs. It aggregated the communications needs of the remote terminals and passed information requests to the communications controller. This controller was dubbed the IBM model 3174 controller.
 - The terminals themselves, otherwise known as ‘green-screens’. These were simple dumb displays that provided the user interface to display and access the host’s information. The software that controlled the display of the software was run by the host itself, with the terminal providing an exact copy of what the mainframe software wants to present. Furthermore, all keystrokes and other user interface inputs are sent directly to the host over the SNA network, with no processing performed by the terminal.

IBM supplied most devices, although competing products from 3rd-party vendors such as Data General or Honeywell were also available. Nevertheless, all of the SNA components lived within the IBM technology umbrella and were therefore tightly bound to IBM’s whim. An example of a basic IBM network, with all 4 components, is shown in figure 1.

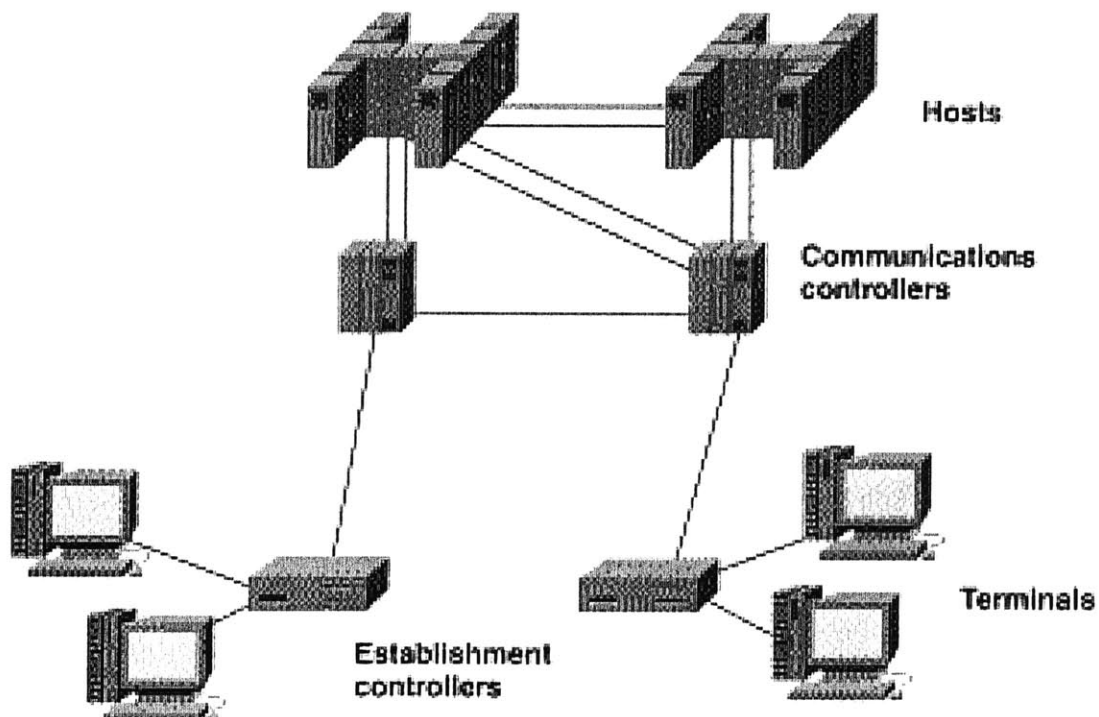


Figure 1 – a traditional SNA network⁶

⁶ Ibid.

Two of the key characteristics that an SNA network provided were *time determinism* and *high reliability*⁷. An SNA network in essence served as a complete and transparent extension of the S/3x0 mainframe computing environment itself, and therefore had to emulate certain key aspects of that environment. In particular, S/3x0 environments presume that all computing functionality be extremely reliable, with redundancies built into the hardware and software components. The environments also presume that no computing calculations or function calls will take a predictable amount of time in order to complete with only slight variability allowed. A mainframe, upon startup, will test all of its components to determine how much time it takes to complete communications with each component, and would record these times in a table held in storage. Any communications that were later found to be wildly divergent from the times of the table were deemed to be the result of a malfunction and would therefore prompt the mainframe to perform a number of software resets. Such a reset is a highly disruptive event that was to be avoided whenever possible. A mainframe subjects any SNA-connected remote nodes attached to it in the same manner – each component must be tested upon startup and end-to-end communications times with those nodes must be held relatively constant to avoid software restarts. IBM fulfilled this time determinism requirement by including time-queuing requirements that must exist in all SNA-compatible hardware and software such that no data transmission could ever be buffered within any SNA process for a length of time that was highly predictable and standardized. Reliability was provided by dictating that SNA hardware and software technologies have built-in redundancies, multiple pathways, and self-healing mechanisms. The SNA networking platform would therefore provide a hand-in-glove seamlessly reliable and time deterministic extension of the mainframe computing environment.

2.2 – The rise of SNA compatible vendors

IBM licensed its SNA technology to a number of companies who then built either SNA-compatible hardware products, or more commonly, add-on SNA software and services that would be overlaid on top of the network. Antitrust considerations meant that IBM was forced to allow for a large plug-compatible industry to develop to compete in various submarkets within the mainframe space, and similarly, IBM allowed for a large SNA-compatible industry to develop. Companies like Amdahl built 9-digit revenue businesses from selling SNA compatible gear. Third-party software vendors like Tivoli and Computer Associates were licensed to sell add-on SNA network management packages that took advantage of the various operations and management hooks available in SNA gear. Essentially, IBM followed the same playbook that it used to allow for its mainframes to be used with gear from plug-compatible vendors. IBM created a pricing umbrella under which you could buy an expensive SNA router from IBM or a cheap one from a compatible vendor, allowing for an industry of cheaper third-party vendors to flourish which would expand the size of the pie.

⁷ Boardman, Bruce. "Picky Picky: SNA over Frame Relay". Feb. 27, 1996.
<http://www.networkcomputing.com/703/703work3.html>

IBM's motives were, unsurprisingly, not altruistic. First of all, by fostering a compatible industry to develop, IBM could observe which new features and functionality customers demanded from the compatible vendors and if the opportunity proved to be promising enough, IBM would then work to include that feature in the next product release. Hence, IBM used the compatible vendors as an outsourced R&D and marketing department. Let the compatible vendors work hard to discover and develop new markets and then sweep in to capture the juiciest morsels. Such a strategy is not dissimilar to how Microsoft constantly bundles ever more utilities and features into Windows under the guise of being able to constantly redefine the operating system to be anything Microsoft wants it to be.

The other strong motivation that IBM had in allowing a compatible industry to grow was to escape the watchful attention of the various national antitrust regulators such as the US Department of Justice. IBM had already signed one consent decree in 1956⁸ with the Justice Department and was under another investigation by the US government again during the 1970's⁹. While IBM was never formally obligated to provide licenses for SNA to 3rd party vendors, IBM management clearly knew that the company ought not to engage in any activities that could be seen as monopolistic. Allowing a compatible industry to form was therefore a logical response to IBM's legal troubles.

2.3 Token Ring – the LAN enhancement to the SNA platform

IBM dutifully enhanced the networking features of SNA over the years not only to take advantage of both new technologies and customer demand, but also to provide reasons for customers to continually upgrade their technology. IBM therefore improved both the strategic power and reach of the SNA platform and, by extension, the mainframe platform, and as a consequence, generated significant incremental revenue from SNA technology.

Most important of the networking advances that IBM developed was the introduction of the Token Ring LAN in the 1970's¹⁰, which was then standardized in the IEEE 802.5 specifications in 1985¹¹. While strictly speaking Token Ring is not a SNA technology per se, the fact that it was so often used as the LAN technology of choice within SNA networks to the exclusion of practically any other LAN technology meant that Token Ring can be treated as a natural extension of the SNA networking platform. Not only did Token Ring carry the IBM imprimatur, but Token Ring had a number of technical features that made it especially appropriate for use to extend SNA networks.

⁸ Chun, David, editor. "IBM 1956 Consent Decree". US Department of Justice Document Group. Jan. 25, 1956. <http://www.cptech.org/at/ibm/ibm1956cd.html>

⁹ Demarter, Richard, editor. "IBM Antitrust Suit Records 1950-1982". 1980. <http://www.hagley.lib.de.us/1980.htm>

¹⁰ Cisco Systems documentation. "Internetworking Technology: Token Ring". March 15, 2006. http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/tokenrng.htm

¹¹ Westnet Learning Technologies and Digital Creators. "LAN Architectures". March 3, 2006. http://ih.esuhds.org/staff/aielloj/westnet/westnet/units/unit4_sec2.html

Token Ring provided a highly flexible way for terminals to connect with either 3174 establishment controllers or with local 3745 communications controller through a highly flexible and feature-laden LAN technology that could theoretically handle several hundreds of nodes and provided with a bridged network that pushed networking intelligence down into the LAN network itself. A token ring LAN was mediated through the passing of a token within a network – a token was periodically rotated through all the entities of the LAN and each entity had to wait until it had possession of the token before it could transmit data. In this way, not only are the resources of the LAN fairly apportioned amongst all the nodes, but more importantly, network communications become deterministically controlled – all nodes are guaranteed to transmit data within a certain amount of time. As mentioned above, such network determinism was crucial to the proper functioning of the mainframe computing environment. To this day, token-ring technology holds great appeal for any real-time applications, SNA or non-SNA, that relies on the strict timing of communications messages.

Token-ring also provided a number of innovative self-healing and self-monitoring technologies that vastly boosted its reliability. Each token-ring LAN will have an 'election' amongst its nodes to determine which of the nodes will be designated the Active Monitor. It is the job of the Active Monitor to ensure that tokens are always being passed properly throughout the ring, to destroy damaged tokens, and generate new tokens if no token is seen for some time. The Active Monitor does this by performing a number of monitoring tests on the token as it cycles through the ring, and by modifying fields within the token to indicate to the other nodes that the Active Monitor has examined the token. A Backup Monitor would also be elected and would activate itself if the cycling token were to no longer indicate that it had been inspected by a functioning Active Monitor. Upon such an event, the Backup Monitor would elect itself the Active Monitor and trigger all the nodes to elect another Backup Monitor. If both the Active and Backup Monitors were to die, then the other nodes of the ring would see that the token was not being inspected and would call a new election. This innovative implementation of self-healing management functionality greatly improved the reliability of Token-Ring technology and made it a natural choice for use within LAN's that required robustness.

A schematic of a token-ring network is shown in figure 2.

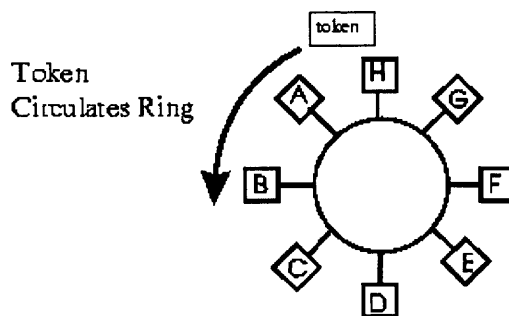


Figure 2 – A token-ring network schematic¹²

Token-ring was an extremely successful LAN networking technology for decades, as IBM leveraged the strength of both its mainframe monopoly and its dominance of the SNA architecture to propagate the popularity of Token-Ring as a LAN technology. IBM's blessing of Token-Ring, combined with its ability to handle deterministic network communications, contributed to its tremendous popularity of it from the mid 80's up until the early 90's¹³. Even to this day, Token-Ring is by far the 2nd most implemented LAN technology after wired/wireless Ethernet, and is still a common feature in existing SNA networks. As of 1998, there were an estimated 22 million Token Ring nodes and Token Ring network technologies were a \$2 billion industry¹⁴.

2.4 The Defeat of Token Ring by Ethernet

Initially, Token Ring held a number of strong advantages, from both a technical and marketing standpoint, over Ethernet. In fact, the inherent time determinism and self-healing/self-manageability features of Token Ring are unmatched by Ethernet to this day. Furthermore, Token Ring enjoyed the full marketing backing and financial muscle of IBM, which certainly dwarfed the marketing power provided by the various Ethernet vendors, many of which were essentially startups. Token-ring ultimately declined in the face of competition from the competing LAN technology of Ethernet for a number of reasons, not least of which was the proprietary control that IBM exerted over Token-ring for many years. This control served not only to jack up the pricing of Token-ring adapters, but also to slow the pace of innovation, thereby contributed to Token Ring's stagnant price-to-performance ratio relative to the fast-paced advancements and seemingly inexorable price drops exhibited by Ethernet.

¹² The Linux Review. Token Ring diagram. Jan. 5, 2006.

http://www.thelinuxreview.com/howto/intro_to_networking/stdimages/page176a.gif

¹³ PC Tech Guide. "Token Ring". Dec. 1, 2002.

http://www.pctechguide.com/29network_Token_Ring.htm

¹⁴ Rudich, Joe. "Ethernet Accelerates". Aug. 23, 2001.

<http://techupdate.zdnet.com/techupdate/stories/main/0,14179,2807260,00.html>

However, let's first deal with three purported reasons for why Ethernet supposedly defeated Token Ring and demonstrate why they are NOT the true reasons for Token Ring's decline. The first purported reason was that Token Ring was supposedly slower than Ethernet. This is only partially true, and was certainly untrue in the early years of the technologies. From the standpoint of effective bandwidth, Token Ring started off life at least on par with Ethernet. While the 4Mbps theoretical bandwidth of the first implementations of Token Ring was nominally slower than the 10Mbps theoretical bandwidth available in the first implementations of Ethernet, in reality, this advantage was a chimera. Ethernet, unlike Token Ring, utilizes a contention based method of network access in which any node that wants to transmit will do so, without regard for whether other nodes are trying to transmit at the same time. If 2 nodes were to attempt to transmit simultaneously, a network "collision" would occur which would force both nodes to retransmit. Token-ring networks, on the other hand, apportion network transmit time via the token. Only the node that possesses the token can transmit. Since only 1 token is present in a ring at any given time, no collisions are possible, and thus Token Ring dispenses with the need for nodes to retransmit because of collisions. Hence, much of the theoretical bandwidth of Ethernet is consumed by collisions such that the effective bandwidth of a 10Mbps Ethernet network is not significantly greater than that of a 4Mbps Token Ring network.

Furthermore, Token Ring bandwidth initially leapfrogged Ethernet and could most likely have kept technological pace with Ethernet in the later years. 16 Mbps Token Ring was standardized almost immediately after 4 Mbps Token Ring was launched. 16 Mbps Token Ring was obviously significantly faster than 10 Mbps Ethernet from a purely nominal basis, and was tremendously faster from an effective bandwidth standpoint. Ethernet did not benefit from a boost in bandwidth from 10 Mbps to 100 Mbps until 1995 with the standardization of the IEEE 802.3u specification, about a decade after 16 Mbps Token Ring was standardized.^{15 16} Continual and timely Token Ring bandwidth upgrades could have been made to keep pace with Ethernet but were not. That is because IBM's focus, as will be discussed later in this paper, was directed elsewhere and few other strong vendors had the incentive to push the technology forward. Even so, 100 Mbps Token Ring was launched in 1998¹⁷, and Gigabit Token Ring was launched in 2001.¹⁸ While these specifications were announced with little fanfare and with few available products, it does demonstrate that the notion that Token Ring could not have been ramped up in speed is a canard. Speeds could and did get ramped up. It is the author's opinion that they could have been ramped up at least as fast Ethernet speeds were if IBM or other vendors had invested the same R&D budgets into doing so as the Ethernet vendors were.

The second flawed reason for why Ethernet supposedly defeated Token Ring is that Ethernet scales better than Token Ring. In fact, the opposite is true. A Token Ring

¹⁵ IEEE 802.3u 1Fast Ethernet Specification

¹⁶ IEEE 802.5 Basic Token Ring specification

¹⁷ IEEE 802.5t 100 Mbps Token Ring specification

¹⁸ IEEE 802.5v 1000 Mbps Token Ring specification.

network scales significantly more gracefully than does an Ethernet network. More nodes on an Ethernet network mean more collisions and ultimately collisions began to consume the bulk of the available bandwidth. In contrast, Token Ring networks transmissions are mediated by the token, which means that an individual Token Ring network is able to scale to a greater number of nodes. Moreover, the 802.5 Token Ring specification allowed for the use of networking devices known as bridges to segment and scale a Token Ring LAN by breaking up one large ring into multiple smaller rings¹⁹. Numerous vendors, including IBM, sold bridge devices. Token ring bridges served as the functional equivalent of hybrid switches and routers in the Ethernet world, yet Ethernet routers would not be invented until the mid 1980's and Ethernet switches would not be invented until the late 1980's.

The third flawed reason for why Ethernet supposedly defeated Token Ring was the aforementioned advent of Ethernet switching. Many people believe that switched Ethernet was the killing blow to Token Ring. It is true that a switched Ethernet network provided a dedicated network path to each connected node, which increased the useful bandwidth in a LAN by orders of magnitude by eliminating the possibility of collisions and hence greatly enhancing not only the effective bandwidth of an Ethernet LAN, but also its scalability. It is certainly true that the advent of Ethernet switching was a major advance that greatly enhanced the effective bandwidth of an Ethernet network. However, it is the opinion of this author that bandwidth alone was not the most serious issue on the table, for as mentioned above, Token Ring also could and did improve its bandwidth to Gigabit speeds. Furthermore, Token Ring itself began to implement switching technology of its own kind through the advent of Dedicated Token Ring (DTR)²⁰, not for the purposes of eliminating collisions, for no such concept existed in Token Ring, but for the purposes of reducing token travel time. Just like switching an Ethernet network effectively provides each node with its own collision-free path to the network, switching a Token Ring network effectively provides each node with its own token. Switched nodes then no longer have to wait to obtain the token in order to transmit, for they always have possession of the token. Hence, switching technology multiplies the effective bandwidth of both an Ethernet and a Token Ring network and cannot by itself be a reason for why Ethernet defeated Token Ring.

The true reason why Token Ring was defeated by Ethernet therefore had little to do with any technological advantages enjoyed by Ethernet. Token Ring quickly was considered to be expensive relative to Ethernet²¹, and IBM did not develop Token Ring's features as quickly as the Ethernet vendors did.²² Nor did IBM open the Token Ring standard to competing vendors so that they could advance the technology. In essence, IBM played Token Ring too close to the vest. IBM did not try to establish Token Ring into any networking protocol environment beyond SNA, for IBM did not see Token Ring as

¹⁹ IEEE 802.5 Basic Token Ring Specification.

²⁰ IEEE 802.5r Dedicated Token Ring Specification.

²¹ Blanchard, Eugene. "Introduction to Data Communications". 1999.
http://www.techbooksforfree.com/intro_to_data_com/page176.html

²² D'Allesandro, Paul; deLeon, Jeff; Littman, Daniel; and Quane, Stephen. "Case 8: Intel's Gigabit Ethernet Strategy". March 4, 2006. <http://www-personal.umich.edu/~afuah/cases/case8.html>

anything more than a simple LAN adjunct to SNA. IBM also applied the same business principles to Token Ring as it did to its mainframe technology – high prices combined with a strong initial technical lead but slow technological development – and its Token Ring technology was surpassed in price and performance by Ethernet just like its mainframes were surpassed by the minicomputers.²³ Antitrust considerations may also have played a role for IBM may have felt loathe to dominate another computer technology for fear of giving the Department of Justice another reason to indict the company.²⁴ In any case, it is not surprising that a technology like Token Ring that was so closely identified with the IBM mainframe would suffer from the same poor losses of market share that most of IBM's mainframe technologies endured in the face of new competition. Token Ring continued to maintain a strong showing within mainframe SNA networks themselves for two decades, but was never able to break out of that niche in any significant way. However, today, a strong push by systems integrators exists to migrate remaining token-ring networks to Ethernet, particularly as fewer and fewer vendors offer support for Token Ring hardware.

2.5 – SNA WAN Technology Developments

IBM also evolved SNA to work with various telecom WAN offerings as they were made available. SNA originally was to be run over a WAN through dedicated 56kbps leased lines by running the SDLC (or Synchronous Data Link Control) protocol that provided for proper synchronization and timing effects for the DLC data transmissions. Leased lines were and still are a highly expensive option, because a leased line is by definition completely dedicated trunk capacity to your enterprise at all times, even if no data transmissions are taking place. Telecoms therefore have no opportunity to aggregate demand for leased line capacity to oversubscribe their networks the way that they do with voice capacity. It was only natural for telecoms to therefore charge an arm and a leg for leased line capacity, which made SNA WAN networks tremendously expensive. Yet for nearly a decade, SDLC transmissions over leased lines were the medium by which SNA WAN signals were sent²⁵.

2.5.1 X.25 and QLLC

In response to the growing demand for data leased lines, the telecom industry created the X.25 packet data network (PDN) standard through the auspices of the International Telecommunications Union (the UN agency devoted to the regulation of international telecommunications),. X.25 was developed to provide an extremely reliable data-specific network protocol with a global footprint that would be provided by telecoms throughout the world, with internationally mediated handoff points across national boundaries. More importantly, X.25 aided the business models of the telecoms for X.25 networks could be aggregated and oversubscribed. Telecoms could sell more data capacity than their X.25 network actually had, banking on the principles of statistical multiplexing to increase

²³ Christensen, Clayton. "The Innovator's Dilemma". Harvard Business School Press. 1997. Chapter 2.

²⁴ Hart, David M. "Antitrust and Technological Innovation". Issues in Science and Technology, Winter 1998 issue. <http://www.issues.org/issues/15.2/hart.htm>

²⁵ Protocols.com. "SNA Protocols". March 3, 2006. <http://www.protocols.com/pbook/sna.htm>

network usage. On occasion, the X.25 network usage would exceed capacity, requiring the telco to pay rebate credits according to their service level agreements, but such an event was considered to be the cost of doing business in return for having to build out less total network capacity.

IBM, to its great credit, quickly modified SNA to interoperate with X.25 by developing the QLLC protocol, which was essentially a modification of SDLC to incorporate X.25-specific data fields. SNA traffic, encapsulated within QLLC, grew to dominate the usage of X.25 networks. X.25 represented one of the great success stories in the early days of telecom, as enterprises were eager to use them to expand their SNA WAN's cheaply and telecoms were eager to sell X.25 networks in order to increase the capacity utilization of their networks. Most of the pre-Internet online services, such as CompuServe and the French Minitel system, were based on dial-up remote-access points that served as conversion points to the online service's X.25 network which ultimately terminated at the online service's mainframe systems²⁶. A typical QLLC-enabled SNA network is shown in figure 3.

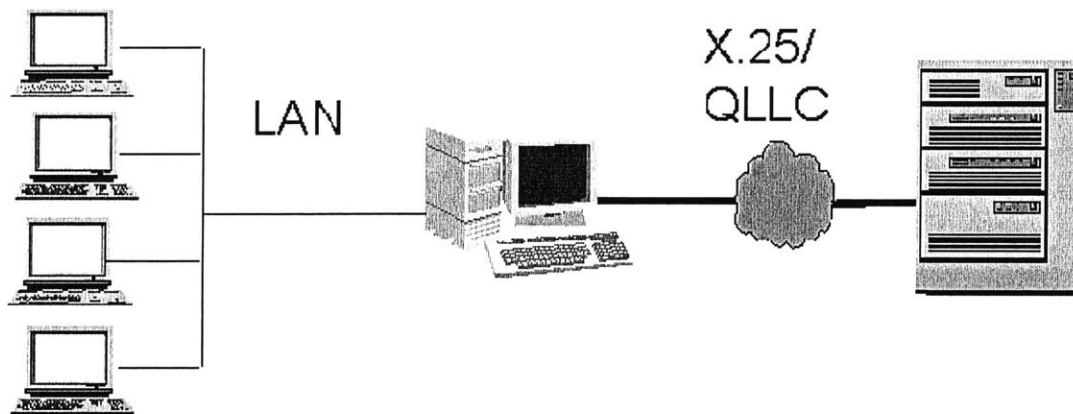


Figure 3 – An SNA network run over an X.25 WAN via QLLCs

2.5.2 Frame Relay and the FRAD

While X.25 proved to be a highly successful telecom data offering, it proved to be only the first iteration of data networking technologies that the telecom community developed. Hot on the heels of X.25 was Frame Relay, which was basically a stripped down X.25 that removed much of its heavy and computationally intensive error correction algorithms. Telecoms quickly realized that X.25 was an over-engineered protocol from an error correction standpoint – X.25 was said to be engineered to, if necessary,

²⁶ Artesyn Technologies. "Teledatcom Glossary: X25". March 4, 2006.
<http://www.artesyn.com/resources/glossary/x-25.html>

successfully run over chicken wire²⁷. As the overall reliability and quality of the international telecom network improved, it was determined that much of that error correction capability could be removed. A number of manageability and interconnection features were then included into the ITU Frame Relay specifications which were completed in 1990 and 1991²⁸. The result was a data networking technology that was more scalable and cheaper to provision than X.25.

IBM dutifully enhanced the SNA platform to access Frame Relay access networks by first developing the concept of the Frame Relay Access Device (the FRAD), which was generally a device that converted SDLC or Token Ring frames into frame-relay compliant frames. Two FRADs were generally placed on either end of a frame-relay network and hid all the elements of the frame-relay network from the SNA network. The Frame Relay network therefore became transparent to the SNA network. An illustration is shown in figure 4.

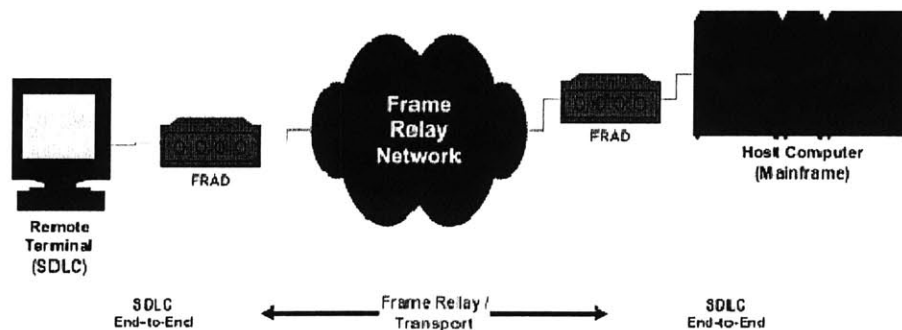


Figure 4 – the FRAD²⁹

2.5.3 Whither ATM?

A discerning student of datacom history would next expect IBM to provide a way for SNA to be made compatible to Asynchronous Transfer Mode (ATM) technology, for ATM is basically the ‘new-and-improved’ version of Frame Relay. ATM was another ITU-standardized technology that utilized the concept of *cell switching* – data transmissions would be broken into fixed-sized units called cells, and these cells would be examined by devices known as ATM switches which would direct, or *switch*, the cell to the destination that was delineated within the destination address of the cell. Because all cells were of the same size and the destination addresses of the cells were all to be found in the same location of the cell, switches would be able to make cell-switching decisions extremely quickly. Furthermore, the use of fixed-size cells, as opposed to

²⁷ Perlman, Radia. “Interconnections 2ed: Bridges, Routers, Switches and Interconnection Protocols”. Addison-Wesley. 2000. Chapter 7.2.

²⁸ Frame Relay Forum. “The Basic Guide to Frame Relay Networking”. 1998.
http://www.tti.unipa.it/~alcuri/Materiale_Telematica/FrameRelay.pdf

²⁹ Burek, Joan. “The Frame Relay Primer”. E-server magazine, Vol. 12, #2, Nov. 1996.
<http://www.tug.on.ca/articles/Volume12/V12N2/V12N2FrameRelayPrimer.html>

variable-sized cells, meant that no cell would ever be stuck waiting for a switch to buffer and process a large cell that preceded it. This meant that ATM networks would suffer from minimal delay variance (or 'jitter' in telecom parlance). The reduction of jitter was a feature that was extraordinarily important to telcos who derived most of their profit from voice calls, for the quality of voice calls is extremely sensitive to delay variance (something that plagues VoIP vendors to this day).

The ATM standard also implemented a feature-rich Quality of Service (QoS) model that allowed for telcos to divvy up an ATM network as it saw fit. Those lines such as voice trunks that needed low delay and jitter could be configured with the highest ATM QoS settings. Lines that could tolerate some delay and jitter, such as customer's Internet data lines, could be provisioned with lower-QoS ATM settings. Therefore, ATM provided a business model for the telcos by which a telco could build a single ATM backbone network and provide numerous services (voice, data, etc.) from that single network, with those services apportioned by QoS settings. ATM therefore was the first technology that allowed for network convergence.³⁰

ATM did become a tremendously popular telecom backbone network technology.³¹ However, despite tremendous efforts from the telcos, ATM never became a popular method for clients to access the network directly. Telecoms strongly pushed ATM-to-the-desktop and other initiatives to increase sales of ATM capacity, but all efforts failed.³² While ATM enjoyed great success as a telecom backbone technology, ATM never became more than a niche direct-access method. IBM therefore put little development effort into creating SNA ATM access devices.

2.6 IP and the Beginning of the End

IBM successfully extended SNA to newer LAN and WAN technologies. What IBM was not able to do, despite several half-hearted attempts, was to extend SNA into the world of IP and smart client technology. As the Internet rose in prominence which drove the convergence of the world's veritable Babel tower of networking families onto IP, SNA fell in importance as a networking platform. SNA itself also eventually converged onto IP such that most SNA networks are often times hybrid networks with IP as the core transport technology, but the terms of this convergence were to be dictated to IBM, not by IBM. The rise of IP signaled the wane of SNA that corresponded with the wane of the mainframe computing environment as the world's dominant information technology paradigm.

³⁰Wood, Robert S. "A Network Administrator's View of Multiservice Networks". Courtesy of Cisco Press. Dec. 9, 2005. <http://www.informit.com/articles/article.asp?p=426645&rl=1>

³¹Lightreading.com. "RAD aids Ethernet over ATM". Aug 29, 2005. http://www.lightreading.com/document.asp?doc_id=79683

³²Dawson, Fred. "A Role for ATM in Managing Local Traffic?". Communications Engineering & Design. April 1997. <http://testced.cahners1.com/ced/9704/9704c.htm>

One might say that IBM could have never hoped to dominate the world of IP the way it did the world of SNA, not only because IBM had no control of any IP patents but because the competition in the IP space was far fiercer than in the SNA space. Hence, IBM may have been destined to lose as the networking game changed to IP. One could say that IBM had a great run with SNA, but every technology is ultimately usurped, and such an event ultimately means doom for the primary vendor of the overthrown technology.

It is the opinion of this author that this argument is plausible, but unlikely, and is in any case unnecessarily defeatist. It is true that IBM exerted far less control over the standards of IP, all of which were documented in non-proprietary and freely available RFC's, than it did over SNA. However, IBM has been able to compete respectably in arenas in which it holds little limited intellectual property. For example, IBM built its AIX UNIX RS6000 workstation and server division into a respectably large and profitable division despite the fact that it did not control the UNIX System V patents. Indeed, IBM had to obtain a System V license in order to layer its own proprietary developments on top to create AIX³³. IBM could have similarly built proprietary networking technologies layered on top of non-proprietary IP and while SNA might have declined in importance, IBM might still have been able to become a stronger player in the IP networking space.

Furthermore, while it is true that many vendors competed vigorously within the IP space, IBM is certainly no stranger to intense competition. Whatever competition existed in the IP space during the early days of IP was certainly no more threatening than the competition from the rival "BUNCH" (Burroughs, Univac, NCR, Control Data, and Honeywell) mainframe vendors that IBM utterly defeated during the 1960's to dominate the worldwide computing industry. IBM then continued to dominate the plug-compatible vendors for decades after that. So certainly, there is little doubt that IBM had the means and the resources to defeat the IP vendors if it had chosen to do so.

More importantly, the argument may be completely irrelevant because it presumes that IP was destined to defeat SNA and take over the world. Such an event was far from inevitable. While it is true that IP was the networking technology that bound together the Internet and its precursor, the Arpanet, the fact is, the Internet for decades was a fringe technology of interest only to academics. IBM could have evangelized SNA and propagated its use throughout the world, not only in mainframe networks, but in all datacom networks. A far-sighted IBM could have provided a royalty-free license for SNA to DARPA to modify it as it pleased. A visionary IBM could have encouraged the use of SNA amongst minicomputer and PC vendors as a tried-and-true networking technology that had the full backing of the biggest computer company in the world and that also offered the benefit of seamless integration with the world's extant mainframe computer systems. While SNA and IP differ in a number of technical respects, the inner workings of SNA could have been modified to accommodate the world of smart clients and dynamic routing, as we shall see later. In fact, today, when you point your Internet browser to www.fedex.com to check the status of your package, or to

³³ Weiss, Todd. "SCO pulls IBM's AIX license in UNIX dispute". Computerworld. Jun 16, 2003. <http://www.computerworld.com/softwaretopics/os/linux/story/0,10801,82200,00.html>

www.bankofamerica.com to check your account, you are interacting with FedEx's or BoA's mainframe computer. Mainframe computers today are often times used as highly scalable web servers³⁴. Instead of having an Internet that runs on IP and provides access to a number of Internet services, many of which are provided by mainframes, we might have had an "SNA-net" that would run on SNA and that could provide the same services that the Internet provides today. Cheap client access for millions of users could be provided by royalty-free licenses to APPN technology, a technology which is discussed in greater depth later in this paper and which potentially provides a means for SNA networks to scale to millions of individual clients. The Internet would, if it even used IP at all, might then still be a low-key and unimportant government research project, with all of its functionality subsumed by this SNA-net. IBM would then be the "toll-booth" and dominant player of the SNA-net, earning money on every SNA-net transaction.

In fact, IBM actually had the network and the technology that could have been tweaked to become the SNA-net. The IBM Global Network was basically IBM's WAN service provider business, offering network services of all stripes to thousands of enterprises. So basically, IBM was already in the telco service-provider business. Furthermore, through IBM's acquisition of Rolm in 1984, IBM was, at least for a few years, the 3rd largest vendor of corporate telecom equipment in the US³⁵. It would not have been a large leap in engineering work to offer interconnectivity services to organizations who were already customers of the IBM Global Network through a tweaked SNA protocol suite. However, this was never done, and the IBM Global Network ironically began offering access to the Internet instead. IGN ultimately was sold to AT&T to become part of AT&T's ISP business³⁶, and IBM eventually sold Rolm to Siemens in 1991³⁷. However, IGN could have been the basis for the SNA-net.

Even if you believe that the concept of the SNA-net to be too visionary for IBM to execute, then that still leaves open the question of why couldn't IBM at least control the bridgehead points between the IP and SNA world? For example, while IBM chose not to participate at all in the market for Frame Relay switches, IBM successfully developed and dominated the market for FRAD's that served to convert SNA to and from a Frame Relay compatible format. Hence, even if IP was destined to defeat SNA for worldwide dominance and even if IBM could not have made a respectable showing in the IP networking world, IBM did not have to lose the market for SNA/IP converters. Yet that's exactly what happened. IBM lost that market to Cisco. Cisco's initial forays into the SNA space were simple replacements for IBM SNA components that could only do exactly what IBM's SNA gear could do and no more. IBM understandably ignored these forays just like they ignored the many other vendors that made SNA-compatible

³⁴ Woodie, Alex. "IBM Rents Linux Partitions Under Utility Sales Model. The Four Hundred: iSeries and AS/400 Insight. Vol 11, #28, July 22, 2002. <http://www.itjungle.com/tfh/tfh072202-story06.html>

³⁵ Webconsult.

The 1980's (Divestiture and a Whole New Ballgame". March 20, 2006.

<http://www.webbconsult.com/1980.html>

³⁶ E-consultancy.com. "AT&T to Acquire IBM's Global Network Business for \$5 billion" (press release). Dec. 8, 1998. <http://www.e-consultancy.com/newsfeatures/19595/at-t-to-acquire-ibm-s-global-network-business-for-5-billion.html>

³⁷ Caslon Analytics. "The Dotcom bubble". Marc 20, 2006. <http://www.caslon.com.au/boomprofile10.htm>

networking gear. However, IBM should have paid attention when Cisco developed the conversion technology of Remote Source Route Bridging (RSRB). RSRB allowed SNA data to be converted to and from IP packets to be transported over an IP core. Customers who already had an IP network would not have to obtain a separate network for its mainframes, but could now run SNA over its IP core. This was the first true example of IP convergence that would later encompass all other networking protocols and is now taking over voice and video. An RSRB network schematic is shown below in figure 5.

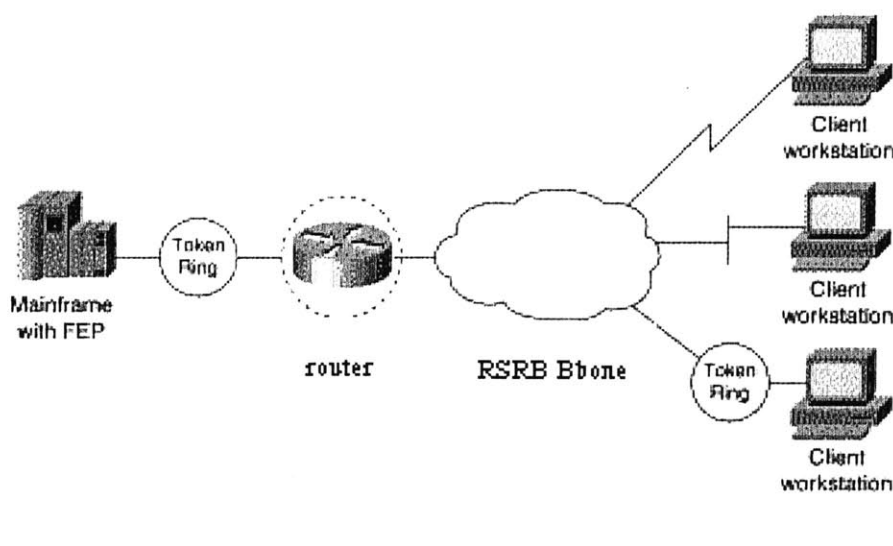


Figure 5 – RSRB network³⁸

To its credit, IBM did not ignore the threat. IBM did pay attention, and struck back with the development of Data Link Switching (DLSw), which offered the same IP-conversion feature as RSRB did, and had a number of additional performance and tuning features. Moreover, the specifications of DLSw were published as a standard in RFC's 1434, 1795, and 2166, although IBM did retain a number of patent rights. This stands in stark contrast to the technical specifications of RSRB which were never published and to this day are not known outside of Cisco. DLSw proved to be a highly successful technology, so much so that a number of networking vendors, including 3Com, Bay Networks, and Cisco itself, eventually obtained licenses to DLSw. Cisco eventually stopped pushing RSRB and promoted DLSw instead.

³⁸Cisco Systems. "Cisco IOS Bridging and IBM Networking Configuration Guide, Release 12.1" – part 2, configuring NCIA server.. March 15, 2006.
http://www.cisco.com/en/US/products/sw/iosswrel/ps1831/products_configuration_guide_chapter09186a00800d982b.html

In 1985, IBM developed the Advanced-Peer-to-Peer-Networking (APPN) suite of technologies which was an ambitious attempt to transform SNA into a flexible client-oriented networking technology that was no longer tethered to the mainframe³⁹. In a nutshell, APPN was IBM's answer to IP. Clients running APPN would still be running SNA, but would be able to locate other clients and servers completely independently of mainframe control or even of whether a mainframe existed in the network at all. APPN clients would be able to download routing and directory information as necessary in order to dynamically locate route paths and destination nodes on the fly, not dissimilar to the way that modern file-sharing technologies such as the FastTrack peer-to-peer technology that serves as the underlying locator technology of Kazaa and Grokster that are highly popular today among Internet users to share songs and movies. In fact, APPN is often times considered one of the first peer-to-peer networking technologies.⁴⁰

However, APPN, as promising as it was in theory, proved to be a case of too little too late. The Internet had already become highly popular and hence IP had become the networking protocol upon which all other data communications would eventually converge upon. Even the advent of DLSw proved to be a false dawn for IBM. Cisco nominally supported standardized DLSw but also pushed its proprietary DLSw+ technology; DLSw+ being Cisco's 'embrace and extend' of the DLSw standard to incorporate a number of Cisco-proprietary tuning features that significantly enhanced the resilience and response time relative to DLSw. More importantly, Cisco bundled DLSw+ software features into its basic IOS package which mean that customers could either buy expensive IBM-model DLSw routers to transport their SNA data over IP, or they could pay nothing extra to turn on the DLSw+ features of the Cisco routers they were already using to run their IP networks. Not surprisingly, few customers opted for the former.

Cisco also engaged in a number of other initiatives that all served to gravely weaken IBM's networking position. These initiatives will be explained fully later in this paper. Basically, Cisco developed and implemented a flurry of SNA and SNA/IP products like Serial Tunneling (STUN), the Channel Interface Processor (the CIP), SNASw (which was basically "DLSw++"), TN3270, CPCC, and numerous others, all of which served to weaken IBM's position in the SNA market. Moreover, by the mid 90's IBM's mainframe business was in visibly terminal decline, which meant that by that time IBM felt it had little skin left in the SNA game. Cisco was by then the clearly dominant router vendor with a strong revenue base in the IP world that would not be easily dislodged by IBM. Cisco had taken over most of the SNA router market anyway, having already captured 2/3 of it by 1993.⁴¹ IBM realized that even if it had invested extensive resources into its networking technology, IBM would still be unlikely to retain more than a sliver of the market.

³⁹Cisco Systems. "Cisco Interface Processors: Introducing SNA on the CMCC". March 18, 2006. http://www.cisco.com/en/US/products/hw/modules/ps2643/prod_technical_reference09186a00802381e9.html#wp813935

⁴⁰ Network Dictionary. "Peer to Peer Networking". March 18, 2006. <http://www.networkdictionary.com/networking/peer.php>

⁴¹ Cisco Systems. "IBM Internetworking". Document ID 12305. March 14, 2006. http://www.cisco.com/en/US/tech/tk331/tk336/technologies_white_paper09186a0080114a38.shtml

IBM did not help itself via its disastrous 1984 purchase and ultimate 1991 divestment of Rolm. Through Rolm, IBM obtained a strong presence in the US market for corporate telecommunication voice equipment and could have theoretically leveraged the Rolm technology to support IP and other corporate datacom networking technologies. Enterprises were already using Rolm equipment to connect branch offices together to stitch together private phone networks, so it would have not have been much of a stretch for enterprises to also use Rolm equipment to create private data networks, either separate from the ubiquitous SNA networks, or perhaps ultimately integrated with and converged with the SNA networks. However, this never happened. IBM's handling of the Rolm acquisition was poor. Much of the top talent at Rolm left, and IBM never saw Rolm as anything more than just a basic corporate telecom equipment vendor. Hence, IBM eventually sold a stunted Rolm division to Siemens, who bought Rolm largely to be able to extract revenue from servicing the installed base⁴².

In 1999, IBM made the decision to surrender to Cisco. IBM sold all of its remaining networking patents to Cisco for \$2 billion⁴³ along with a partnership to promote and sell each other's products. By this time, IBM's transition to a services-oriented company was well underway and IBM saw little value in continuing a battle with Cisco that it was destined to lose, not when there was substantial revenue to be made as a major Cisco reseller. One industry pundit put it this way:

"IBM is a computing company; they've always been a computing company," Nolle observed. "And so they thought: OK, let's do what we do best; let's do the computing side of ebusiness, and leave the networking side to somebody else...Cisco can't make money *except* on networking; IBM can't make money *on* networking. It really is as simple as that in the long run,"⁴⁴

The above quote certainly was true at the time it was written, IBM had clearly had its lunch eaten in the networking battle. Hence, the sale of the networking business was the right choice at the time. However, as stated previously, it may never have had to come to this point. It's not true that IBM can't make money on networking; in fact, it had been doing exactly that up until the 1990's. The quote is therefore true only at that particular moment in time. If IBM had developed and pushed APPN earlier, or had otherwise popularized SNA throughout the world and not just in mainframe networks, IBM might be the world's dominant networking vendor today. Even if SNA had been defeated by IP, IBM could still have become a respectable IP player or at least a respectable vendor of SNA/IP converters and at the very least surely could have sold its networking division from a stronger bargaining position to get a better price. As it happened, IBM's 1999 exit from the networking marketplace and anointing of Cisco as its successor was nothing

⁴² Dix, John. "A Timeline of Network Developments". Network World. March 26, 2001.
<http://www.networkworld.com/anniversary/line/line.html>

⁴³ Krapf, Eric. "IBM-Cisco: End of an Era for Big Blue". Business Communications Review. Oct 1, 1999.
<http://www.bcr.com/bcrmag/1999/10/p16.php>

⁴⁴ Ibid

more than a formality – IBM had already effectively handed the crown to Cisco several years previously.

3. IBM SNA and the Platform Levers

The greatest strength and greatest weakness of the IBM SNA platform strategy was its tight ties to the IBM mainframe technology. As long as the mainframes were doing well, SNA would do well. However, when mainframes began to falter, and when PC's and especially when the Internet rose to prominence, the seeds for the destruction of SNA as a platform were sowed.

The central flaw in IBM's SNA platform strategy therefore lies in Lever 1 – what IBM took to be its scope. IBM defined SNA's scope unusually narrowly. Not until the advent of APPN quite late in the game did IBM view its networking technology as being anything more than a simple adjunct of its mainframe business. SNA was never seen to be anything more than the technology by which mainframes were networked, just like Token Ring was never seen to be anything more than the way that SNA data was transmitted over LAN's. Hence, new opportunities were pursued only half-heartedly. IBM could have likely defeated Ethernet in the battle for the LAN but felt no urge to do so. SNA could have been initially engineered for cross-platform usage but was not. IBM therefore missed out on a whole slew of potential complements and markets.

Basically, in the eyes of IBM, all roads led to the mainframe, which was seen as the ultimate 'complementary good'. As long as the network supported IBM's cash cow mainframe business, the network division was doing its job. IBM's SNA strategy was initially extremely successful in accomplishing its goals as it did serve to enhance IBM's mainframe monopoly position. However, the strategy was brittle in the sense that if another computing paradigm were to rise to overthrow the mainframe, SNA was ill-prepared to deal with that eventuality.

As far as lever 4, which has to do with IBM's internal culture and structure, the analysis is similar to how IBM utilized lever 1. IBM's management was fixated on the mainframe, which is understandable given that the mainframe represented IBM's entire reason to exist. Hence, the IBM culture saw the mainframe as the superhero and networking technology as the superhero's sidekick and could never see any more of a role than that for SNA. As long as SNA extended the reach and functionality of the mainframe, that's really all that IBM cared about. Few people within IBM were ready to champion the use of SNA outside of the mainframe paradigm. Lever 4 specifically discusses how internal tensions are to be managed⁴⁵. However, in the case of IBM, there were no internal tensions. SNA was oriented to aid the mainframe, nothing more nothing less. IBM's organizational structure reflected that. The SNA and Token Ring division reported to the mainframe division. The networking engineers were subordinate to the mainframe engineers.

⁴⁵ Cusumano, Michael & Gawer, Annabel. "Platform Leadership". P. 262.

However, let's give credit where credit is due. The IBM SNA platform strategy, as it was defined within the auspices that IBM decreed, was extremely successful for several decades. IBM maintained proprietary control over SNA, but SNA licenses were sold at terms that were reasonable enough to allow for substantial competition amongst SNA licensees. IBM did foster a thriving ecosystem of compatible SNA hardware and software vendors. IBM allowed for a large plug-compatible industry to develop to compete in various submarkets within the mainframe space, and similarly IBM allowed for a large SNA-compatible industry to develop. Companies like Amdahl, Olicom, and Madge built 9-digit revenue businesses from selling SNA and Token Ring compatible gear. Numerous other vendors garnered highly respectable livings under the IBM pricing umbrella.

Allowing these vendors to survive would also provide cover in case the Department of Justice was to come calling with antitrust indictments. As long as these vendors remained weak and subservient to IBM and as long as no-one threatened the mainframe, these vendors were relatively free to do whatever they wished. IBM defined the SNA API's and interfaces in such a way to ensure that any vendor that followed the SNA standard would build gear that truly would be compatible with any other SNA gear. Even Cisco started competing within the SNA space by acquiring an SNA license⁴⁶.

Hence, IBM's role, as defined in lever 1, was to serve as the pricing umbrella and the 'vendor of choice', but also to tolerate a broad level of competition underneath that umbrella. IBM would occupy the high-end, high-margin space but leave it to lower-priced 3rd party competitors to sell to highly price sensitive customers or customers that demanded niche or highly customized networking features. Any new business that these 3rd party vendors unearthed that proved to be unusually profitable would most likely be subsumed by IBM's next product release. Tivoli, for example, always had to stay one step ahead of IBM in improving its Netview management product. That it did. Tivoli was in fact so successful in maintaining a technical edge in its management software package that IBM eventually gave up competing and simply acquired Tivoli in 1996.⁴⁷

The above describes how IBM utilized levers 2 and 3. All SNA specifications and interfaces would be available to licensees, but IBM would retain control of all of the intellectual property. The relationship between IBM and the compatible vendors would be that IBM would serve to hold the pricing umbrella high and the compatible vendors would serve to compete underneath that umbrella with the understanding that any new technology and new markets discovered would eventually accrue to IBM. However, IBM would dare not move too quickly to subsume markets for fear of reprisals from the Department of Justice, and the compatible vendors knew this.

Furthermore, IBM correctly realized that greater mainframe use would be fostered by lower telecommunications costs, and so IBM was dutiful in communicating what new

⁴⁶ Gittlen, Sandra. "IBM to Charge for QoS Patents". Network World. Nov. 9, 1998.
<http://www.networkworld.com/news/1109mpls.html>

⁴⁷ Software Magazine. "IBM gathers Moss & Co.". Software Magazine. March 1996.
http://www.findarticles.com/p/articles/mi_m0SMG/is_n3_v16/ai_18030924

technical offerings it wanted the telecoms to provide and committed to developing SNA to become compatible with those new offerings. IBM therefore once again utilized levers 2 and 3. IBM was dependent upon the telcos to improve their data services offerings to serve as complements to SNA. As the telcos developed X.25 and then Frame Relay, IBM committed to building the necessary interfaces to allow SNA to run across those new WAN services. SNA WAN's therefore became cheaper for customers to build, which increased the demand for both SAN hardware and mainframe computing capacity. The telcos were able to offer a new data service that allowed them to aggregate and oversubscribe line demand, thereby improving telco capacity utilization. Neither side (IBM or the telcos) had much interest in competing with each other - IBM had little interest in running its own public telco and most telcos had little interest in selling computers. - so the partnership between the two sides to develop new data services was highly collaborative. The one telco that could have challenged IBM - AT&T - was operating a consent decree of its own not to compete in computing⁴⁸

IBM's use of lever 5 – how network effects can be leveraged – also illustrates IBM's fatal mistake. While IBM did greatly increase the overall desirability of the SNA platform by inviting compatible vendors to offer competing SNA and Token Ring products, and IBM did work closely with telcos to provide incentives to innovate to create new WNA data services, IBM did not try to expand the networking platform to beyond the mainframe computing world. While it may have been too much to ask IBM to sell the initial incarnation of the IBM PC with full SNA and Token Ring capability, why not offer this as the IBM PC quickly progressed to become the world's dominant PC platform? Surely it was not a question of a lack of computing power on the PC – if dumb 3270 green-screen terminals could be attached to an IBM network, then surely an IBM PC could be also. If IBM had done this, then Novell might have never gotten off the ground. Why not also sell a networking stack for the other PC architectures like Apple's? Why not popularize and sell a stack to the UNIX market? Why not also to the minicomputer makers? While surely not all of these strategies would have worked, some would have and hence the overall value of the SNA platform would be enhanced. The more nodes you are able to include in your network sphere, the more valuable the network technology becomes. I would again ask the reader to imagine a world where the Internet is run not on IP but rather on SNA. If SNA can today run some of the world's most farthest-flung private networks for banks, and government organizations, then surely SNA surely had the technical scalability to run the entire Internet. {Note: whether IBM would have provided the free-flowing innovative atmosphere to create new services that the Internet provides is an entirely different question. In fact, given IBM's culture of the time, it is probably unlikely that IBM would have allowed the Internet to flourish under its tutelage. It is therefore fortunate for the world, but unfortunate for IBM, that the Internet was built on top of IP. However, whether the SNA platform could have handled the Internet from a purely technical standpoint is undeniable.}

⁴⁸ Bell System Memorial. "AT&T Divestiture". March 18, 2006.
http://www.bellsystemmemorial.com/att_divestiture.html

Finally, IBM had an unusually turgid response to the challenges to its platform. Surrendering the high ground of the LAN market to Ethernet was bad enough. The launch of Cisco's RSRB technology should have served as a red flag to indicate that IBM had a competitor that was prepared to eviscerate SNA as a networking platform, for the message behind RSRB was that all of your SNA networking components can be replaced with multiprotocol routers, and all of your DLC links can be transported over IP. The development of DLSw only served to delay the inevitable. IBM should have realized that Cisco was going to use its strong base in IP to repeatedly attack SNA. IBM could have gone on the offense by attacking the IP market to weaken Cisco's strategic position. If an offensive strategy was not tenable then IBM could have at least played better defense by matching Cisco feature-for-feature in the market for SNA/IP converter technologies like DLSw and its progeny. As it turns out, IBM played both no offense and shoddy defense. That's clearly not going to give you much chance to win, and certainly not against a relentless competitor like Cisco.

4 The Pretenders to the Throne

IBM was not the only company that created a suite of networking technologies. Numerous other vendors came up with networking stacks of varying technical merit and business success. A number of them became fairly successful in their own right and could be seen as competitors to the SNA networking platform. However for various reasons, all of them fell by the wayside.

4.1 Xerox

The legendary fecundity of Xerox PARC in producing ground-breaking technologies is probably matched only by the legendary futility of Xerox in commercializing any of those technologies. PARC researchers developed, amongst many other inventions, the Alto, which was the first true personal computer, the graphical user interface, Ethernet, which later became the most popular LAN technology in the world, and the Xerox Network Services, or XNS, protocol suite, a highly innovative networking technology that would later serve as the inspiration for a number of 'child' protocol suites.⁴⁹ Compare the Xerox triple combination of the GUI-enabled Alto running XNS over an Ethernet LAN connection with the present day typical computer user running a Windows-enabled PC running IP over an Ethernet LAN connection - the parallels are simply eerie. Xerox figured out what today's computing paradigm was going to be before anyone else did.

Sadly, what Xerox lacked was the business vision to move forward with the inventions that PARC created. It was Steve Jobs with the Macintosh and then later Bill Gates with Windows who benefited from the GUI, not Xerox. It was Apple, Commodore, Tandy, (ironically) IBM, and then later the IBM PC clones like Dell and Compaq who benefited

⁴⁹ Xerox PARC. "About PARC: History". March 25, 2006.
<http://www.parc.com/about/history/default.html>

from the PC, not Xerox. While Xerox did manage to become the 'X' in the original DIX standard that defined the interworkings of Ethernet⁵⁰, the entire Ethernet business basically walked out the door of Xerox when Bob Metcalfe left to found 3Com. Metcalfe also took XNS along with him and combined it with Ethernet to create the 3Com 3Plus product line such as the 3+Share/3+Mail email and file-sharing solutions for PC's in the days before IP became popular.

Sadly, while Xerox invented all the technical pieces of the puzzle of today's computing paradigm, Xerox never put them together to create a commercially successful networking platform of its own. Ethernet is by far the dominant LAN standard in the world, and XNS not only served as the basis for a number of other networking protocol variants⁵¹, but is still in use today in niche markets. We must view Xerox as a progenitor of a number of technologies that changed the world, but not as a company that could challenge IBM.

4.2 Banyan

Banyan is a now defunct software company that, in its heyday, produced a fairly popular line of networking software and, more importantly, overlaid services. Banyan invented the VINES system, which was comprised of client software that could run on a number of the early PC operating systems including DOS and Windows, a UNIX-derived VINES server operating system, and the XNS-derived VINES network protocol suite to interconnect everything⁵². The complete VINES system provided file and print services and the ability to scale multiple networks through the use of Streetwork, a globally consistent namespace⁵³, a precursor to modern-day directory services and a parallel to the hierarchical DNS Domain naming service used by the Internet.

⁵⁰ Computerhope.com. "Ethernet". March 25, 2006. <http://www.computerhope.com/jargon/e/ethernet.htm>

⁵¹ *Xerox System Integration Standard - Internet Transport Protocols*

⁵² Wikipedia. "Banyan VINES". March 20, 2006. http://en.wikipedia.org/wiki/Banyan_VINES

⁵³ Cisco Systems. "Internetworking Technology Handbook: Chapter 26, Banyan VINES". March 20, 2006. http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/vines.pdf

A schematic of a Vines network is shown in figure 6.

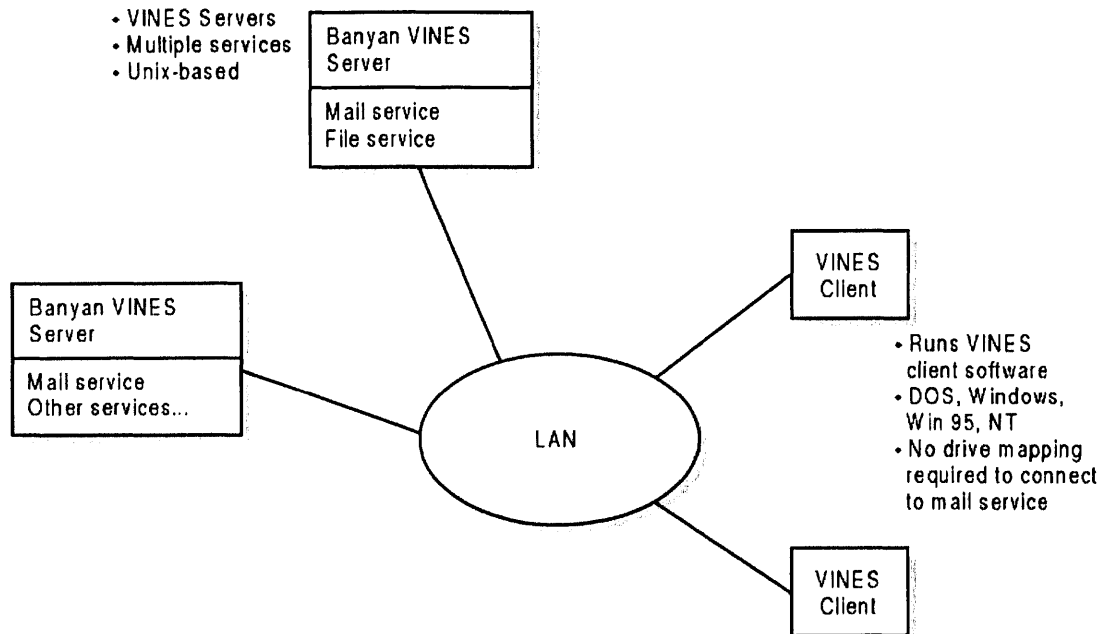


Figure 6 – A schematic of the Banyan VINES network services portfolio⁵⁴

However, Banyan could not survive intense competition for the PC networking market against other UNIX vendors like Sun that rapidly improved its network offerings to eliminate the technology lead that the VINES operating system enjoyed, and against Novell which rapidly cornered the market for PC file/print services. With pun intended, Banyan VINES withered on the vine, and Banyan limped along as a services company for a number of years, mostly providing security consulting services for legacy VINES

⁵⁴Online Books Library. "VINES diagram". March 15, 2006.
<http://www.emu.edu.tr/english/facilitieservices/computercenter/bookslib/MS%20BackOffice%20Administrator's%20Survival%20Guide%202nd%20Ed/24mba02.gif>

systems until the company was acquired by Unisys, who then proceeded to liquidate most of its assets⁵⁵.

4.3 DEC

In parallel to the development of the highly successful PDP-11 line of minicomputers, DEC developed the DECnet protocol suite to provide minicomputer networking services. DECnet proved to be highly successful and was ported to every subsequent computer architecture platform that DEC developed, including the extremely successful flagship VAX and ALPHA minicomputer systems as well as the DEC Ultrix flavor of UNIX. Coupled with DEC's status as the 'D' in the DIX Ethernet standard, as well as DEC's line of SNA-compatible hardware and software, the development of DECnet made DEC the 2nd strongest networking vendor in the world.

DEC's initial networking products allowed for simple point-to-point connectivity between DEC minicomputers. However, DEC quickly fleshed out and scaled up its network offerings, becoming a vendor of SNA compatible gear and offering a line of gateway products to allow for DEC minicomputers to interact with SNA mainframes. DEC also quickly developed its networking software to provide increasingly advanced remote file access, remote management services, multidrop network architectures, and network clustering. DECnet was also one of the first protocols to incorporate link-state routing, and DECnet later served as the basis of today's Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (IS-IS) routing protocols that are used by most of today's ISP's to manage network routing⁵⁶. DEC also developed the Local Area Transport, or LAT service which supported complete terminal emulation to a remote system, similar to what the Telnet application offers to IP-compliant servers of today⁵⁷. In short, DEC rapidly developed one of the most complete set of networking products this side of IBM.

Perhaps even more importantly, and in direct contrast to IBM, DEC actively pushed the use of DECnet onto other computing architectures. DEC integrated DECnet into Ultrix, its flavor of UNIX, and sold DECnet software implementations to other UNIX vendors. DEC also created a version of DECnet for the Apple Macintosh and for the PC. DEC saw the value of promoting an interplatform networking standard that would allow for a wide variety of systems to interconnect and share network services. DEC even offered a version of DECnet for the IBM mainframe. DECnet was therefore arguably the first networking protocol lingua franca that could be implemented by almost every single enterprise computing system in the world. While today we may think nothing of interconnecting a hodgepodge of computer systems together using IP, back in the 1980s',

⁵⁵ Unisys. "Unisys completes Acquisition of ePresence Security Services Business" – Press release. June 3, 2004. http://www.unisys.com/services/news_a_events/all_news/06038416.htm

⁵⁶ Data Network Resource. "ISO CLNP, IS-IS, and ISO IGRP". March 15, 2006. <http://www.rhyshaden.com/isis.htm>

⁵⁷ Linux-DECnet Sourceforge Project. "DECnet for Linux". March 15, 2006. <http://linux-decnet.sourceforge.net/lat.html>

the ability to use one protocol to interconnect systems from multiple vendors was nothing short of remarkable. If DEC had continued to play its cards right, DEC might have had a shot at becoming the dominant networking provider in the world.

But it was not meant to be. Just like IBM's fortunes and managerial attention were fixated on the mainframe, DEC's fortunes and managerial attention were fixated on its proprietary line of servers and workstations. As DEC's computing businesses declined in the face of withering competition, especially from Wintel workstations and servers, DEC's interest in networking waned. DEC's Ethernet division generally was successful in selling Ethernet adapters and switches only into DEC accounts and was therefore not a division that could be spun off and stand alone. While DEC did spread the use of DECnet throughout various computing architectures, DEC, like IBM, never saw its networking products as anything more than a way to sell more DEC computers. The DEC networking division had a chance to become the next Cisco, or perhaps a company even more powerful than Cisco because DEC actually had control of a proprietary and powerful DECnet networking protocol, but as long as DEC never saw the division as being anything more than an adjunct to the sale of DEC servers and workstations, it was never going to happen.

4.4 Apple

The contributions that Apple has made to the development of personal computers are legendary. What is not particularly well known is that Apple also created a respectable networking business centered on the Appletalk protocol suite that was integrated with the original Macintosh that was launched in 1984, along with a wide variety of routers and gateways to build and scale an Appletalk network. Apple developed the first true network platform for a personal computer and until only recently with the development of the TCP/IP networking suite of MacOS X, many of the more advanced networking features on the Macintosh that provided the vaunted Apple-style ease-of-use were only available if you ran your networking services via Appletalk.

Appletalk, like several of the other networking protocol suites, borrowed heavily from XNS⁵⁸. Appletalk used the concept of networks and zones. Networks were a single LAN link, and zones were a bunch of networks that were grouped together for administrative purposes. The zones were usually apportioned according to what the systems in the zone were going to do – for example, the Sales Zone comprised all of the networks that held all of the systems that a company's sales staff used. The concept of networks and zones is illustrated in figure 7.

⁵⁸ Perlman. "Interconnections". Chapter 18.

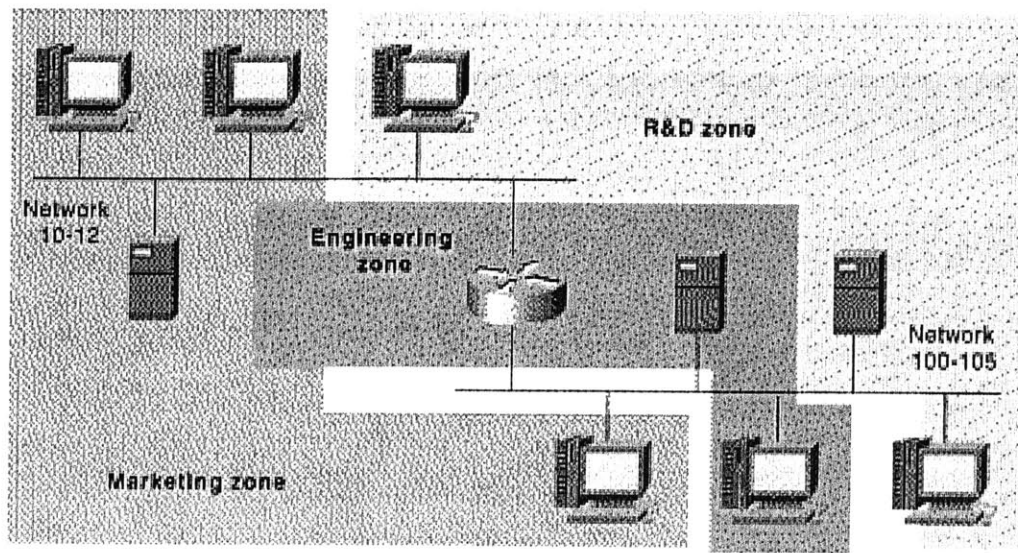


Figure 7 – Appletalk networks and zones⁵⁹

In addition, Appletalk was one of the first networking protocols that strongly emphasized ease of maintenance and ease of use. Appletalk provided an automatic addressing feature which allowed clients to provide themselves with their own network address and register themselves onto the network. An Apple client that had just booted would listen on the network for periodic network advertisement packets that contained information about what network number and zone the network link belonged to and where the routers were located. With this information, an Apple client would then have enough information to attempt to assign itself an address. It would randomly pick address numbers and send probe packets out to see whether any other system was utilizing that address, and if not, it would assign the address to itself. This reduced the need for systems administrators to preprogram Apple clients with proper address information before they could be placed on the network.

In addition, Appletalk also made use of a naming system called the Name Binding Protocol (NBP) that not only converted network address numbers into user-readable names⁶⁰, but also allowed clients to dynamically register their names into the NBP directory. Again, the idea is to improve ease of use by reducing the need for systems administrators to manually reconfigure settings every time a new client is loaded onto the network. While both the TCP/IP DNS and the Banyan Vines systems also provided a mechanism for converting address numbers into user-readable names, neither of them had

⁵⁹ Cisco Systems. "Internetworking Technology Handbook: Appletalk". March 20, 2006.
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/applet.htm#xtocid6

⁶⁰ Library noi.net. "Routing Protocols Reference Guide, Chapter 9, Appletalk". March 1, 2006.
<http://library.noi.net/networking/protocols/ro-routing-refg/ch9-apl.html>

a mechanism for clients to dynamically register names until the standardization of Dynamic DNS through the passage of RFC 2136 in 1997⁶¹ allowed IP clients to do so (Banyan never developed such a capability). Hence, Apple NBP was more than a full decade ahead of its time.

Finally, arguably the most memorable feature of the Appletalk services from a usability standpoint was the integration of the features of the Name Binding Protocol along with other features that allowed for dynamic discovery of network resources into an easy-to-use client utility known as the Chooser. Available on the Macintosh since its inception, the Chooser was by far the easiest way for a nontechnical user to find network resources. You would simply choose the type of service you want to use, such as AppleShare (file sharing), a printer (for print sharing), or some other type of network service. The client system would then poll the network to see what zones are available and provides a list of zone names, such as the Sales Zone or the Marketing Zone. Once a Zone is picked, then the user-readable names as designated by NBP are displayed of all the available devices that correspond to the type of service and zone name that were selected. The Chooser was far more intuitive to use than any system available on UNIX, and to this day, Windows struggles to offer this ease of use in choosing network resources. The Chooser is shown in figure 8.

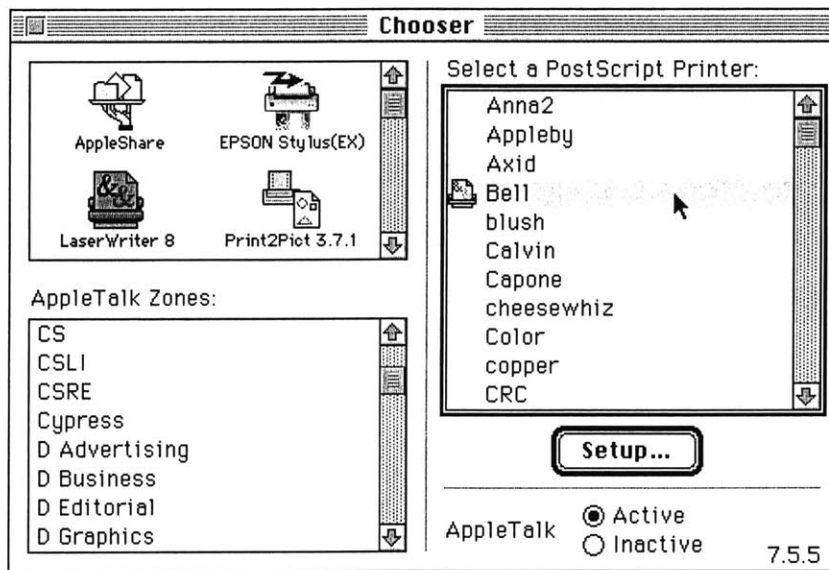


Figure 8 – Apple's Chooser⁶²

⁶¹ RFC 2136. <http://rfc.sunsite.dk/rfc/rfc2136.html>

⁶² Cheshire, Stuart. "Appletalk NBP (Name Binding Protocol). Sep. 1997. <http://www.stuartcheshire.org/rants/NBP.html>

Apple also proved fairly successful in selling a line of networking hardware. Apple sold a complete set of Appletalk routers that were especially built to be easy to install – in effect, the world’s first “plug and play” routers⁶³. Apple also launched LocalTalk in 1987⁶⁴, which was a proprietary style of LAN that could be run on a variety of transmission median, most notable being a variant known as Phonenet, which used telephone-style unshielded-twisted pair cabling that connected to a centralized hub in a topology known as the star topology. The star topology was a major advance over the single-cable bus-topology that was popular with Ethernet at the time principally because a break of the cable of a bus-topology network would cause the entire network to cease functioning, and hence all systems on the network would lose network services. In contrast, a break of one of the ‘arms’ of the star would cause only that particular section of the network to malfunction. All of the remaining systems on the network would remain functional. The idea of building a network on telephone-style unshielded twisted pair cable connected to a star topology proved to be so technically superior that it was soon copied by Ethernet in the 10BaseT standard, which, along with its faster cousins 100BaseT and 1000BaseT are the most popular Ethernet implementations in the world. The star and bus topologies are illustrated in figures 9 and 10.

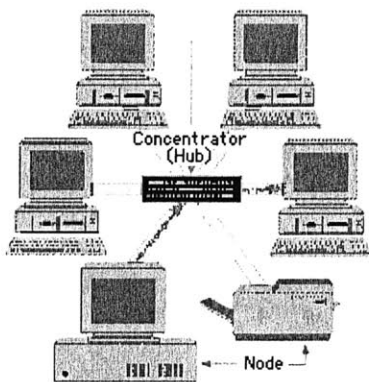


Figure 9⁶⁵

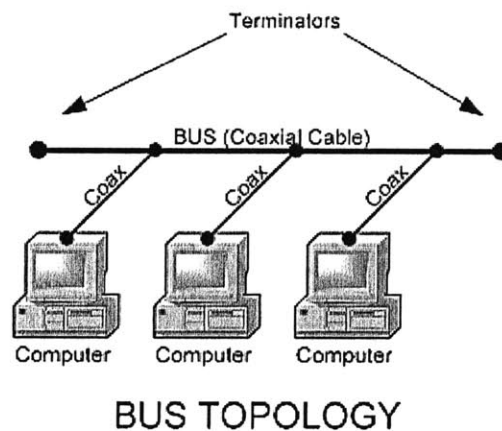


Figure 10⁶⁶

One drawback from which the Appletalk networking platform did suffer is its *chattiness*, meaning its reputation for producing a lot of network traffic, especially broadcast traffic. Broadcast traffic was especially problematic because it forced every client on a particular

⁶³ Core Competence. “Appletalk Networking: A Lower Layers Primer”. March 1, 2005.

<http://www.corecom.com/html/appletalk.html>

⁶⁴ Oppenheimer, Allan. “A History of Mac networking”. Jan 7,, 2006.

<http://macslash.org/article.pl?sid=04/01/07/1555239&mode=thread>

⁶⁵ MundoPC. “Star topology”. March 15, 2006. <http://www.mundopc.net/cursos/redes/images/star.gif>

⁶⁶ InetDaemon. “Bus Topology”. Feb. 2, 2006. http://www.inetdaemon.com/img/topology_bus.png

subnetwork to process the traffic, and therefore constant broadcast traffic would mean that client processing power would be constantly wasted. This chattiness was inherent in the way that Appletalk provided its ease of use. The fact that Appletalk clients could self-address themselves was precisely because networking advertisements were constantly being broadcast. Zone data was constantly being propagated throughout the network to ensure that all systems had a consistent view of all of the available zones.

Thus you will often times hear religious wars where Appletalk detractors will accuse Appletalk of wasting network resources because of its chattiness,⁶⁷ and that was supposedly the reason for Appletalk's decline. However, Appletalk's chattiness was not as serious of a problem as it is often made out to be and became less and less of a problem going forward. While it is true that Appletalk network management and maintenance data did consume more network capacity than it needed to, this needs to be placed in the context of the vast amounts of network capacity that exist. Consider an Appletalk network running over a LocalTalk network. The slowest LocalTalk LAN configuration provided 230 kbps of bandwidth⁶⁸. Even if an entire 2000 bit Appletalk maintenance packet were to be transmitted once every second, a tremendously high rate, this would still consume less than 1% of the entire capacity of the LocalTalk LAN, a miniscule amount by any measure. Furthermore, running Appletalk over a standard 4Mbps Token Ring or 10Mbps Ethernet LAN would render maintenance packets to consuming an infinitesimal amount of the total LAN network capacity. Now, it is true that chattiness was a more serious problem over relatively slower WAN links. However, this problem could be solved by simply tuning WAN routers to throttle the number of maintenance packets that they send over the WAN from once per second to perhaps once per minute or so. Furthermore, while broadcasted maintenance packets do force the client to process every packet on the LAN, the truth is, Moore's Law dictates that this processing would become less and less important over time. Today's microprocessors have massive processing power, the vast majority of which goes unused. Having the processor 'waste' computing power to process an extra packet per second is not a serious issue when a modern microprocessor can process many millions of instructions per second with ease.

No, what really killed Appletalk as a networking platform was the same thing that killed the IBM SNA platform. Apple saw the Appletalk protocol accompanied by the LocalTalk LAN technology as merely a way to connect Mac's together, and not as a product in its own right. Just like Apple failed to license the MacOS widely to other PC vendors and instead forced its superior operating system software to be forever bound to its (at that time) inferior hardware⁶⁹, Apple failed to popularize the Appletalk networking protocol or the LocalTalk LAN technology via licensing it on favorable terms to other computer vendors, particularly other PC vendors, and most notably to the IBM PC clones. Not only that but Apple never seriously competed to try to create a line of strong Appletalk

⁶⁷ Brisbin, Shelly. "Appletalk Chattiness: Fact or Fiction". Mac Publishing. 1998. <http://mac-guild.org/appletalk.html>

⁶⁸ US Patent 5237566. "Network Hub for Maintaining node bandwidth in a Single Node Network. <http://www.freepatentsonline.com/5237566.html>

⁶⁹ Ferguson, Charles. "High Stakes, No Prisoners". Times Books. 1999. p. 24-27.

networking gear – the Appletalk routers and switches that Apple did create were rudimentary and Apple basically abandoned the market to the dedicated networking vendors like Cisco. Once Cisco and the other router vendors had implemented a full Appletalk protocol stack within their routers, there was little need for anybody to purchase a router from Apple. While it may have made perfect business sense to abandon a market that was seen as peripheral to the success of Apple, it ultimately contributed to tossing away all the R&D effort that Apple put into building networking technology in the first place. Why even bother creating a brand new networking protocol if you're just going to hand the market for transporting that protocol off to the router vendors? Hence, the combination of a Mac-centric mentality and the disdain in developing network gear to exploit its newfound market ultimately contributed to the decline of Appletalk. A sad development this was because to this day, the Appletalk suite of protocols still held great advantages in ease-of-use and ease-of-manageability.

Granted, Apple faced a number of challenges, particularly from the intense competition from the IBM PC and PC-compatible vendors, and therefore had a lot on its plate. Asking Apple to sustain both development of AppleTalk as a technology and with a networking division devoted to selling routers and switches may have been unrealistic given all of Apple's other battles it had to fight. However, a spin-off may then have been in order. Apple could have created a separate company that contained both its AppleTalk intellectual property and its network division and IPO'd that company or sold it to private investors, with the agreement that Apple would always be able to obtain a royalty-free license to any AppleTalk improvements that this company developed. One might say that it would be dangerous for Apple to trust a spun-off company to advance AppleTalk, yet the fact is, Apple has always historically relied on other vendors for critical technology components that make the Macintosh useful, such as Motorola and IBM (and now Intel) for its microprocessors and Microsoft for MSOffice. So having to rely on the spin-off would be any worse than those situations, and in fact would almost certainly be better if Apple gets a royalty-free license to any and all AppleTalk technology upgrades. In any case, such an outcome would have monetized Apple's investments in networking technology. So, if nothing else, Apple could have at least gotten some cash back, which is far better than getting nothing at all, which is what Apple essentially ended up getting. AppleTalk is now a thoroughly marginalized technology.

4.5 Novell

We next turn our attention to Novell. Of the 5 pretenders to the throne that are profiled here, Novell probably had the most chance to become a great networking vendor. In fact, Novell was a force to be reckoned with throughout the 1980's and early 90's, at one point capturing over 70% market share in server operating systems that served PC networks, and essentially defined PC networking for an entire decade⁷⁰. Novell was extremely highly respected for producing technically sound products and cultivated a large

⁷⁰ Heskett, Ben. "A Fresh Vision for Novell?". CNET. March 1, 2002. <http://news.com.com/2100-1001-848816.html>

ecosystem of fanatically loyal dealers and systems integrators.⁷¹ Microsoft may have been the monopolist when it came to PC client OS's, but Novell was the monopolist when it came to PC server OS's.

Novell did many things right. Like DEC, Novell also developed IPX, a proprietary networking protocol suite that was feature-rich and technically sound. IPX was derived from XNS and was built with high levels of reliability and scalability in mind. Novell baked into IPX the concept of the Service Advertisement Protocol (SAP) which provided the means for servers to announce what sorts of services they were offering to clients and how they were to be reached. Hence, servers could be switched on and off and the discovery of the servers would be performed dynamically. If you had a server running IPX that you wanted to add to the network, you did not have to implement a long procedure of 'prepping' the network for the new server as you had to do within an SNA network. You just booted the server, and through listening to SAP's, the clients would learn of the presence of the new network server.

Novell also developed the Novell Link State Protocol (NLSP), a link-state routing protocol that enhanced the scalability of a large IPX network by providing for advanced routing features. Most network protocol suites, including DECnet, Appletalk, and Vines, included a basic distance-vector routing protocol where learned routes were passed from one router to another. A distance-vector routing protocol would dictate that each router would communicate information about the routes that it knows about to its adjacent routers every X number of seconds. The routers would then each calculate a routing table that dictates where that router should shunt packets that are destined to a certain place. While such a routing protocol worked well in small networks, as networks became larger, a network that was using a distance-vector protocol would run into severe scalability problems due to the fact that in a large network, it may take several minutes before information about one section of the network will have propagated to routers in another section of the network, and by that time, the transmitted routes of the network may no longer be accurate. In other words, a large network may not be able to *converge* because the routers do not have a consistent view of the network. The mechanism of a distance vector routing protocol is illustrated in figure 11.

⁷¹ Grygus, Andrew. "Novell Netware: The King Returns from the Dead". March 2, 2006. <http://www.aaxnet.com/design/novell.html>

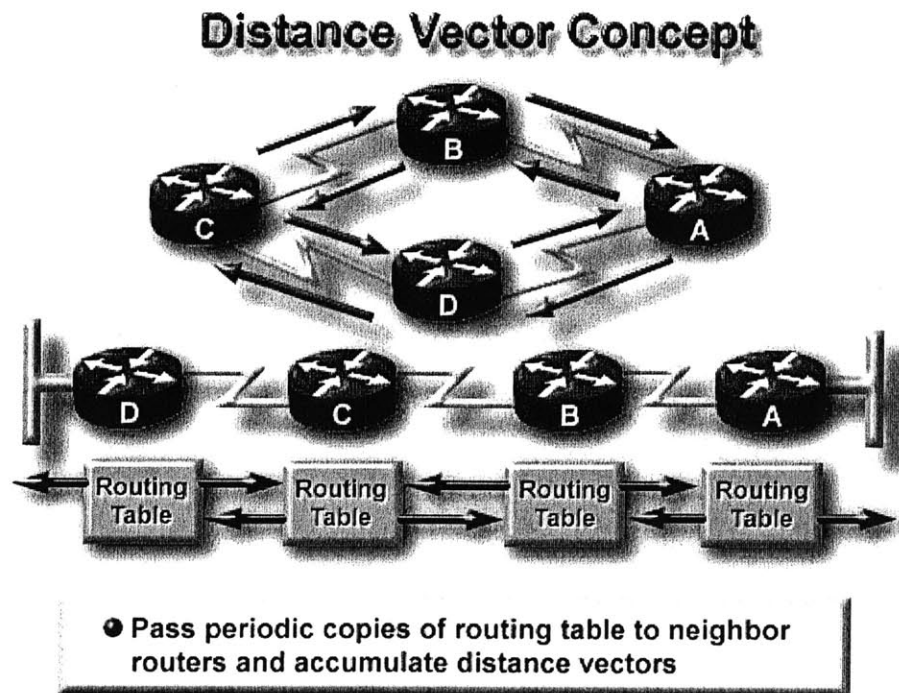


Figure 11 – The mechanism of a Distance-Vector Routing Protocol⁷²

The answer to the scalability problem lie in what are known as link-state routing protocols. While such routing protocols differ in their details, the general idea is the same. All routers flood to all other routers on the network information about all of the routes that are local to it. With each router now having information from every other router about the routes that are local to it, all the routers can determine which routers are adjacent to it, and which routers are adjacent to those routers, and so forth. Hence, each router can then calculate what the entire network looks like and where it stands in that network. The router can then calculate, using an optimization algorithm such as the Dijkstra algorithm (named after legendary computer scientist Edsger Dijkstra), each router would calculate the shortest path to every single route on the network. Whenever any change occurs on the network (i.e. if a new link was to come up), information about that link would be flooded to every router on the network and every router would then recalculate its shortest-path first algorithm. Periodic “Hello” messages would be passed between adjacent routers so that routers would check to see if an adjacent router had inadvertently shut off or if a link was now malfunctioning. While a link state routing protocol requires greater router processing power in order to run the shortest-path algorithm, the protocol ensured that every router had a consistent view of the network.

⁷² Bacon, Chet. “Distance Vector Concept”. March 3, 2006.
http://www.chetbacon.com/lessonplans/sem3/images/5_d.gif

Novell developed NLSP, one of the first link-state routing protocols in the world. The Internet Community had also developed its own link-state protocol, Open Shortest Path First (OSPF)⁷³, but it was victimized by constant political wrangling with the ITU-developed OSI-based link-state protocol dubbed Intermediate Systems to Intermediate Systems (ISIS)⁷⁴, a conflict that would not be fully resolved for years⁷⁵. "In fact, a number of members in the OSPF working group feared that IS-IS would be preferred for "political" reasons, as part of a grand plan to convert the Internet to OSI correctness, even if that meant a complete disruption of the service."⁷⁶ For a few years Novell therefore had the only link-state routing protocol that was free of political wrangling and could legitimately say that it possessed the most scalable and intrigue-free PC networking platform in the world. {Note, DEC had also implemented link-state routing into DECnet via the so-called DECnet phase V version, but DEC was never a serious player in the PC business⁷⁷}.

In addition, the IPX suite contained a method for clients to determine their own network address. However, the method that IPX used was even simpler than the method used by Appletalk. IPX borrowed from the fact that every Token Ring or Ethernet LAN adapter had what is called a Media Access Control, or MAC address, that is burned in through ROM to every single adapter and that is unique for every single adapter in the world. Hence, since the adapters for the 2 most popular LAN technologies in the world had unique addresses already, it was a simple matter of merely appending that unique MAC address to whatever IPX network number a particular LAN network the local IPX router assigned to that LAN, to come up with a fully network address (network number + local address) for that client. For example, if the IPX network number assigned to a certain Ethernet LAN was '1', and the MAC address of an Ethernet adapter for a client was 1234.5678.1234, then the complete address of that client would be 1.1234.5678.1234. This is illustrated in figure 12.

⁷³ RFC 1131. <http://www.faqs.org/rfcs/rfc1131.html>

⁷⁴ RFC 1195. <http://www.faqs.org/rfcs/rfc1195.html>

⁷⁵ Perlman. p. 367-368.

⁷⁶ Huitema, Christian. "Routing and the Internet, 2ed.". Prentice Hall. 2000. p. 324-327.

⁷⁷ Christensen. Chapter 2.

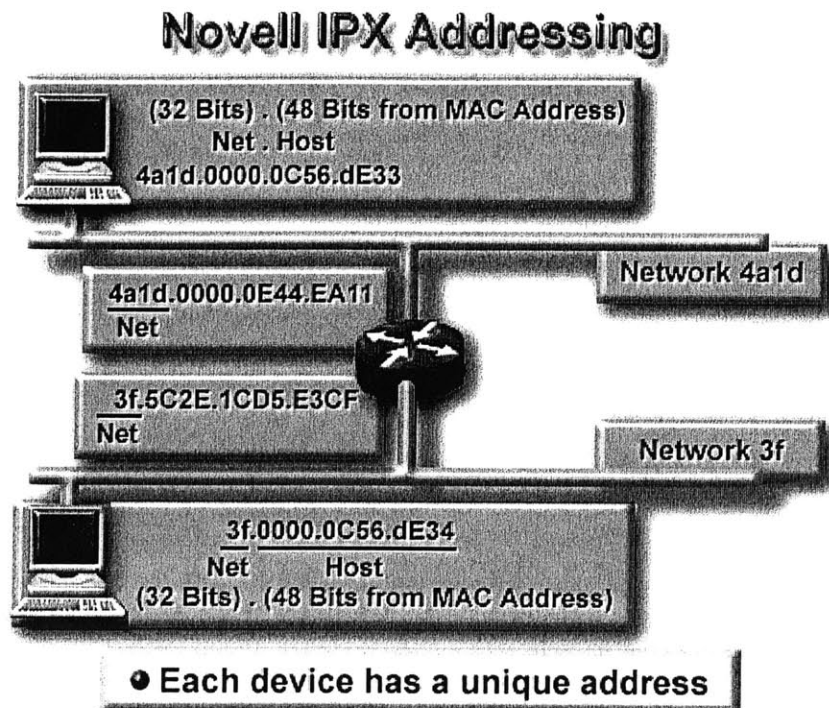


Figure 12 – How IPX clients automatically obtain unique addresses for themselves⁷⁸

In addition to a tremendously intricate and well-designed network protocol suite, Novell had Netware, one of the most technically elegant server operating systems in history, and certainly the most highly regarded server operating systems built for a PC during its time. Netware was noted for its extreme stability and reliability, often running for years without requiring a reboot⁷⁹, a level of reliability that Microsoft Windows even until recently struggled to match⁸⁰, and arguably still hasn't matched to this day. Moreover, not only was the Netware core operating system extremely reliable, it also included a number of features such as the System Fault Tolerance and implementation of raid mirroring, and transaction locking that greatly enhanced the reliability of the data it was storing. Other features such as the integration of Novell Directory Services, made Novell the premier way to store a wide variety of data that would integrate with various directory systems available in other networking platforms such as Vines (through Streettalk) and the various open directory systems such as UNIX's LDAP and NIS. Novell hence boasted of a wide portfolio of well integrated networking applications and

⁷⁸ Bacon, Chet. "IPX Networking". March 2, 2006.

http://www.chetbacon.com/lessonplans/sem3/images/7_c.gif

⁷⁹ Ibid.

⁸⁰ Musthaler, Linda. "Netware: It Just Works:". Network World. July 15, 2002.
<http://www.networkworld.com/columnists/2002/0715musthaler.html>

protocols that strongly complemented each other and formed the basis for the development of a large ecosystem of service providers.

Netware also obviated the need for explicit IPX routers by building the routing functionality within Netware itself. In fact, running Netware servers concurrently as routers was probably actually preferable to using routers run by a third party for the simple reason that combining a server with a router made the server more robust against network failures. Routers, by definition, need to be multihomed – meaning that it has multiple network links - as there is no point for a router to have only a single network link (for if there is no alternative path for traffic to emerge, then there is point in routing the data as there is only one path by which the traffic can be sent anyway.) However, a server that is multihomed is robust against a failure of one of the networks to which it is attached. It can simply offer its services across the other link. Hence, combining server with routing improves the reliability of the server, and is yet another reason why Netware developed such a strong reputation for dependability.

Novell was also a key figure in popularizing Ethernet and contributing to the downfall of Token Ring. While Netware and IPX were fully compatible with Token Ring, Novell saw that high prices of Token Ring adapters were going to inhibit the uptake of PC LAN's. So Novell became one of the largest resellers of Ethernet adapters in the world, selling them for at-cost in order to put convenient and cheap LAN technology in the hands of customers. One pundit wrote:

3com had been doing a good business selling Ethernet cards for \$180 each when Novell waded into the market selling cards for a third of that price. [Novel CEO Ray] Noorda called it "growing the market" 3Com founder Bob Metcalfe called it "attempted murder". The result was an explosion of Ethernet sales all going to customers for Novell's Netware operating system, which quickly became the de facto standard.... Growing the market was brilliant – if counterintuitive – in an era when companies prided themselves on high profit margins. Novell made high margins, just not on its network cards, which quickly became a standard. What Noorda did was turn on its head King Gillette's notion of effectively giving away safety razors in order to make big profits on razor blades. Noorda's adapter cards were the razor blades, given away to encourage the sale of razors (network operating system licenses)..."⁸¹

⁸¹ Cringely, Robert X. "Breaking the Bank". I, Cringely. May 9, 2002.
<http://www.pbs.org/cringely/pulpit/pulpit20020509.html>

The crowning networking technical achievement of Novell may have been the Netware SFT-III clustering system which provided the first commercial method to cluster PC's together in what is known as a shared-nothing architecture. Two clustered Netware servers would apportion themselves in a master/slave (or primary/secondary) relationship. The slave server would mirror all of the applications and settings of the master server. A heartbeat monitor link would be used by the slave server to constantly check on the status of the master server. If the master server was to go down, the slave server would then notify the network that it was now taking over for the master server and thus all requests for services on the master service ought to be directed to the slave instead. If and when the master server is restored to health, the master and slave resync each other's data and settings and the master server reclaims its mantle. The mechanisms of this clustering feature is shown in figure 13

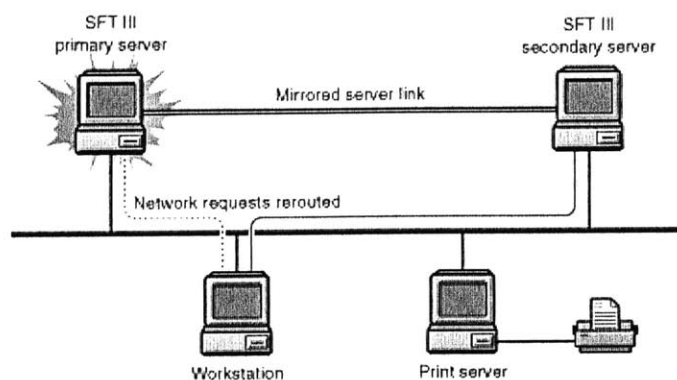


Figure 13 – Novell clustering⁸²

Novell also widely disseminated its client software to a wide variety of platforms. Novell offered clients for every version of Microsoft DOS and Windows, as well as versions for the Macintosh and for the most popular versions of UNIX and Linux. In short, Novell correctly saw that it could not be dependent on the PC but rather had to offer an abstracted network operating environment that was accessible from all of the most popular client systems. Novell's calling card would be that you could change your client systems to whatever you wanted and be assured that your Novell services would remain reachable. In other words, Novell hoped to commoditize the client system and thereby make the network the strategic lever point of the corporate IT system.

Novell also courted a wide range of systems integrators and consultants who touted the use of Netware amongst their corporate clients. In part, this was due to Netware's extremely feature-rich design which resulted in a bevy of bewildering choices and accompanying sheer complexity in installation which meant that tremendous revenue opportunities existed in simply configuring Netware systems for enterprises and then

⁸² Grant, Todd. "Will it be Lobster or Linguini?". Novell Cool Solutions Feature. Apr. 25, 2001. <http://www.novell.com/cool solutions/feature/90.html>

making maintaining and overseeing changes to those systems throughout the years. Basically, it was easy money. A consulting firm can hire employees and put them through a series of training classes to learn how to implement Netware, and then those employees can be contracted out at high profit margins to companies who wanted to install a Netware system. Netware encouraged this by creating a wide-ranging training sequence known as the Certified Novell Engineer sequence of training certifications and made millions of dollars in revenue from offering training classes through Novell Training Services⁸³. Novell-trained consultants had a strong incentive to recommend that clients install Novell, and the more clients that install Novell, the more people would want to become Novell-trained consultants. This was a virtuous cycle of training and systems integration that Novell would not be the last vendor to exploit. Cisco would later leverage this concept to devastating effect with the Cisco Certified Internetworking Expert program as will be discussed later in this paper.

Finally, Netware enjoyed significant performance advantages over its competitors. Netware was not a general purpose OS, but was built from the ground up to handle server requests. In particular, Netware operated on the file-service request paradigm, where data was requested as files as opposed to a sequence of individual blocks. Single requests for files and opposed to multiple requests for the multiple blocks that comprise a file vastly increased the efficiency of Netware. Furthermore, the IPX protocol suite included the Netware Core Protocol which dispensed with the long series of handshaking and acknowledgements that most remote file-sharing network protocols required. The result was a network operating system that “reduced the overhead of extensive process switching and scheduling”⁸⁴ and therefore offered vastly improved performance over its competitors. Novell boasted that a single-processor Netware system could outperform a quad-core NT system⁸⁵. Even if one might think that Novell would naturally state that Netware performs better than competing platforms, certainly many information technology professionals at least believed that Netware was a strong performer⁸⁶.

Given all of these advantages, technical and otherwise, that Novell’s products enjoyed, and given their apparent chokehold they enjoyed in PC corporate networking in the early 1990’s, one can ask the well-founded question: what happened? While some would say that the rising tide of the TCP/IP based Internet was bound to swamp Novell, the fact is Novell IPX networks were offering far more network services than the Internet was during the 1990’s. It would not have been a stretch for Novell to imagine a worldwide network based on IPX that could have connected the various IPX corporate islands together. Novell could have subsidized the initial development of a global IPX network backbone and thus created the worldwide ‘IPX-net’, just like IBM could have created a global network backbone to create a worldwide ‘SNA-net’. As mentioned previously, IPX is in certain ways actually technically superior to TCP/IP, and combined with the

⁸³ Novell Inc. “Novell Reports Third Fiscal Quarter 2002 Results”. Novell Press release. Aug 22, 2002. <http://www.trainingpressreleases.com/newsstory.asp?NewsID=392>

⁸⁴ Lee, Ron & Major, Drew. “Netware on One CPU Outperforms Windows NT Server on Four CPU’s”. Novell Appnotes. Jun 1, 1996. <http://support.novell.com/techcenter/articles/ana19960601.html>

⁸⁵ Ibid.

⁸⁶ Wikipedia. “Netware”. March 3, 2006. <http://en.wikipedia.org/wiki/Netware>

rich set of network services that Novell offered, was arguably a more compelling network suite than TCP/IP was.

However, even if we accept that TCP/IP were to swamp IPX, then that still leaves the myriad other components that comprise the Netware networking platform. Novell still boasted of an extremely reliable network operating system that could provide a wide variety of reliable server functions. Novell still had Novell Directory Services, which was later rebadged as eDirectory, which as of 2001 still held a near 90% share.⁸⁷ Novell still offered a wide range of software clients that spanned the range of the most popular client operating systems. Novell did successfully transition all of its network services to TCP/IP with the release of Netware 5 in 1998.⁸⁸ So while the decline of IPX was a blow, it was far from fatal.

The real problems were twofold. First of all, Novell developed a reputation for poor judgment both on a managerial and a technical level. Netware became known as a server operating system that was unusually difficult to write applications on top, something that even Novell acknowledge when it stated that "...it has traditionally been difficult to write to these [Netware network] services"⁸⁹. Novell was quite slow to aid the development of a thriving ISV market that could sell applications on top of Netware. Furthermore, Novell management decided to embark on a series of ill-advised acquisitions, purchasing Dr-DOS, the Unix Systems Labs, QuattroPro, and most infamously, WordPerfect⁹⁰. All of these acquisitions proved to be costly failures,⁹¹ with the WordPerfect fiasco alone costing more than \$700 million when it was finally sold to Corel.⁹²

The other and far more dire problem for Novell that contributed to its decline was the relentlessly harsh and arguably illegal competition from Microsoft. It is believed by some that Microsoft deliberately broke the compatibility of Windows with the Novell client⁹³, just like what Microsoft did to DR-DOS in the early 90's⁹⁴⁹⁵. Microsoft crushed the Novell Wordperfect gambit in a possibly illegal fashion by bundling Microsoft Word with its Windows monopoly, although since the case was settled for over half a billion

⁸⁷ Novell Inc. "eDirectory Positioning". Novell Cool Solutions. Jan 18, 2001.
<http://www.novell.com/coololutions/feature/5318.html>

⁸⁸ Chappell, Laura. "Migrating to Pure IP with Netware 5". Netware Connection. Sept 1998.
http://support.novell.com/techcenter/articles/nc1998_09c.html

⁸⁹ Novell Inc. "Netware SDK for Java Platform". Novell Appnotes. Jun 1, 1996.
<http://support.novell.com/techcenter/articles/dnd19960607.html>

⁹⁰ Morettini, Phil. "The Rise and Fall of Novell". PJM Consulting blog. Nov. 16, 2005.
<http://www.pjmconsult.com/2005/11/rise-and-fall-of-novell.html>

⁹¹ Murphy, Paul. "Goodbye SuSE, SuSE Goodbye". LinuxInsider. Nov. 13, 2003.
<http://www.linuxinsider.com/story/32137.html>

⁹² Anthes, Gary. "Changing Fortunes". Computerworld. Nov. 12, 2001.
<http://www.computerworld.com/networkingtopics/networking/story/0,10801,65534,00.html>

⁹³ Ibid.

⁹⁴ Goodlin, Dan. "Microsoft Emails Focus on Dr. DOS Threats". CNET. April 28, 1999.
http://news.com.com/Microsoft+emails+focus+on+DR-DOS+threat/2100-1001_3-225129.html

⁹⁵ Ferguson, p. 317.

dollars⁹⁶, it will never be known for certain whether what Microsoft did to WordPerfect was legal or not. However, certainly the fact that Microsoft paid that sum of money indicates that its actions were probably not on the up-and-up.

However, the biggest factor of all was that Microsoft simply made Windows NT Server operating system better and better, narrowing the performance gap between Netware and tying NT Server with the NT Workstation client OS. Microsoft, unlike Novell, enjoyed excellent relationships with the developer community and made it a point to provide tools and technical material to ease the creation of applications to work on top of Windows NT, even including emulation software called WoW (Windows on Windows) to allow Windows NT to run all of the applications that Windows 9x could run. Furthermore, Microsoft developed the same close relationships with systems integrators and consultants through judicious use of the highly successful Microsoft Certified Systems Engineer (MCSE) training program which served as the direct parallel to the CNE program. In essence, Microsoft leveraged its monopoly in the client OS to muscle itself into the server OS space, and used every weapon at its disposal to do so.

Novell lives on as a shadow of its former self, having merged with Suse Linux and offering the Linux-based Open Enterprise Server in parallel with Netware. Few industry pundits believe that Netware will last for long and even Novell itself has stopped promoting it⁹⁷. NDS/eDirectory lives on in greatly straitened circumstances as a somewhat popular directory server but is facing stiff competition from Microsoft's Active Directory and Sun Microsystems's Java Enterprise System suite. Novell still offers interesting network service offerings such as the Internet-integrated iPrint and iFile services, but the future outlook for Novell is dicey. Certainly, Novell's network platform strength is nothing like what it was during the halcyon days of Netware and IPX.

4.6 UNIX

Finally, we turn our attention to the various flavors of UNIX. Considering the eventual and total victory of TCP/IP over every other networking protocol suite out there and of the long history that UNIX has had in transporting and providing IP network services - particularly after the development of Bill Joy's BSD UNIX - you might think that UNIX would emerge as **the** major player in the networking space. "The network is the computer" goes a famous saying by the marketing department of Sun Microsystems; so famous that Sun has actually trademarked the phrase.⁹⁸ Much of the initial

⁹⁶ Novell. "Novell Settles One Antitrust Claim with Microsoft for \$536 million, Plans to File Suit for Second Claim". Novell Press Release. Nov 8, 2004.

<http://www.novell.com/news/press/archive/2004/11/pr04076.html>

⁹⁷ Silwa, Carol. "Q&A: Novell CEO touts Linux, Identity Products". Computerworld. April 27, 2005. <http://www.techworld.com/opsys/features/index.cfm?featureid=1385&Page=1&pagePos=17>

⁹⁸ Sun Microsystems. "Sun Trademarks". Sun Trademarks Homepage. <http://www.sun.com/suntrademarks/>

Arpanet/Internet was comprised of UNIX servers, which is not surprising as both UNIX and the Arpanet/Internet were used predominantly within academia⁹⁹. Today, over 75% of web servers are either UNIX or UNIX-derivatives systems (i.e. Linux)¹⁰⁰.

The UNIX vendors were certainly well-positioned to become powerful networking platform vendors. UNIX systems enjoyed (and still enjoy) tremendous popularity within the research and academic community, and particularly within the “geek” hacker class of young technophile college students and computer programmers who would later contribute to much of the infrastructure of today’s Internet. UNIX systems, via the integration of BSD into the S5R4 release of UNIX, were one of the first systems in the world to incorporate a fully functional and highly reliable TCP/IP protocol stack that also offered a powerful programming interface through the Berkeley Sockets API that has become the de facto standard for not only UNIX network programming, but has served as the basis for the networking programming interfaces for several other system socket API’s, most notably the Microsoft Windows Sockets API.¹⁰¹ Today, any interested person can install a version of Linux on a cheap PC, pick up a few books, and learn how to write in Sockets. Contrast that with the significantly larger barriers to entry to learning how to develop a Netware Loadable Module or even trying to obtain development access to an SNA system, and you can see why an entire generation of computer science students has been weaned on the Sockets method of network protocol development and knows no other way.

Much of the Internet that we take for granted is based on network services that were built on top of UNIX first, and for which UNIX still remains dominant. UNIX services are known as ‘daemons’ in UNIX parlance, where the term daemon signifies a spirit from Greek mythology that handles tasks for the gods¹⁰². Many UNIX network services are therefore designated by appending the letter ‘d’ to the end of the name of a process. Most Web servers run the freeware open-source Apache HTTP (Hypertext Transport Protocol) server which is itself derived from the HTTPD server that was developed at the National Center for Supercomputing Applications (NCSA) at the University of Illinois, the same place where Mark Andreessen was concurrently developing the first Internet browser, Mosaic. The explosion in popularity of the World Wide Web due to the advent of the browser and the web-server has been well-chronicled and need not be repeated here.

However, the Web is only one subsection of the Internet. Not only does the Internet offer a bevy of other networking features other than the Web, the proper functioning of the Web actually intimately depends on some of those other features. Most notable of them would be email, which was and still is the killer application of the Internet. Email is still

⁹⁹ Raymond, Eric Steven. “Origin and History of UNIX, 1969-1995”. Chapter 2: History. 2003. <http://www.faqs.org/docs/artu/ch02s01.html>

¹⁰⁰ Broesma, Matthew. “At Web Sites, Windows Outpaces Linux”. CNET. July 18, 2003. http://news.com.com/At+Web+sites,+Windows+outpaces+Linux/2100-1016_3-1027188.html

¹⁰¹ Microsoft Inc. “Windows NT Server Resource Kit” Appendix D – Windows Sockets. May 1, 2006. http://www.microsoft.com/resources/documentation/windowsnt/4/server/reskit/en-us/net/sur_tsok.msp

¹⁰² Reference.com. “Daemon (mythology)”. [http://www.reference.com/browse/wiki/Daemon_\(mythology\)](http://www.reference.com/browse/wiki/Daemon_(mythology))

predominantly provided by UNIX/Linux applications such as sendmail (which was historically been the market leader of mail servers), Exim, Postfix, and several others. As can be seen in figure 14, UNIX/Linux mail applications represent at least 40% of the mail server market as of April 2004.

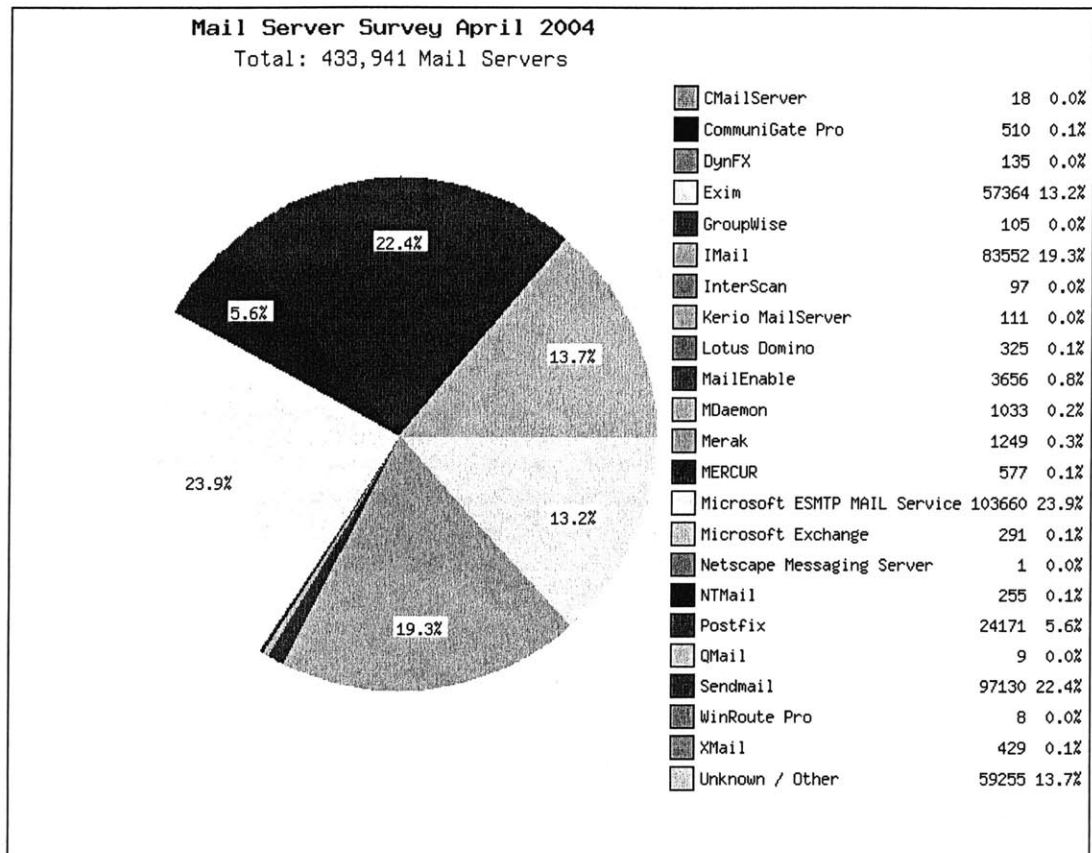


Figure 14 – Mail Server Market Share¹⁰³

Proper function of the Internet requires a user-friendly way to translate between Universal Resource Locators (URL's) such as www.yahoo.com and the IP address that correspond to that URL. Hence, a naming service is required. The Domain Naming Service, or named (for the naming 'daemon') has been baked into most versions of BSD-derived UNIX versions since the 1980's. DNS is integral to most interactions on the Internet from pointing Web browsers to the correct IP address to allowing mail servers to

¹⁰³ Falkotimme.

"Mail Server Survey". April 2004. http://www.falkotimme.com/projects/survey_smtp.php?id=170

find each other to deliver email to allowing instant messenger services to locate servers, and a wide range of other infrastructural duties. Microsoft embraced DNS only haltingly until the launch of Windows 2000¹⁰⁴ – prior to that, Microsoft networks tended to rely on the proprietary Windows Internet Naming Services (or WINS) naming protocol. Instant messaging and Internet chat trace their lineage to the venerable UNIX *talk* utility¹⁰⁵. Peer-to-peer Internet services are basically a modified UNIX ftpd (File Transfer Protocol Daemon) coupled with a locator application that lets peers discover each other. UNIX also offers a smorgasbord of directory services such as the Network Information System (NIS) pioneered by Sun and the Directory Application Protocol service which spawned the Lightweight Directory Application Protocol (LDAP) which is by far the most popular type of computing directory service implemented today. Novell's NDS/eDirectory and Microsoft's Active Directory are nothing more than proprietary implementations of LDAP. Hence, the UNIX vendors were well ahead of their time in offering an entire networking platform.

Furthermore, UNIX servers offered routing functionality starting in the late 1980's, and thus offered the advantages of combining routers and servers that Novell Netware servers enjoyed. Bill Joy's BSD modifications of UNIX included the rpd Routing Information Protocol daemon which provided UNIX systems to act as a basic router running a distance-vector routing protocol. Other routing utilities like the gated (the gateway daemon) and the Zebra GNU routing software package provided full OSPF link-state routing capabilities as well as the BGP protocol to link various routing domains together. Hence, UNIX vendors could sell their systems as servers, or could customize them to be routers, or as both. Even Cisco IOS shows heavy resemblances to UNIX, as can be seen by the output of the **show processes** IOS command that looks fairly similar to the output of the UNIX **ps** utility, where IOS processes are known as UNIX commands (CMD).

```
router#show processes
CPU utilization for five seconds: 0%/0%; one minute: 0%; five minutes: 0%
  PID Q   Ty      PC  Runtime(ms)   Invoked    uSecs      Stacks  TTY Process
   1 C   sp  602F3AF0      0         1627      0 2600/3000  0 Load Meter
   2 L   we  60C5BE00      4          136     29 5572/6000  0 CEF Scanner
   3 L   st  602D90F8  1676          837    2002 5740/6000  0 Check heaps
   4 C   we  602D08F8      0           1      0 5568/6000  0 Chunk Manager
   5 C   we  602DF0E8      0           1      0 5592/6000  0 Pool Manager
   6 M   st  60251E38      0           2      0 5560/6000  0 Timers
   7 M   we  600D4940      0           2      0 5568/6000  0 Serial Background
```

Figure 15 – IOS show processes output¹⁰⁶

¹⁰⁴ Microsoft Inc. "Windows 2000 DNS". Microsoft Technet. May 1, 2006.

<http://www.microsoft.com/technet/prodtechnol/windows2000serv/plan/w2kdns2.mspx>

¹⁰⁵ Techiwarehouse. "Instant Messaging". May 12, 2003.

http://www.techiwarehouse.com/cms/engine.php?page_id=1746eedd

¹⁰⁶ Cisco Systems. "Cisco IOS Software Releases 12.0 Mainlane: The Show Processes Command"

Document ID 15102. Cisco IOS documentation. May 6, 2006.

http://www.cisco.com/en/US/products/sw/iosswrel/ps1828/products_tech_note09186a00800a65d0.shtml#s_howproc

```
$ ps -ef
  UID    PID  PPID  C   STIME      TTY      TIME    CMD
    root   736   724   0  15:17:53 pts/4    0:01    sh
    root   737   736   2  15:19:16 pts/4    0:02    elm
    root   595   594   0  14:05:46 pts/3    0:00    sh
    root   599   595   0  14:06:00 pts/3    0:01    tcsh
    root   594   357   0  14:05:45 pts/0    0:02    xterm
 othello  724   722   0  15:13:06 pts/4    0:01    -csh
    root   722   104   0  15:13:06 ?        0:00    in.telnetd
    root   747   744   2  15:20:08 pts/5    0:00    ps -ef
```

Figure 16 – The UNIX ps command¹⁰⁷

So why are the UNIX networking vendors only ‘pretenders to the networking throne’, and not comfortably ensconced on the throne? The first and most fundamental reason is that many of the UNIX vendors were lukewarm at best towards the Internet, an ironic stance to take when you consider how dominant UNIX was in the delivery of Internet services. The majority of the UNIX flavors were sold by large computer conglomerates with a wide variety of products, including products that did not fit well within the Internet or TCP/IP paradigm. In a number of cases, vendors would actually make significantly less money by selling its UNIX systems than selling its true bread-and-butter. IBM was far more incentivized to sell mainframe systems and SNA networking components than AIX UNIX servers. The management at DEC almost preferred to pretend that it didn’t have the Ultrix UNIX (later known as Tru64 UNIX) operating system to sell you, strongly preferring to instead sell its homegrown OpenVMS OS running DECnet. Ken Olsen, the cofounder of DEC, even said that “UNIX is snake oil” although he later claimed that he was misquoted.¹⁰⁸ HP was deeply conflicted – it could sell you HP9000 systems running the HP-UX flavor of UNIX, or it could sell you those same systems running the Domain/OS software that it acquired from Apollo. Novell offered UnixWare as a consequence of Novell’s acquisition of the Unix Systems Lab and the Unix System V source code from AT&T, as a way to blend the best features of Netware and UNIX, but after Ray Noorda’s ouster as CEO from Novell, UnixWare lost its biggest backer in management and UnixWare became stagnant, ultimately being sold to the SCOGroup¹⁰⁹.

Hence, until the rise of the various Linux vendors, there was only one UNIX vendor that was 100% enthusiastic about the rise of both UNIX and the Internet, and that was Sun. And that was only be happenstance as Sun started life as a workstation vendor and could have easily stayed that way if not for the extraordinarily fortuitous acquisition of the Cray Superserver business from SGI. The superserver business ultimately became the Sun E10000 line of servers which allowed Sun to exit the brutally competitive workstation market which they were almost certainly destined to lose to Windows NT and instead

¹⁰⁷ Heard, Kevin. “UNIX Tutorial: Section 12 Viewing Processes”. Berkeley SIMS. Jan. 27, 2006. <http://www.sims.berkeley.edu/~kevin/unix-tutorial/section12.html>

¹⁰⁸ Sysprog. “Quotations -> Operating Systems”. May 15, 2006. <http://www.sysprog.net/quotos.html>

¹⁰⁹ Grygus. “SCO – Death Without Dignity”. AAXnet.com. Jun 2003.

become primarily a server business¹¹⁰. Yet even having one enthusiastic backer of the Internet within the UNIX community still meant that UNIX as a whole was destined to have problems in becoming a strong and credible source for networking technologies.

The second major factor for why UNIX became a pretender to the networking throne is related to the first reason – the proliferation of various versions of proprietary UNIX flavors, few of which were either software or hardware compatible, precipitated a series of market battles known as the UNIX wars. Having a war is bad enough, what was even worse was that the outcome of the war was inconclusive. Various vendors would take various sides to support different technologies and different standards and sometimes switch sides to further their strategic aims. The result was to create great customer confusion and distract the attention of the UNIX vendors from competition from the PC vendors¹¹¹. A competitive response would have to wait until a certain unknown university student in Finland developed his own Unix clone and posted it on Usenet. Linux would prove to be able to stave off the competition from Microsoft, but only at the expense of accelerating the decline of traditional Unix. The entire Sun business model, for example, is greatly discomfited by the rise of Linux¹¹².

The third and final factor is that, quite simply, the dedicated networking vendors like Cisco stole the thunder of the Unix vendors. Sun or one of the other Unix vendors could have become the dominant router vendor in the world. In fact, the very first routers in the world used Sun processing boards¹¹³. At the very least, the Unix vendors could have become significant vendors of IP routers. Unix vendors started offering routing features in the late 1980's, a mere few years after Cisco was born, and the much larger Unix vendors could have either acquired Cisco or crushed it. The fact that they did neither and Cisco thrived is both a tribute to the deft play of Cisco and of the foolishness of the Unix vendors. In fact, this would probably serve as an excellent segue into the next major topic of this paper.

5. Cisco Systems – the King of the Networking Platform Vendors

Cisco has been one of the greatest success stories of our times. To talk about Cisco is to talk in superlatives. Starting from the stereotypical humble beginnings of a typical Silicon Valley technology startup, Cisco grew like a weed on steroids. In 1998, Cisco market value reached \$100 billion, which made it the fastest company up to that time to ever reach such a market capitalization¹¹⁴. Cisco dominates the datacom networking world of today with market share of over 50% in almost every single subsector of the

¹¹⁰ Morgan, Timothy Prickett. "Sun to Buy Supercomputer-Maker Cray?" IT Jungle. June 30, 2004. <http://www.itjungle.com/breaking/bn063004-story02.html>

¹¹¹ Raymond.

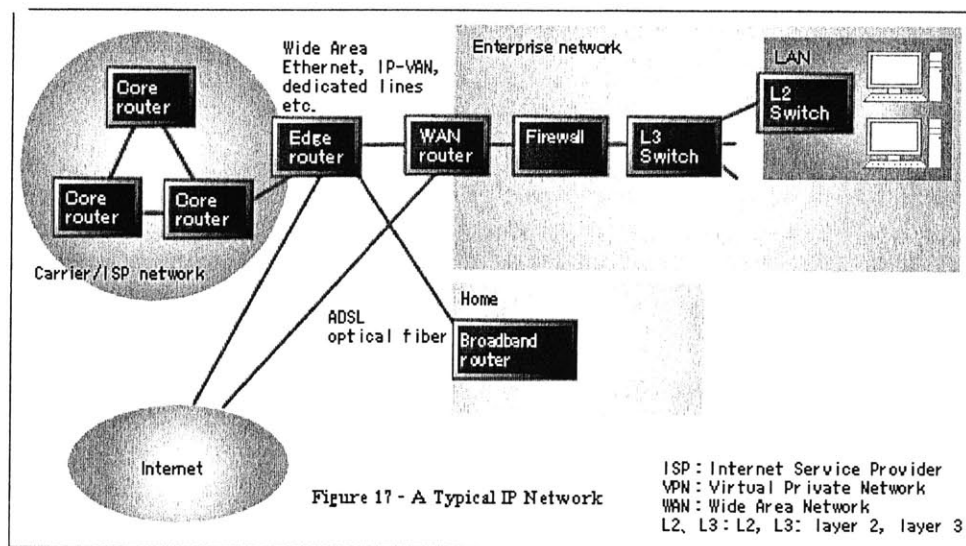
¹¹² Greene, Thomas C. "Sun's Linux Killer Shows Promise". The Register. Aug 16, 2005. http://www.theregister.com/2005/08/16/solaris_x86_not_too_shabby/

¹¹³ Cringely. "Valley of the Nerds: Who Really Invented the Multiprotocol Router, and Why Should We Care?". I, Cringely. Dec. 10, 1998. <http://www.pbs.org/cringely/pulpit/pulpit19981210.html>

¹¹⁴ Wollack, Todd. "Cisco Passes Microsoft in Worth". San Francisco Chronicle. March 28, 2000. <http://www.sfgate.com/cgi-bin/article.cgi?file=/chronicle/archive/2000/03/28/MN1065CH.DTL>

router market, and around 90% of the market for branch and access WAN routers¹¹⁵. Cisco is also one of the most respected company in the world, consistently earning high marks in the industry lists as one of the Most Admired companies on the planet and one of the Best Companies to work for.¹¹⁶

In fact, Cisco could almost be said to be synonymous with corporate networking, especially Internet Protocol (IP) networking. The datacom networking world is filled with a bewildering set of hardware terms such as core routers, edge routers, layer 3 switches, layer 2 switches, firewalls, WAN routers, broadband routers, and so on. The differences between the various pieces of hardware are rather technical and for the purposes of this discussion not highly important to understand, but what is important to understand is that every single one of the markets for those hardware pieces are dominated by Cisco. The majority of computer networking transmissions passes through at least one Cisco device somewhere along its path. Figure 17 illustrates the architecture of a typical IP network, and where various networking hardware infrastructural pieces reside within the network. Note that the majority of this hardware will be Cisco.



5.1 In the beginning

The roots of Cisco sound like they came from a fairy tale. Two Stanford graduate students met, fell in love, and got married. The groom, Len Bosack, graduated with a master's degree in computer science and took a job as a staffer in the Stanford CS department. The bride, Sandy Lerner, had finished her master's degree in statistics and computer science, and was working as an IT director at the Stanford Graduate School of

¹¹⁵ Duffy, Jim. "Juniper Gains Corporate Network Ground". Network World. Oct 10, 2005.
<http://www.computerworld.com.au/index.php/id;1133614807;relcomp;1>

¹¹⁶ Fortune Magazine: "100 Best Companies to Work For".
<http://money.cnn.com/magazines/fortune/bestcompanies/>

Business. The two departments had their own local LAN's, but data could be passed from one department to another only through the Arpanet terminals that each department had. Legend has it that the young newlyweds were lovesick and wanted to pass love e-letters to each other during working hours but wanted to do so without always having to use the Arpanet terminals. Whatever the truth really was, the fact is, the two began to devote their time and their talents to the problem of connecting multiple disparate types of networks in a way that would be seamless to the systems involved.

However, first, some myth-destruction was in order. First off, contrary to legend, Bosack and Lerner did NOT invent the router. In fact, if anybody should be credited with inventing the first router, it should be Bill Yeager, a computer staffer working at Stanford Medical School, who was commissioned with the task of designing the first router. As one pundit puts it:

"A Stanford researcher, Bill Yeager, who worked for the SUMEX-AIM resource, located in the Medical School, was assigned the task to produce the router technology. By June, 1980, PDP11/05 based router was in place which connected the medical school and department of computer science. By 1981 Yeager developed a unique network operating system, which would be the basis for the MC68xxx version of the code. This was completed later that year, and was ultimately licensed by Cisco Systems....In the final royalty agreement that Cisco Systems signed, Yeager granted 15 percent of these royalties to the department of computer science for the contribution."¹¹⁷

Yeager also had substantial assistance, most notably from other Stanford staffers like Kirk Loughheed and students such as Benjy Levy and Bill Nowicki. Hence, the notion that Len and Sandy invented the router all by themselves in order to satisfy their passion for each other is a nice fantasy, but nothing more than that.

Nor was Cisco the only, or even the first company to offer internetworking technology during that time. 3Com, which was already a giant in Ethernet adapters, was starting to branch out into simple interconnection hubs and bridges. Protean Associates was building interconnection technology for Token Ring networks. IBM of course was already selling a full suite of SNA interconnection gear. And, as mentioned before, a number of server vendors, notably the UNIX vendors, would only a few years later be selling servers that could act as RFC-compliant routers. However, Cisco had the advantage that they were the first company to commercialize a truly intelligent router, within which packets could be manipulated as needed and not just a simple hub that just served to interconnect networks but with no intervening processing intelligence. The difference was like connecting two roads with a system of traffic lights at the

¹¹⁷ Cringely.

intersection, and just connecting two roads with no means for traffic control. Excellent management helped too. Cisco's VC's installed highly competent management, especially Cisco's first CEO and current Chairman of the Board, John Morgridge, who implemented one of the first Internet-centric ordering and customer-service systems and vied to make its operations as automated and Internet-centric as possible. Cisco basically used its internal systems as a showpiece to demonstrate what customers could gain through good networking.

In fact, Cisco's information technology systems were vaunted for the ability to perform the "virtual close", meaning the ability to completely close the company's books electronically and provide detailed financial reports within a single day¹¹⁸. This was a remarkable feat that even to this day companies have great difficulty achieving because most companies have still not bothered to properly unify their information technology systems and place all of the information about the company's financial health into a single repository. Larry Ellison, founder of Oracle, once famously railed about the fact that Oracle was paying many millions of dollars to run a vast number of internal customer and accounting information systems yet still had tremendous difficulty in closing the books whenever earnings needed to be announced¹¹⁹.

5.2 The internetworking wars

As Cisco grew up, it attracted a tremendous amount of competition. Wellfleet offered competing routing technology that was highly reliable and scalable, arguably more scalable than Cisco's was. Cabletron was offering what were called intelligent hubs, which preserved the simplicity of the hub but included some advanced management features and more importantly could connect various LAN technologies together, like Ethernet to Token Ring, and process traffic to translate between the LAN standards as needed. Synoptics was extremely strong in offering standard (non-intelligent) hubs and was getting into attempting to branch out into other internetworking fields as well. Hubs didn't scale well but they were significantly cheaper and faster than routers were. Other startup router vendors were cropping up that offered superior performance in certain niche markets.

Cisco came out on top of the scrum with a two-pronged strategy. To deal with the other router vendors, especially Wellfleet, Cisco opted to make its routers as multi-protocol as possible. That meant that a Cisco router could be used to interconnect any networking protocol suite a customer might be running. Many customers found themselves with a giant mishmash of different and incompatible computer systems – they might have a DEC network running DECnet and a Novell IPX network and UNIX systems connected through IP and a bunch of Apple Mac's that needed to be connected through Appletalk. A mono-protocol router would require entirely separate links and entirely separate routers for each one of those networks. However, a multiprotocol router offered the promise of connecting all of these different systems on the same network and with only one set of

¹¹⁸ Stauffer, David, "Nothing but Net. Business the Cisco Way". Capstone Publishing. 2001. p. 68.

¹¹⁹ Symonds, Matthew. "Softwar: An Intimate Portray of Larry Ellison and Oracle." Simon & Schuster. 2003. P. 46-50.

routers. The industry term that is used to describe it is “ships-in-the-night” routing, where a router can hold an Appletalk routing table, an IPX routing table, a DECnet routing table, and so on, and each of these tables will be consulted separately and will not interfere with one another.¹²⁰ However, a more modern name for it would be *convergence*. Multiprotocol routers allowed multiple services to be converged onto the same links and the same routers.

Multiprotocol routers became known as the safe choice to make for corporate IT departments. They could buy such routers knowing that they would be able to route any sort of traffic that the company might later want to run. They could also be easily repurposed as necessary. If, say, the DECnet network were to shrink and the IPX network were to grow (as happened rather frequently as DEC declined and Novell expanded), then Cisco routers that were servicing the defunct sections of the DECnet network could be shunted over to handle the expanding IPX network. Cisco routers became known as the most multiprotocol of all the routers – offering the ability to route IP, IPX, Appletalk, DECnet, Apollo’s Domain protocol, XNS, Vines, the CLNS protocol that was popular with some government and military networks, the MIT-derived CHAOSnet protocol, the Xerox PUP protocol (which was the precursor to XNS), and several others. No other router vendor came close to matching this protocol breadth.

The other method that Cisco used to deal especially with the hub and switch vendors was Cisco’s vaunted acquisition strategy. Simply put, Cisco followed the time-honored adage that “if you can’t beat ‘em, join ‘em”. However, Cisco didn’t make it a point of simply acquiring companies that had the technologies it needed. Cisco made it a point of having highly successful acquisitions. Cisco’s acquisition of the startup switch-maker Crescendo Communications in 1993 was arguably the greatest acquisition in the storied history of Cisco, and one of the most important in the IT industry¹²¹. Crescendo technologies became the Cisco Catalyst 5000 and then the Catalyst 6000 and now the modern-day Catalyst 6500 line of switches, which is arguably the most successful line of switches in history, with just the Catalyst 6500 switch line alone bringing in about \$6.7 billion, or about 1/3 of Cisco’s revenue in the year 2004.¹²² Cisco further rounded out its Catalyst switch line with the 1994 acquisition of Kalpana which eventually became the Catalyst medium-range 3000/3500 line of stackable switches and the 1995 acquisition of Grand Junction which became the low-end Catalyst 1900 switches and hubs, and the 1996 acquisition of Granite Systems to provide Gigabit Ethernet switching technology. Cisco therefore offered a complete set of switching and hub technologies and thus gave the switch competitors no oxygen to breathe.

Of course it’s easy to say that a company should just make highly successful acquisitions, but clearly such a thing is quite difficult to actually pull off. Cisco devised a method of performing acquisitions that has proven to be so successful that much ink has been

¹²⁰ Pulse. “What is OSI Routing?” 2000. http://www.pulsewan.com/data101/osi_routing_basics.htm

¹²¹ Reardon, Marguerite. “The Skinny on Cisco’s Product Strategy”. CNET. June 9, 2004. http://news.com.com/The+skinny+on+Ciscos+product+strategy/2008-1033_3-5223552.html

¹²² Hochmuth, Phil & Duffy, Jim. “Cisco to Juice 6500 Switch”. Network World. Aug 15, 2005. <http://www.networkworld.com/news/2005/081505-cisco.html>

consumed in elucidating its principles. Current Cisco CEO John Chambers lists the following 5 rules of thumb that are used to size up a buying opportunity¹²³.

- **Shared Vision.** “You’ve got to be in agreement with where you think the industry is going and what role each partner wants to play in that.”
- **Cultural compatibility.** “If your cultures are different, they just never merge”
- **Geographic Proximity.** Widely separated units may “never get the efficiencies” of having key people in a single location.
- **Short-term wins.** Acquired employees must quickly find benefits in acquisition.
- **Long-term wins.** Projections must show gains for all four of Cisco’s key constituencies: shareholders, employees, customers, and business partners.¹²⁴

Furthermore, it should be noted that Cisco’s acquisitions have not always been successful. Nor has Cisco always followed its own rules of thumb. John Chambers once stated that “Probably 80 to 95 percent of our acquisitions have worked. It’s off the charts.”¹²⁵ However, many people, both inside and outside of Cisco, find that claim to be highly optimistic at best.¹²⁶ In particular, some of Cisco’s acquisitions have gone poorly, such as the notorious \$4.7 billion acquisition of Stratacom, a major vendor of frame-relay and ATM WAN switches, which even John Morgridge freely acknowledges has been extraordinarily problematic. Morgridge noted in particular that, among other things, “Cisco wasn’t successful in integrating Stratacom’s sales force.”, and was not successful in handling the customer discontent associated with customers purchasing Cisco’s previous WAN-switch line, Lightstream, which was now being sidelined in favor of Stratacom.¹²⁷ Cisco did not even adhere to its own rules of thumb when deciding to acquire Stratacom. Not only did key Stratacom employees, especially the sales staff, not enjoy any short-term wins, which motivated them to leave quickly, but Stratacom was also a well-established company with a culture that clashed with Cisco¹²⁸. In fact, Cisco may have inadvertently created one of its most powerful competitors by purchasing Stratacom – as one of the founders and executives of Stratacom, Scott Kriens, quickly left to become CEO and Chair of Cisco’s arch-rival, Juniper Networks¹²⁹.

¹²³ Stauffer, p. 172-173.

¹²⁴ Ibid.

¹²⁵ Slater, Robert. “The Eye of the Storm. How John Chambers Steered Cisco Through the Technology Collapse.” Collins Publishing. 2003. p. 214. s

¹²⁶ Ibid.

¹²⁷ Bunnell, David & Bate, Adam. . “Making the Cisco Connection.” John Wiley & Sons. 2000. p. 84-86.

¹²⁸ Mayer, David. “Ecosystems and Acquisition Management: Understanding Cisco’s Strategy”. <http://hcd.ucdavis.edu/faculty/kenney/articles/siliconvalley/cisco%20ver.%203.doc>, Industrial and Corporate Change (forthcoming), Dec 22, 2003.

¹²⁹ Juniper Network Company Profile. May 20, 2006.
<http://www.juniper.net/company/profile/leadership.html>

However, what is undeniable is that Cisco has been an extraordinarily acquisitive company that, while certainly not perfect in this regard, has certainly done better than many of its rivals. Cisco has managed to successfully round out its product line through acquisitions and has managed to successfully if sometimes painfully integrate most of the technology that it has acquired. Even the Stratacom acquisition has resulted in Cisco's integration of WAN-switching features into the latest versions of IOS and the acquisition of Stratacom's important sales accounts, most notably AT&T.¹³⁰ Hence, it could be said that even Cisco's worst acquisitions were still moderately beneficial to Cisco.

5.3 Cisco's foray into the SNA world

Arguably the most successful piece of Cisco's multiprotocol strategy was its decision to enter the SNA networking world. At the time, IBM's SNA was the most successful networking platform in the world. Cisco began competing in SNA by licensing SNA and offering simple SNA hardware replacements and living under the IBM pricing umbrella, similar to what many other SNA compatible vendors were offering. However, Cisco was undoubtedly the most successful of the compatible vendors, in fact, so much so that Cisco ended up essentially killing IBM's SNA business and replacing it with SNA/IP convergence technology of which Cisco was the clear leader.

While Cisco started life in the SNA world as a simple SNA-replacement vendor, Cisco was quick to pioneer several unique SNA replacement technologies. In 1992, Cisco became one of the first vendors to offer SDLLC (Synchronous Data Logical Link Controller)¹³¹, a method to merge SDLC technology that is the standard WAN technology to carry SNA with the Local Link Controller technology that is central to Token Ring. Essentially, SDLLC allowed for SNA WAN and LAN technology to be fused together in a way that is completely seamless to the rest of the SNA network. SDLLC operates by making the SDLC part of the network look like an extension to the LAN of the Token Ring network via the concept of the "virtual ring", and as an extension of the WAN by disguising the LAN section of the network. Hence, the Token Ring LAN section of the SNA network simply thinks it's connected to a giant LAN, and the WAN section simply thinks it's connected to a giant WAN. One example of this, showing the true Token Ring LAN and the virtual ring emulated by a Cisco router is shown in figure 18.

¹³⁰ Greene, Tim & Rohde, David. "MCI Worldcom's Frame Net Face Lift". Network World. Nov. 9, 1998. <http://www.networkworld.com/news/1109mciframe.html>

¹³¹ Cisco Systems. "Flexible Migration Paths for IBM Networking". Document ID 12305. Cisco Technology White Papers. . Sep. 9, 2005. http://www.cisco.com/en/US/tech/tk331/tk336/technologies_white_paper09186a0080114a38.shtml

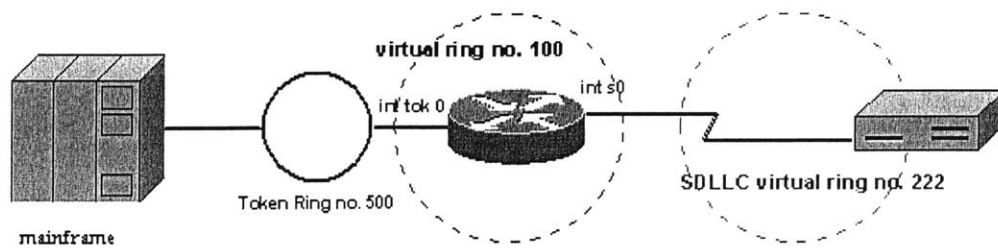


Figure 18 – An example of an SDLLC network¹³²

However, far more important than the simple SNA-internetworking technologies that Cisco developed are the Cisco technologies that allowed customers to merge SNA with IP. The first such offering was RSRB as was discussed previously, which was the first means of carrying SNA traffic over an IP backbone. Cisco therefore leveraged its strength in IP to provide a transport alternative over which SNA traffic could be transmitted. This was the first step in the marginalization of SNA as a networking platform.

In 1992, Cisco developed the STUN (Serial Tunneling) technology and its cousin, BSTUN (bisynchronous serial tunneling) which provided the ability to transport any protocol over an IP network¹³³. While STUN gets very little press, STUN and BSTUN were arguably the most aggressive technologies that Cisco has ever devised. STUN and BSTUN offered the capability of completely deconstructing any network protocol down to the basic wiring signals, transporting those raw signals over an IP backbone, and then reconstructing those signals, wire-by-wire on the other end. In other words, Cisco basically put its stake in the ground in stating that even if IBM were to completely rebuild SNA from scratch or even come up with a brand-new networking protocol entirely, Cisco would be able to perfectly emulate it by reading every electronic signal of that new protocol and rebuilding it bit-by-bit on the other end. Cisco may not have a license to this new technology and may not even be able to understand, but no matter. Cisco would behave like the perfect copy machine. Just like a person doesn't need to understand Chinese in order to reproduce an exact copy of a Chinese character, Cisco asserted that it would need to understand nothing about SNA in order to emulate and reproduce an exact bit-by-bit copy of SNA through STUN/BSTUN. An illustration of STUN is shown in figure 19.

¹³²Cisco Systems. "SDLLC Direct Configuration Example". Document ID 12281. Cisco Configuration Examples and Technotes. Sep, 9, 2005.
http://www.cisco.com/en/US/tech/tk331/tk336/technologies_configuration_example09186a0080093d9a.shtml

¹³³ Cisco Systems. "Flexible Migration Paths for IBM Networking".

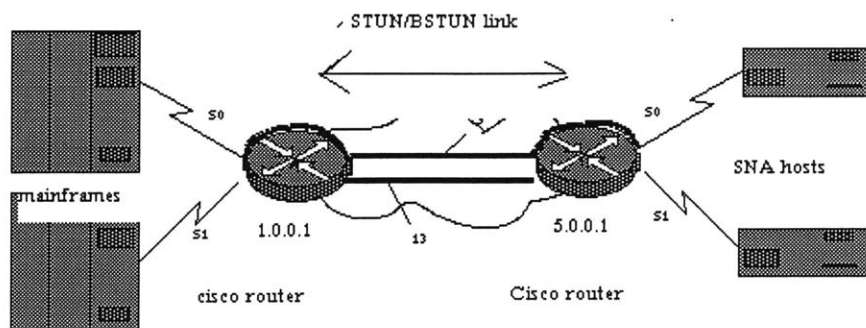


Figure 19 - STUN¹³⁴

Cisco implemented a number of other SNA conversion technologies as well. In addition to the leapfrogging of IBM's DLSw technology with the Cisco-proprietary DLSw+ extensions, Cisco in the early 90's developed the Channel Interface Processor, or CIP, cards and the Cisco Channel Port Adapter, or CPA, as add-on modules for the Cisco 7500/7200 routers respectively and that served as a complete replacement for the SNA Front-End Processor. Cisco's basic sales pitch was to ask "Why buy a FEP from IBM when you can buy a module for your router that not only does everything a FEP does, but also contains STUN and DLSW+ technology that allows for your SNA traffic to be directly converted onto your IP network"?

Cisco also engaged in a number of strategic acquisitions and partnerships to obtain technologies that it lacked to defeat IBM. In 1996, Cisco purchased Nashoba Networks to obtain token-ring switching technology and also entered a joint partnership with Olicom in which Olicom would provide Cisco with token-ring interface technology.¹³⁵ Much of Cisco's presence in the Token Ring networking market was due to the Nashoba acquisition and the Olicom partnership. Not only did they provide Cisco with the technology to sell into the Token Ring market, the Nashoba token-ring switching technology was something that IBM had great difficulty in matching, as IBM suffered from product delays and engineering weaknesses, particularly the glaring lack in its initial products of integration between switching and source-route bridging.¹³⁶ Cisco was therefore offering Token Ring technology that was actually technically superior to that offered by IBM, further contributing to Cisco's inroads into a market that IBM traditionally owned.

¹³⁴ Cisco Systems. "STUN Basic with Multiple Tunnels". Document ID 12275. Cisco Configuration Examples and Technotes. Sep 9, 2005. http://www.cisco.com/en/US/tech/tk827/tk369/technologies_configuration_example09186a00801241f6.shtml

¹³⁵ Cisco Systems. "Cisco Systems to Acquire Token Ring Switch Innovator Nashoba Networks". News@Cisco. Aug 6, 1996. http://newsroom.cisco.com/dlls/1996/corp_080696.html

¹³⁶ Cooney, Michael. "Determined but Battered IBM in Fight of its Token Ring Lift". Network World. May 13, 1996. http://www.findarticles.com/p/articles/mi_qa3649/is_199605/ai_n8745983

Furthermore, Cisco was instrumental in pioneering several other technologies that further marginalized SNA in favor of IP. TN3270 (or Telnet 3270) technology, as defined by RFC 1041¹³⁷, provided the means of taking standard 3270 traffic from a dumb terminal and converting it to IP. Cisco quickly baked TN3270 functionality into its router product line. Cisco also developed the Cisco Transaction Connection Service as a means to directly access both IBM's DB2 database and the IBM CICS operating system directly through IP¹³⁸. Cisco was a key member of the APPN Forum that pioneered the first peer-to-peer version of SNA. Ultimately, Cisco established a completely dominant position in SNA networking, by 1996, ranking "...number one in the IBM SNA/Internetworking marketplace, with 73% of the market share."¹³⁹ IBM ceded more and more of the SNA market to Cisco, by 1994, providing all of SNA's source code to Cisco, and on 1999, famously selling its entire networking division to Cisco for \$2 billion and converting itself to a Cisco reseller.¹⁴⁰ IBM tried to position the deal as an alliance and partnership with Cisco, but that is a thin reed to hide behind, as what IBM was really doing was resigning its king in the face of a *fait accompli* by Cisco. From this point forward, the traditionally most lucrative of all corporate networking fields, the SNA market, now belonged to Cisco. IBM's role in the SNA market has now been reduced to that of a simple enterprise server vendor and not that of a powerful network entity.

5.4 Integration of voice and video

When John Chambers once infamously remarked that "Voice will be free", a pronouncement that caused a major row between Cisco and the phone companies¹⁴¹, he was being unnecessarily confrontational, but he was also being prescient. Voice communications are now just another form of data, to be transported across data networks that have been tuned with QoS guarantees that can provide the low delay and echo-cancellation that have become the expected level of service of phone users everywhere. Since voice not only consumes small quantities of network bandwidth even when transmitted raw, but can also be compressed to consume even less bandwidth,¹⁴² Chambers was merely expressing the economic truism that voice will be free because competitive markets will dictate that marginal revenue will equal marginal cost. Since networks constantly expand in capacity in lockstep with the improvement of network chips via Moore's Law, the cost to transmit low-bandwidth voice packets will tend to approach zero over time. Furthermore, as the ubiquity of IP networks allows anybody to become an IP telephony service provider virtually overnight, increase market entry will inevitably mean that the telephony market will become more competitive and hence

¹³⁷ RFC 1041. <http://www.faqs.org/rfcs/rfc1041.html>

¹³⁸ Cisco Systems. "Cisco Transaction Connection Overview". Nov. 5, 2002. <http://www.cisco.com/univercd/cc/td/doc/product/ibd/ctrc/ctrcin.htm#xtocid3>

¹³⁹ Slater, p. 210.

¹⁴⁰ Songini, Marc. "IBM Has All But Thrown in the Towel on its Networking Hardware Division". Network World. Aug 31, 1999. <http://www.networkworld.com/news/1999/0831ibmcisco.html>

¹⁴¹ Reinhardt, Andy. "John Chambers". Businessweek Online. May 15, 2000. http://www.businessweek.com/2000/00_20/b3681028.htm

¹⁴² Biran, Gil. "Voice over Frame Relay, IP and ATM". Protocols. May 1, 2006. <http://www.protocols.com/papers/voe.htm>

pricing will tend to approach the competitive market equilibrium price, which would be zero. Chambers was therefore merely expressing a painful market truth that the telephony-dependent telcos didn't want to hear.

However, while voice probably will be free eventually, substantial amounts of revenue can still be made in the interim, particularly on the hardware side of selling new voice gear. Corporate PBX (private branch exchange) phone equipment, which essentially serve as miniature versions of the giant telephone switches that the telcos use in their Central Offices, represents a \$20 billion market.¹⁴³ Cisco chose to enter the PBX market in classic Cisco form – by acquiring companies with promising voice technology, by pushing its multiprotocol routing expertise and by incentivizing its channel partners.

Voice hardware revenue was not the only reason for Cisco to enter the voice market. While data networks have become an ever-more-critical piece of a typical company's daily routine, many company departments, particularly sales departments, would come to a dead halt if its phone systems were to go down. Cisco therefore saw voice as a way to increase its strategic grip on customers by providing a feature that is crucial to its customers' operations.

In 1998, Cisco acquired Selsius Systems, a voice hardware startup that offered a line of PBX's and corporate phones that ran on IP networks¹⁴⁴, and followed that up with acquisitions of other VoIP startups such as ActiveVoice for its voicemail capabilities, Sentient Networks that sold technology that allowed had voice-over-ATM technology, and Maxcomm for its voice-over-DSL technologies, all in an effort to shift voice communications away from the circuit-switched PSTN technologies that traditionally carried voice (and a market in which Cisco had no presence) to newer network technologies in which Cisco had a strong presence. Cisco followed this up with the giant \$2 billion acquisition of Geotel, a cell-center software vendor, which was at the time the 2nd largest acquisition in Cisco history.¹⁴⁵ The Selsius products later became known as the CallManager line of VoIP equipment, the ActiveVoice voicemail product was redubbed the Unity voicemail server, and Geotel's products were rechristened as the Cisco IP Contact Center suite. All of these technologies put together became the backbone of the Cisco Advanced Voice Video and Data, or the AVVID, voice strategy and marketing rubric. Cisco therefore acquired a portfolio of voice technologies that allowed for voice to be transported over a number of non-traditional networks, especially IP networks. This obviously played to Cisco's strength in IP.

Cisco also unsurprisingly bundled voice technologies into its routers and switches. Cisco integrated a mini-Selsius and a mini-ActiveVoice feature into IOS and sold them as Call

¹⁴³ Commweb. "VoIP/IP Telephony Statistics". Oct 15, 2002.

<http://www.techweb.com/article/COM20021015S0002>

¹⁴⁴ "Cisco Acquires Dallas-based Selsius Systems". San Francisco Business Times. Oct. 14, 1998.

<http://www.bizjournals.com/sanfrancisco/stories/1998/10/12/daily9.html>

¹⁴⁵ Heskett, Ben. "Cisco Raises the Voice Market Stakes". CNET. April 13, 1999.

http://news.com.com/Cisco+raises+the+voice+market+stakes/2100-1033_3-224348.html

Manager Express and Unity Express respectively¹⁴⁶. No longer would customers have to buy separate hardware servers to run voice over their IP networks, they could just pay a license fee to activate the voice features that already existed inside their routers. Cisco also repositioned its routers as voice gateways to convert VoIP packets to formats that would be understood by traditional phone networks. This was the key technology that allowed corporate VoIP networks to easily connect to outside PSTN and to other traditional PBX's that a customer may still have around. Hence, similarly to how Cisco built inroads into the IBM SNA market by providing a complete portfolio of technologies to transport SNA traffic over an IP core, Cisco built the same inroads to enter the corporate voice market. Figure 20 shows how a gateway-enabled Cisco router serves as the intermediary between the VoIP and traditional voice worlds.

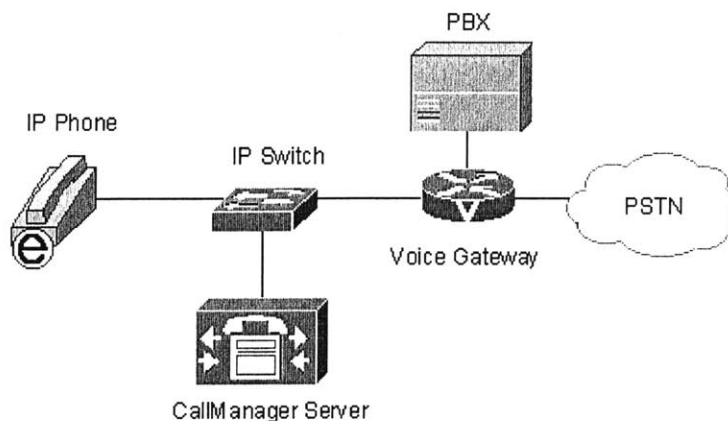


Figure 20 – The role of the Voice Gateway.

Cisco also integrated into its newer Catalyst switches a technology known as inline power. The world's consumer telephone systems are engineered in such a way that individual telephones draw power from the phone lines themselves as opposed to separate power line that draws power from a wall socket. Electric power would be provided directly from the telco's phone switch, which would be backed up by an array of large batteries. The phone network therefore works even during a power outage, an important safety feature in case one needs to call 911 during a blackout. Corporate PBX's emulated the same feature by powering corporate phones directly from the PBX through the phone lines, with the PBX's power being backed up by a battery backup system. IP phones, being essentially miniature computers running an IP stack, initially required power to be provided through a standard wall jack, and so would therefore be rendered useless were power to be cut to the building. Corporate PBX manufacturers hence would constantly tout the public safety dangers of IP phones, emphasizing that a customer might be sued if employees needed to dial 911 during an emergency but were unable to do so because a power outage had cut off the customer's IP phones. In response, Cisco implemented a

¹⁴⁶ Krapf, Eric. "Cisco Folds IP-PBX into IOS". Business Communications Review. Nov 1, 2003. <http://www.bcr.com/bcrrmag/2003/11/p56.php>

method for sufficient electricity to power an IP phone to be carried over an Ethernet cable. This power would be drawn *inline* from the Ethernet switch, and the switch, along with the accompanying Call Manager PBX server, would be connected to battery back-up systems. Hence, Cisco VoIP networks now enjoy the same protections against power blackouts as traditional voice networks do. As an additional benefit, Cisco switches would now be able to act as the “universal power source” for any small electronic devices that could be connected to a network, such as IP Webcams, IP video security camcorders, IP video monitors, etc. While some of these uses may not be popular at this time, the ability to provide inline power to IP phones eliminated a major obstacle in the migration from traditional voice to voice over IP, and therefore allowed for Cisco networks to encompass voice transmission.

Finally, Cisco provided strong marketing and resale incentives to its systems integration and reseller partners to sell VoIP. Cisco used a broad swath of resellers and integrators to widen the penetration of its technologies, and influenced what they sold through the management of the partnership program. Cisco made the voice specialization path one of the least stringent paths by which a reseller could become a Cisco partner and obtain discounts on Cisco gear by awarding more ‘partnership points’ to the voice path than to any other path.¹⁴⁷ Moreover, customers are far less likely to have in-house engineers who understand Cisco AVVID than understand standard Cisco IOS routing and switching¹⁴⁸, meaning that a Cisco systems integrator who has voice specialists will be able to charge far more for professional services to install and maintain a Cisco AVVID network. Hence, through a combination of strong Cisco resale discounts and the potential for large service sales, Cisco partners are heavily incentivized to popularize Cisco’s AVVID strategy, which serves to ensconce Cisco further into the heart of customer’s IT infrastructure.

6. The Cisco Networking Platform and its evolution

Before we can proceed further, we need to define what is meant by the Cisco networking platform. The networking platform consists of far more than just the technological portfolio that Cisco brings to the table, but also includes a number of ‘softer’ managerial strategies and market advantages that Cisco provides to further its grip on the networking market. It is this combination of hardware and software technology accompanied by strategy and market positioning that has allowed Cisco to rise to the top of the datacom world, defeating all challengers that it has faced to date. Cisco’s technology is itself considered to be unimpressive by many observers, with IOS in particular being the target of much criticism for a number of years as being unstable, for having security holes, and easy to misconfigure.¹⁴⁹ Cisco has also been frequently cited as a technology laggard

¹⁴⁷ Cisco Systems Partner Website. May 1, 2006.

http://www.cisco.com/warp/public/765/partner_programs/specialization/spec_partner_points.shtml

¹⁴⁸ Cisco Systems CCIE website. May 5, 2006.

http://www.cisco.com/web/learning/1e3/ccie/certified_ccies/worldwide.html

¹⁴⁹ Passmore, David. “A Modern IOS”. Business Communications Review. Oct. 3, 2005.

<http://www.bcr.com/bcrrmag/2005/10/p16.php>

that “hasn’t invented anything significant since the router [and it’s rather doubtful as to whether Cisco truly invented the router], and uses acquisitions as a crutch.”¹⁵⁰ However, as is often the case in many technology markets, it’s not just about having good technology. The best technology does not always win. Rather, it’s how you choose to wield the technology, and in particular, how you wrap your tactics and strategy around your technology that will determine whether it will become a dominant networking platform. Cisco has utilized all 6 levers to obtain and maintain its network platform.

6.1 Hardware componentization

While Cisco still retains substantial proprietary control over its hardware architectures, Cisco now outsources a substantial portion of its hardware from 3rd party vendors. Cisco therefore relies less and less on custom hardware and manufacturing facilities to derive its competitive advantage, preferring to incorporate more off-the-shelf components and manufacturing capacity.

Cisco used to design, develop and fabricate itself almost all of the special ASIC’s that provided Cisco routers and switches with their plethora of features. ASIC’s provided hardware vendors like Cisco with the ability to customize and fine-tune buffer capacities, registers, ring spaces, serializers/deserializers, and timing circuits that networking equipment needed to operate properly. However, Cisco has moved away from designing and especially from manufacturing its own ASIC’s, preferring instead to outsource much of those tasks to specialized ASIC vendors such as IBM, Vitesse, and Cypress Semiconductor¹⁵¹. Cisco and IBM have an especially strong partnership where the two companies collaborate to develop high-end network ASIC’s for Cisco’s most advanced routers, leveraging IBM’s vast state-of-the-art semiconductor fabrication capacity.¹⁵²

Cisco has also been quick to adopt the technology of network processors to replace ASIC’s. An ASIC is designed to do only what it has been fabricated to do in hardware. Designing an ASIC is like writing a software program with semiconductors. Hence, while ASIC’s provide you with maximum flexibility in terms of tweaking the semiconductor budget to optimize the performance of whatever features you designate, ASIC’s are completely unprogrammable. They can therefore only do precisely what its semiconductors have been positioned to do, but no more. You therefore cannot upgrade an ASIC to incorporate new features. Network processors, on the other hand, blend the capabilities of an ASIC and general microprocessor. They give you the ability to perform limited programmability and therefore provide limited upgradeability. Cisco has steadily reduced the importance of ASIC’s in its routers, replacing them with network processors,

¹⁵⁰ Reinhardt. “Meet Cisco’s Mr. Internet”. Businessweek. Sep. 13, 1999.
http://www.businessweek.com/1999/99_37/b3646001.htm

¹⁵¹ Morrison, Gale. “Outsourcing Comes to Network OEM’s”. Electronic News. Aug 27, 2001.
http://findarticles.com/p/articles/mi_m0EKF/is_35_47/ai_77673033

¹⁵² IBM. “Cisco and IBM Collaborate to Design and Build the World’s Most Sophisticated, High-Performance 40 Gbps Custom Chip” IBM Press Release. June 9, 2004. http://www-03.ibm.com/chips/news/2004/0609_cisco.html

notably with the Parallel Express Forwarding line of network processors in the Cisco 10000 series routers.¹⁵³

Finally, Cisco is moving away from the use of proprietary transport mechanisms that are internal to its hardware. While the very first Cisco router line, the AGS line, used standardized Multibus technology for its internal motherboard bus mechanism, Cisco quickly developed several proprietary bus technologies such as the Cbus (later dubbed the Cxbus), which was quickly followed up by the proprietary Cybus extensions. The Cbus provided a 533 Mbps of internal bandwidth that was microcoded which meant that it would be able to run independently of the router's general microprocessor¹⁵⁴. The Cybus doubled the available internal bandwidth to 1.066 Gbps. These proprietary bus technologies became the backbones of Cisco's highly successful 7000 and 7500 series routers. Cisco developed proprietary buses because no off-the-shelf bus technology fit its needs. However, as chip vendors improved their offerings, Cisco began to rely on 3rd party vendors selling bus controllers that complied to a number of industry-standard buses, most notably the standardized PCI bus. Cisco no longer develops proprietary bus technology.

To a lesser extent, Cisco has also shifted to off-the-shelf switch fabric technology for its Catalyst line of switches. While Cisco still designs many of its switch fabrics in-house, it now outsources an increasing amount of its switch-fabric capacity, particularly on its low-end switches. As low-end switching becomes commoditized, Cisco has engaged fabric vendors such as Broadcom, AMCC, and a number of startups such as Azanda in order to maintain its position in the low-end vendor market presence while keeping its product development costs low¹⁵⁵.

Cisco also engineers its gear to use standardized memory components, including DRAM, SRAM and flash. Cisco router "memory chips" are therefore really judged rebadged chips obtained from various vendors, either large-scale manufacturers like Micron for commodity DRAM or specialty manufacturers like Viking for customized jobs.¹⁵⁶ While these chips have to be 'certified' by Cisco, meaning that they undergo a number of certification and quality-control checks, in general these components involve relatively little input from Cisco. Cisco states the memory parameters that need to be satisfied, and the vendor delivers products that meet those parameters.

Finally, Cisco has made extensive use of contract manufacturing, being one of the first technology companies to do so. Cisco has long-standing relationships with the likes of Soletron, Jabil Circuit, and Flextronics, and outsources the bulk of its manufacturing to them. While the very first Cisco routers were often times bolted and screwed together by

¹⁵³ Lawson, Stephen. "Network Processors Enter New Generation". Network World. Jun 19, 2002.

<http://www.networkworld.com/news/2002/0619processors.html>

¹⁵⁴ Cisco Systems. "Product Overview – the Cisco 7010 Router". 1997.

<http://www.cisco.com/univercd/cc/td/doc/product/core/cis7010/7010ug/prodo1ug.htm>

¹⁵⁵ Matsumoto, Craig. "Azanda Lands a Buyer". Heavy Reading. Jan 20, 2005.

http://www.heavyreading.com/document.asp?doc_id=66312&site=lightreading

¹⁵⁶ Zycko Networking Products. "Product Catalog – Cisco Memory". 2005.

http://www.zycko.com/networking/products/networking_memory.asp

Len Bosack and/or Sandy Lerner themselves in their living room¹⁵⁷, nowadays, less than 10% of today's Cisco's hardware is actually directed manufactured by Cisco.¹⁵⁸ ¹⁵⁹As of late 2004, Cisco had less than 2000 employees on its staff devoted to manufacturing, compared to an additional estimated 12,000-14,000 manufacturing employees who manufactured Cisco products, but who were on the payroll of a contract manufacturer.¹⁶⁰

Hence, Cisco has utilized the first four levers to effect its hardware componentization strategy. Cisco has pulled lever 1 and redefined its role as being less of a full soup-to-nuts hardware vendor, instead choosing to increasingly source individual hardware components from 3rd party vendors. Cisco is using fewer ASIC's, and more off-the-shelf network processors. Cisco has scrapped the use of proprietary buses and is using fewer Cisco is therefore no longer trying to derive competitive advantage through better/more customized baseline hardware. Cisco now leaves much of the hardware component innovation and chip design to others, choosing instead to differentiate itself through systems component integration and software features.

Cisco has also utilized lever 2 to redesign its products to increase system modularity so that they can utilize whatever 3rd party vendor offers the best price. Whenever Cisco designs a new router or switch architecture, Cisco tries to standardize many of its component interfaces in order to maximize its ability to choose amongst outside vendors. This is particularly true with bus and memory interfaces which adhere to industry-standard input/output interfaces. Cisco also, whenever possible, tries to find second sources for its components, in order to be able to mix and match vendors as necessary. Hence, by maintaining open component interfaces, Cisco encourages wide-ranging and intense competition among its suppliers to develop features and cut costs.

Cisco manipulates lever 3 via various deep partnerships with its suppliers. Cisco engages in a number of long-standing strategic partnerships with various suppliers who are expected to be able to meet Cisco's exacting technical demands in return for long-term business. Cisco and IBM, for example, maintain a deep relationship to design and manufacture chips and chip adapters for Cisco's latest backbone router architecture, and Cisco has essentially handed off almost all of its manufacturing through partnerships with various contract manufacturers. Cisco has essentially signaled to the market that it is uninterested in becoming a semiconductor component company or a manufacturing company, and so it will never be a direct threat to those companies. Conversely, after IBM's loss of the SNA networking market, only once has one of Cisco's component vendors made a bona fide attempt to become a networking hardware vendor, and that was the short and ill-fated attempt by Intel to enter the networking market by acquiring Shiva

¹⁵⁷ Bunnell, p.8.

¹⁵⁸ Gibney, Frank. "You Name It, We'll Make It". Time Global Business. August 2001. <http://www.time.com/time/global/august/cover.html>

¹⁵⁹ Takahashi, Dean. "Crunching the Numbers". ESM. Nov 15, 2004. <http://www.my-esm.com/printableArticle.jhtml?articleID=51202436>

¹⁶⁰ Ibid

and selling home-grown networking gear under the Intel Netstructure and Express line of products.^{161 162}

Finally, Cisco has oriented itself internally as a 'virtual company', particular from a manufacturing and procurement standpoint. As stated above, Cisco owns relatively little manufacturing capacity and employs relatively few manufacturing personnel, and the people that Cisco does hire tend to be managers of outsourced manufacturing rather than people who have a direct hand in manufacturing. Cisco was recognized with the European Supply Chain Excellent Award in 2002 for its widely respected innovations in distribution and production.¹⁶³ Hence, Cisco utilizes lever 4 in molding its internal organization to emphasize the importance of outsourcing the component development and manufacturing functions that Cisco chooses to hand off to its partners.

6.2 Hardware Systems that are sold to Customers

From a hardware supplier standpoint, Cisco has chosen to engage in a substantial number of partnerships to offload much of its component development and manufacturing capacity, for Cisco feels that it will obtain little competitive advantage through superior base-level components or superior production facilities. Hence, from a supplier standpoint, Cisco's hardware platform is relatively open. Not so for Cisco's customers. Cisco has chosen to sell a quite closed hardware platform to its customers.

Cisco's most successful product lines are chassis-based switches and routers.. A chassis-based system is one in which individual expansion cards, called 'blades' or 'modules' that contain specific technical functionality, can be inserted into slots of the chassis. It is the combined chassis and blades/modules that provide a complete networking solution. Cisco sells a large variety of cards that provide varying quantities of interface types and features. Some of these cards provide more exotic networking features like firewalling, intrusion detection, VPN sourcing and termination, network packet analysis, caching and VoIP logic. Other cards, dubbed "Supervisor cards", govern the behavior of all the other cards in the chassis, such that better Supervisors result in better overall performance. In fact, some pundits have termed such a strategy the 'God-box' strategy in which a network vendor would ultimately like to cram an entire set of IT infrastructural features inside one fully-loaded chassis.¹⁶⁴ By design, only Cisco cards will work inside a Cisco chassis. Hence, Cisco sells a hardware platform that is not interoperable with any other vendor. A Catalyst 6500 that has its slots fully loaded with cards is shown in figure 21.

¹⁶¹ Intel Inc. "Intel to Acquire Shiva Corporation". Intel Press Release. Oct 19, 1998.
<http://www.intel.com/pressroom/archive/releases/HN101998.HTM>

¹⁶² Intel Inc. "Network Connectivity". Intel Support document. March 20, 2003.
<http://support.intel.com/support/network/sb/cs-009528.htm>

¹⁶³ UPS. "UPS Supply Chain Solutions Awards for Business Excellence". UPS Fact Sheets. May 15, 2006. <http://www.pressroom.ups.com/mediakits/factsheet/0,2305,1053,00.html>

¹⁶⁴ Ibid.

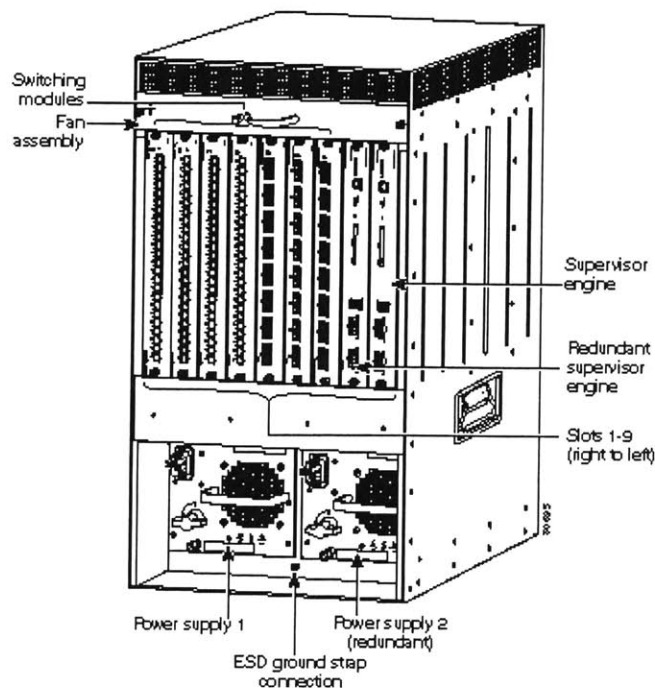


Figure 21 – A Catalyst 6500, with its expansion slots fully loaded¹⁶⁵

Now, to be clear, Cisco hardware is interoperable with hardware from other vendors that sit outside of the box. In fact, Cisco networking gear adheres to a long list of networking standards that provide for inter-hardware operability, a topic which will be discussed in greater detail in the IOS section of the paper. However, what is at stake here is the interoperability within a particular box. In the case of Cisco hardware, there is none. Only Cisco sells the blades and modules that will work inside a Cisco chassis. And vice versa is also true – only Cisco chassis will work with Cisco blades and modules. Cisco refuses to publish any of the hardware interface specifications that might allow for third-party vendors to sell ‘clone’ modules, blades, and chassis, and has never expressed interest in engaging with any vendors to provide such alternate hardware or industry standard. It is unlikely that a technical holdup is preventing Cisco from doing so. Several initiatives currently exist to standardize blade-servers¹⁶⁶. The Cisco chassis architecture is essentially the same as that of a blade server. It is therefore a conscious choice of Cisco’s not to open its hardware in this fashion.

On the one hand, you could say that Cisco has made a deliberate “Lever-1” decision to keep the entire hardware platform that it presents to its customers all to itself. Not only will this obviously enforce a uniform standard of quality, but more importantly, it maintains the lock-in that Cisco has over a customer. For example, Cisco’s chassis are generally engineered to contain as little intelligence as possible, and instead most of the intelligence is embedded within individual cards. For example, the Cisco Catalyst

¹⁶⁵ Cisco Systems. “Cisco 6500 Catalyst Switch”. http://www.asl-cisco.co.uk/images/screenshots/cdccont_0900aecd8006994b.gif

¹⁶⁶ Fried, Ina & Kanellos, Michael. “Dell Calls for Blade Server Standards”. CNET. Sep. 8, 2003. http://news.zdnet.com/2100-9584_22-5072603.html?tag=nl

chassis offer passive backplanes and are therefore little more than hardware housing and roadways for the cards to communicate with each other.¹⁶⁷ Since the chassis contains very little intelligence and functionality itself, that means that it rarely has to be upgraded. Functionality upgrades are to be done through “card-swaps” rather than through “chassis-swaps”. Furthermore Cisco engineers its cards in such a fashion that newer cards are backwards-compatible with older cards in the same chassis. Hence, a customer that finds that it needs incremental network functionality upgrade does not need to toss out an entire card-laden chassis, but can rather purchase the specific card that provides the desired functionality. Cisco calls this “investment protection” such that it assures customers that they can buy a Cisco chassis knowing that they will be able to grow their business with that chassis by purchasing appropriate cards as needed¹⁶⁸. However, another way to put it would be “sales account protection”. A customer that has already bought a Cisco router or switch and populated it with some cards can treat those past purchases as a sunk cost. Therefore, whenever the customer desires some new IT functionality, the customer has an incentive to purchase that functionality in the form of another Cisco card as opposed to buying an entirely new piece of hardware from another vendor. Purchasing that functionality in the form of another Cisco card as opposed to a separate piece of hardware is advantageous because doing so doesn’t take up extra rack space, takes up less incremental power, and is easier to manage. By therefore increasing the temptation to stay with previously purchased Cisco chassis, Cisco perpetuates the presence of its hardware inside a customer’s datacenter. A Gartner analyst once succinctly said:

“The Catalyst 6500 is the best example of its strategy of selling large chassis that never have to be swapped out. Cisco will make continuous upgrades with new features, but they never want it to open up to a platform discussion.”¹⁶⁹

However, such a tight grip on the hardware platform means that Cisco is deliberately neglecting levers 2-4. Cisco might be better served by opening the platform to third-party vendors, who could then sell adjunct cards. This might be especially promising if Cisco were to license such opportunities to startups who could sell cards that offer a variety of extra functionality that Cisco does not currently offer, and the most successful of these startups could then become targets of acquisition that would be easily integrated with Cisco as they would already be selling technology that inter-fits with the Cisco chassis. In fact, Cisco could treat this as a simple extension of its current M&A philosophy. Cisco would be able to maintain control of the commanding heights by keeping control of the chassis and the Supervisor engine itself, but create a space where startups could innovate and create more value for the Cisco hardware platform ecosystem as a whole. Such a move would also mean that Cisco would be utilizing levers 5 and 6 because it would make Cisco hardware even more versatile and useful to customers, and

¹⁶⁷ Cisco Systems. “Cisco Catalyst 6500 and 6500E Series Switch Data Sheet”. April 2, 2006. http://www.cisco.com/en/US/products/hw/switches/ps708/products_data_sheet09186a00800ff916.html

¹⁶⁸ Cisco Systems. “Cisco Catalyst 6500 Series Switches”. Feb. 15, 2006.

<http://www.cisco.com/en/US/products/hw/switches/ps708/>

¹⁶⁹ Hochmuth, Phil & Duffy, Jim. “Challenges Dog Cisco as Company’s Role Expands”. Network World. May 26, 2003. http://www.findarticles.com/p/articles/mi_qa3649/is_200305/ai_n9264446

also deter startups from partnering with Cisco's rivals. Given the large installed base of Cisco equipment, a startup that wanted to ramp quickly would be loathe to turn down the opportunity to build a Cisco-compatible card. Hence, Cisco since today spends a great deal of R&D money on developing new cards, it can treat the opening of its hardware platform as a way to outsource some of that R&D.

6.3 The Cisco Software Platform

Cisco software, which comprises a wide variety of technologies such as not only the well-known IOS, but also CatOS (for Catalyst switches), Callmanager, Unity, PixOS (for firewalls), has often been described as a semi-open platform, in the sense that while certain aspects of it is relatively open and interoperable, other components are quite closed and proprietary¹⁷⁰. The Cisco software platform is undeniably more open than is the hardware platform. Cisco increases the reach of its software by allowing for some interoperability, while also maintain control by keeping a number of technical information about its software proprietary. Cisco's choices therefore exhibit a unique combination of choices of the strategic levers that it is using.

We shall begin by describing the various ways in which Cisco software could be described open. First off, Cisco's complete software technical documentation is available online, including bug reports, patch versions, software roadmap, and even examples of interoperation with competing vendor's hardware. Cisco also encourages a large textbook industry that publishes numerous examples of working configurations, operational guidelines, and teaching tools, and Cisco even licenses the "Ciscopress" textbook brand-name series to Pearson Education to encourage the dissemination of advanced information regarding how to configure Cisco software¹⁷¹. This extends not only to basic information about how to configure standard Cisco software, which one might expect as a given, but also extends to the publication of the meanings of various error codes and how to operate within 'maintenance mode' or 'bootROM' mode, which is a special mode somewhat akin to the PC's BIOS mode that Cisco routers and switches run in when a serious mechanical problem has been encountered. This stands in stark contrast to other vendors who prefer to cloak the meaning of their error messages and will allow only licensed technicians to learn to use their equipments' maintenance mode. Cisco has even built in a Tool Command Language (Tcl) scripting hook into its software and has allowed for its publication.¹⁷² Cisco has therefore been unusually forthcoming in allowing the public to learn how to use Cisco software.

Cisco software also adheres to most of the commonly used networking standards which, by definition, provide for a certain baseline of vendor interoperability. For the most part, Cisco implements most of the major routing, switching and LAN technologies in a standards-compliant manner. For example, Cisco switches are complaint with the industry-standard 802.1d and 802.1w Spanning Tree protocol that allows for a switched

¹⁷⁰ Cusumano & Gawer. "Platform Leadership". P. 176-177.

¹⁷¹ Cisco Press. "About Us". 2006. <http://www.ciscopress.com/about/index.asp>

¹⁷² Welcher, Peter. "TCL'ing your Cisco Router". Chesapeake NetCraftsman. Jan 3, 2005. <http://www.netcraftsmen.net/welcher/papers/iostcl01.html>

network to find faults in the switched topology and rebuild themselves dynamically, and the 802.3ad Link Aggregation Control Protocol that allows for multiple switch links to be grouped together to act as one high-bandwidth link. Cisco routers implement the most common IP routing protocols such as RIP, OSPF, ISIS, and BGP in an RFC-compliant fashion, and also implement a host of other non-IP routing protocols - such as IPX, Appletalk, DECnet, and Vines – according to the standards for those protocols. Cisco IP phones and CallManager do provide for standards-compliant H.323 and SIP VoIP operation. Hence, you can purchase a mixed-and-matched set of network gear from Cisco and other vendors and build an interoperable network out of it.

Cisco software, to its credit and in great contrast to the competitors which we discussed previously, is completely agnostic to networking protocol suite. Cisco did not push for IP or IPX or SNA or Appletalk to win. Cisco supported all of them and more and let the market decide. Whichever protocol suites you were running on your network, Cisco gear could understand it. Hence, when the market converged upon IP, Cisco was not wedded to another proprietary suite and so could develop its products to aid that convergence. In fact, not only did Cisco ride along with the convergence wave, but Cisco used convergence to kill competitive network platforms like IBM's SNA platform as was discussed previously.

Finally, Cisco licensed its software code extensively, most notably during the 1997 Network Interoperability Alliance (NIA), in which Cisco IOS specifically was licensed to a wide swath of players in the IT industry including HP, Microsoft, Compaq, Intel, and DEC.¹⁷³ While licensees were prohibited from changing IOS code, licensees were provided with complete information on the workings of IOS and could optimize their products to work with IOS.¹⁷⁴ The NIA served to further popularize IOS within the market and cemented the status of Cisco software as the de-facto software platform that governed the networking industry.

However, Cisco's software is distinctly un-interoperable in one key respect. Consider Microsoft's infamous "embrace-and-extend" software strategy in which Microsoft purports to adopt certain industry standards and in fact does so, but adds certain proprietary features to the implementation of that standard in order to 'superset' the standard. Microsoft has used this tactic, for example, in its Active Directory technology, which does adhere to industry-standard LDAP and DNS technology, but also includes Microsoft-proprietary add-ons such that Active Directory can serve other standards-compliant LDAP and DNS clients, but Microsoft clients that are expecting to be serviced by an Active Directory server cannot be serviced by a standard DNS or LDAP server. Cisco has its own twist to this strategy, which could be described as an "extend-then-embrace" strategy. Cisco will promote the use of proprietary software interface for a particular technology, and then later may choose to also adopt the industry-standard mechanism, but will usually still have its gear's default settings correspond to that of the proprietary interface.

¹⁷³ Bunnell, p. 119.

¹⁷⁴ Cusumano & Gawer, , p. 178.

For example, consider the problem of how IP clients on a LAN can know where their nearest router is. Unlike IPX, Appletalk, and DECnet clients, IP clients have no easy dynamic way to determine where the nearest IP router is and so have to be provided with the IP address of that router. The question then becomes what if that particular router were to go down, or the network link to that router were to go down, then all the clients on the LAN would not be able to access the rest of the network. Because of this, one would have 2 routers to serve a particular LAN, with one backing up the other. The problem is that that backup router would have an IP address of its own, and so while that backup router would be ready to service clients were the primary router to fall down, the question then becomes how are the clients supposed to know that the primary router is down so that they are supposed to use the backup router.

Cisco solved this problem through the use of the Hot Swap Router Protocol (HSRP), a software interface in which 2 routers on a LAN emulate a single router. Hence, while both routers have their own individual IP addresses of the LAN, they also have a 3rd “virtual” IP address that they both share. Furthermore, the two routers send heartbeat messages to each other to verify that the other is up. If both routers hear each other’s heartbeats, then the primary router knows that it is the router that “owns” the virtual IP address. However, if the primary heartbeat were to fail, then the backup router would take over control of the virtual address until such time as the primary router was to return to good health. HSRP was hence an example of a ‘redundancy protocol’.

In response to the utility of HSRP, in 1998, the IETF standardized its own redundancy protocol - dubbed Virtual Router Redundancy Protocol (VRRP) as documented in RFC 2338 - that operates in largely the same fashion as HSRP.¹⁷⁵ However, it was only in 2003 with the launch of IOS version 12.3 – 5 years after VRRP had been standardized – that Cisco finally integrated VRRP support into its mainline software.¹⁷⁶ In fact, for a number of years, the Cisco product line exhibited the interesting conundrum that its niche VPN 3000 concentrator hardware supported VRRP, but not HSRP, whereas the rest of Cisco’s product line supported HSRP but not VRRP. This was because the VPN 3000 line came from the acquisition of Altiga, and Altiga obviously could not incorporate HSRP technology while it was still an independent company, and so instead chose to support VRRP¹⁷⁷. It took awhile after the Altiga acquisition before the incompatibilities in redundancy protocols could be resolved. Hence, Cisco’s insistence on running only its proprietary redundancy protocol resulted in more customer confusion than necessary.

Numerous other examples can be found of Cisco’s “extend, then embrace” strategy. Cisco has developed 2 of its own proprietary routing protocols - IGRP (or Interior Gateway Routing Protocol), and EIGRP (or Enhanced Interior Gateway Routing Protocol) and so far has refused to license it separately from IOS or to submit its inner-

¹⁷⁵ RFC 2338. <http://www.faqs.org/rfcs/rfc2338.html>

¹⁷⁶ Cisco Systems. “Cisco IOS Software Release 12.3 New Features and Platforms”. 2006. http://www.cisco.com/en/US/products/sw/iosswrel/ps5187/prod_bulletin09186a0080199900.html

¹⁷⁷ Network World. “Response to RFP: Altiga”. May 10, 1999. <http://www.networkworld.com/reviews/0510rfpaltiga.html>

workings to a standards body¹⁷⁸. Cisco supported its own version of switch-link bundling, Etherchannel, for years before the 802.3ad Link Aggregation Control Protocol standard was published, and still uses Etherchannel as the default method of bundling multiple switch links. Callmanager and Cisco IP Phones can now speak the standardized H.323 and SIP voice-control protocols, but prefer to speak the Selsius-developed proprietary SCCP protocol to each other. The list of examples of Cisco pushing its own proprietary methods and then only reluctantly and belatedly integrating standardized methods goes on and on.

Cisco could be said to be clearly pushing lever 1 in that it is positioning its software to be the de-facto operating system of the network. It provides a set of relatively open interfaces, including a scripting tool, as well as licensing schemes to allow other vendors to interoperate with Cisco software. Cisco's software is built to be agnostic with regard to networking protocol suite and so can be used as the infrastructure for virtually any kind of network. Cisco publishes a great deal of information regarding its software, including information that most vendors would prefer to keep secret.

By the same token, Cisco could be said to be manipulating levers 2 and 3. Cisco positioned its software suite, especially IOS, as the network operating system of choice that would serve as the baseline infrastructure that all networking functionality could be built on top of. Cisco's software functionality is essentially open and available for scripting access and licensing, in direct contrast to the highly closed nature of SNA or IPX. Cisco engages in a wide range of partnerships that are largely collaborative and serve to enhance the value of the partnered solutions. However, Cisco software cannot be changed by any vendor except Cisco itself. Furthermore, Cisco reserves the right to develop a number of proprietary extension features that strengthens the notion that Cisco gear really does work best with other Cisco gear.

However, from a lever 4 position, Cisco could be said to be in a rather weak position. While Cisco tries to position itself as a software company, Cisco sells relatively few standalone software products. Only the Ciscoworks and other related management software could be considered a truly standalone software offering¹⁷⁹. Other software products like IOS and CatOS are packaged within Cisco routers and switches and even a purportedly standalone piece of software like the Cisco Callmanager IP-PBX package is useful only when you have Cisco IP Phones. Cisco currently does not offer a retail version of IOS or CatOS such that customers could 'roll their own' router or switch and in fact, no version of IOS or CatOS exists that can compile on any x86 architecture. Yet the truth is, much of Cisco hardware, particularly the low-end gear, is not particularly powerful. Renegade versions of the Cisco PIX firewall, dubbed the "Frankenpix", have been built by networking enthusiasts on cheap commodity x86 hardware, and have been shown to run just as fast as Cisco PIX firewalls and routers do¹⁸⁰. Cisco's ambivalence

¹⁷⁸ Malhotra, Ravi. "IP Routing – Chapter 4, EIGRP". O'Reilly Publishing. Jan. 2002.
<http://www.oreilly.com/catalog/iprouting/chapter/ch04.html>

¹⁷⁹ Cisco Systems. "Network Management Product Catalog". 2006.
<http://www.cisco.com/en/US/products/sw/netmgts/index.html>

¹⁸⁰ Packet Attack. "FrankenPix 101". Jan 5, 2006. <http://www.packetattack.com/frankenpix.html>

regarding becoming a true software company and reorganizing itself accordingly may mean that Cisco may be making an Apple-style mistake in wedding its software and hardware unnecessarily. Network administrators have often stated that they are unimpressed with Cisco's hardware and are really buying Cisco networking gear for the software which is both feature-rich and familiar. While Cisco has reorganized itself internally a number of times, by product line, by geography, and by customer base, it has never attempted to reorganize itself to make software a true standalone division.

However, that is not to say that Cisco should attempt to remake itself into a pure software company, at least not in the near future. It is clearly true that Cisco's tight coupling of its software to its hardware has resulted in a fantastically profitable business that competitors find exceedingly difficult to replicate. What Cisco ought to do is to centralize its software development in order to not only rationalize the costs associated with supporting different IOS packages that support different pieces of equipment, but also to remove some of the inconsistencies associated with configuring the various kinds of software. For example, it is the opinion of this author that there are few compelling reasons why the Cisco CSR service-provider router division has to develop and maintain its own IOS build (the so-called "IOS-XR" train of software) and the other Cisco router divisions have to develop and maintain their own builds of IOS. Surely Cisco could make one master build of IOS and simply remove or disable features from this master build to fit the fit the less capable routers, similar to how Microsoft makes one master build of Windows for its Server version of Windows, and then removes or disables features of that version to create the Client version of Windows. Hence, Microsoft is basically incurring the development costs once to create two products. Cisco can do the same.

This would also provide the additional benefit of standardizing configurations. There is little reason for an access-list (a feature to block certain IP addresses from transmitting) to be configured differently depending on which Cisco product you are using. For example, if you wanted to block the IP address 1.2.3.4 from transmitting, you shouldn't have to remember that doing so requires that the access-list on a normal router be configured with the numerical designation of "1.2.3.4 0.0.0.0" (where, on a router, the 0.0.0.0 means that you want to block only the 1.2.3.4 address), but on a Pix firewall, you configure such an access-list with the designation of "1.2.3.4 255.255.255.255" (where, on a Pix, the 255.255.255.255 denotes that you want to block only the 1.2.3.4 address). In fact, if you mistakenly use the 0.0.0.0 designation on the Pix, then instead of blocking only 1 address, you end up blocking every address. There is no need for this inconsistency to exist. If the whole idea of Cisco's software strategy is to provide a consistent user interface for network administrators, then Cisco should try to make sure that its software is in fact consistent. One should not have to remember obscure and arbitrary rules regarding how certain software is configured on certain devices is completely different from how it is to be configured on other devices. It is simply a waste of brain cells to be forced to remember that one particular upgrade of IOS (and thus certain features and certain bug-fixes) may be available for one series of router, but not available on another series of routers, such that a particular bug might affect your edge routers but not affect your backbone routers.

A standalone Cisco software division could then explore the possibility of selling IOS to third-party vendors, as well as creating x/86-compatible versions of its software. True, Cisco would have to tread carefully to ensure that Cisco does not turn itself into a pure software company only to see its profits get eviscerated by the open-source movement, in the way that many enterprise software companies have been deeply wounded by competing open-source products. Hence, the move to x/86 is certainly promising, and has a strong precedent – namely Juniper. Juniper’s JunOS software is fully compatible with Intel x86 microprocessors. In fact, the highly technically impressive Juniper T-series service-provider router basically have a PC embedded inside them that consists of a Intel Pentium III or Pentium IV processor, 2GB of DRAM, and a 30GB hard-drive¹⁸¹. Juniper uses this ‘embedded PC’ to provide the user-interface and management control of its routers. It is that embedded PC that provides the means to configure a JunOS as well as a mechanism to store and process logs, save error messages, and perform other management tasks. Nearly all of the heavy lifting on all routers, either Juniper’s or Cisco’s, is performed either in ASICS or in the network-processors. Hence, Cisco might also benefit from the opportunity to migrate IOS to x86 architecture and therefore take advantage of the economies of scale and rapid technological advancement of the PC world, yet avoid the destruction of pricing power that is also inherent to the PC world by maintaining sufficient hardware differentiation (via the use of ASICS and network processors) and by improving the popularity of its software. For example, one could compare the situation to that of Apple today in adopting x86 chips yet maintaining Mac hardware and software differentiation to maintain profit margins.

Cisco did display great savvy in utilizing lever 5 regarding its software. Cisco software, and IOS especially, has become the networking software suite of choice amongst networking managers, so much so that other networking vendors such as Foundry and Enterasys have taken to copying the IOS command-line interface to that network administrators can learn to use their gear quickly,^{182 183} and the Chinese vendor Huawei was accused of actually stealing IOS code, including complete (and renegade) implementations of IGRP and EIGRP.¹⁸⁴ Cisco has licensed its software far and wide and used it to popularize the reach of its products. Theoretically, anybody who knows how to configure IOS knows how to configure the entire gamut of Cisco products. While this claim does not stand up to scrutiny, particularly when it comes to Cisco’s optical (Cerent)¹⁸⁵ and ATM (Stratacom)¹⁸⁶ gear, and is only partially true when it comes to other pieces of hardware like the PIX firewall which are configured almost, but not quite the same as IOS is, it is still largely true that much of Cisco’s software configurations

¹⁸¹ Juniper Networks. “Juniper T-Series”. May 10, 2006. <http://www.juniper.net/products/tseries/>

¹⁸² Rash, Wayne & Rist, Oliver. “10Gig Slugfest”. InfoWorld. April 25, 2003.

http://www.infoworld.com/article/03/04/25/17FEtengig_4.html

¹⁸³ Newman, David. “Enterasys 10G Switch Goes its Own Way – Fast”. Network World. Nov 19, 2001. <http://www.networkworld.com/reviews/2001/1119rev.html>

¹⁸⁴ Hochmuth, Phil. “Cisco Drops Suit Against Huawei”. Network World. July 28, 2004.

<http://www.networkworld.com/edge/news/2004/0728huawei.html>

¹⁸⁵ Cisco Systems “Cisco ONS 15400 Series Command References”. May 1, 2006.

http://www.cisco.com/en/US/products/hw/optical/ps2006/prod_command_reference_list.html

¹⁸⁶ Cisco Systems. “Cisco MGX 8900 Series Switches Command References”. May 1, 2006.

http://www.cisco.com/en/US/products/hw/switches/ps3873/prod_command_reference_list.html

have numerous similarities such that once you know how to configure one piece of Cisco software, it is easy to learn how to configure others. In this way, Cisco is able to take advantage of somewhat viral marketing in the sense that once you buy one piece of Cisco hardware, you are probably going to buy more.

Finally, Cisco has used its software to push lever 6 several times, most ruthlessly with the drawing and quartering of the IBM SNA platform. It was in software that Cisco implemented the multiprotocol-integration and especially IP-integration that ultimately doomed SNA to irrelevance. It was within software that Cisco implemented 'better' SNA technology like DLSw+ that marginalized IBM's networking hardware. Cisco won the war with IBM not through superior hardware, but through superior software. Similarly, Cisco leveraged its feature-rich software to marginalize Apple, Novell and DEC by providing a method within software to encapsulate and thus superset Appletalk, IPX and DECnet. EIGRP was built to not only handle IP, but also IPX and Appletalk. Cisco, through the use of RFC-compliant General Routing Encapsulation software 'tunnels', provided a way to transport any networking protocol over IP.¹⁸⁷ Cisco has therefore used its software as the sharp end of the stick to eliminate competition and is still to this day a monumentally powerful competitive weapon. While other vendors may sell better hardware and arguably sells software that is better at certain features, no vendor offers the wide breadth of features and compatibility that Cisco does.

Hence, Cisco's software strategy demonstrates great acumen in utilizing the 6 networking platform levers. Cisco's software is somewhat open in that it is interoperable with most of the important networking standards, it is scriptable with the open TCL scripting language, and it is widely licensed. However, Cisco maintains control of the commanding heights by being the only vendor to be able to change its software and maintaining a number of proprietary features within its software in an 'extend-and-embrace' strategy that optimizes the performance of a Cisco-only network. Cisco has used its software to enhance the ubiquity of its networking products by reinforcing the preference of the 'look-and-feel' of its software command-line interface such that other vendors have had to resort to copying the user interface to reduce training costs. Finally, Cisco has bundled features in its software to subsume the capabilities of its competitor's gear and marginalize it. However, Cisco has steadfastly refused to consider running software as a separate division within Cisco, either as an internally staffed division that would serve Cisco's various hardware divisions, or potentially (in the long-term future) as a division that might sell Cisco software, especially IOS, to 3rd-party hardware vendors. Nor has Cisco ever deigned to sell an x-86 compatible version of IOS that would ultimately allow Cisco to source processors from a supplier market that exhibits keen competition and rapid technological progress. If Cisco is truly the software company that it purports to be, then Cisco may want to consider such an internal reorganization. In particular, Cisco might be making the mistake that companies like DEC, Apple, and IBM made in becoming too wedded to their hardware packages rather than understanding that much of their competitive advantage was derived from software and that they should therefore be working to build their software capabilities. Much of Cisco's low-end gear is technically unimpressive and is purchased by customers just to

¹⁸⁷RFC 2784. <http://www.faqs.org/rfcs/rfc2784.html>

get access to IOS. Hence, it is the software that most customers really care about. Cisco ought to ensure that if customers are really buying Cisco gear for the software, that Cisco manages its software development correctly to ensure consistent configuration and consistent upgradeability.

One might wonder why Cisco has not come under greater pressure to open-source its software the way that operating systems vendors have clearly been subject to, and of which application software vendors have also felt the sting. Truth be told, Cisco has indeed felt competitive pressure from the open-source community for the general software applications that it sells. For example, Cisco's Callmanager has been targeted by the open-source Asterisk IP PBX project, and several technical websites have been published detailing how to integrate Asterisk and Callmanager PBX's together, as well as how to emulate Callmanager features with Asterisk.¹⁸⁸ The Ciscoworks management software is challenged by open-source management applications such as MRTG and Nagios/NetSaint. IOS is emulated by open-source applications such as the GNU Zebra project, the XORP project, and the Linux Router Project that purport to be able to turn any x86 system into a router that will support many of the popular routing protocols and features while presenting an IOS-inspired user interface.^{189 190} So it is clearly true that Cisco has had to deal with pressure from the open-source community.

However, it is true that Cisco has not felt the sting of the community as much as other vendors such as Sun and Microsoft. This is most likely due to the fact that Cisco's routers and switches are not fully-fledged computing systems and have therefore proven to be less interesting to the underground hacker community that has chosen to direct its energy at developing alternative OS's and applications. It is also true that, unlike Microsoft, Cisco does build its equipment to work with standardized networking protocols. While adherence to networking standards is a far cry from being open-source, it has deflected much of the ire of the open-source community away from Cisco. While you can't see what is happening inside a Cisco router or switch, you can interoperate with it in the way that the standards say you should be able to. Hence, there is less of a desire to reinvent the wheel. Furthermore, Cisco maintains an active relationship with the Linux community, funding a rather active Sourceforge site¹⁹¹ that provides a range of utilities that either emulate or enhance the functionality of¹⁹² a Cisco network¹⁹³, and itself running Linux within its acquired Linksys line of consumer routers.

However, this might just be the calm before the storm. Vyatta, a startup company, announced in 2004 that it will attempt to dethrone Cisco by selling open-source routers

¹⁸⁸ VoIP-info.org. "Asterisk Call-Manager Integration". May 27, 2006. <http://www.voip-info.org/wiki-Asterisk+Cisco+CallManager+Integration>

¹⁸⁹ Zebra Project homepage. May 1, 2006. <http://www.zebra.org/>

¹⁹⁰ Linux Router homepage. May 1, 2006. <http://www.linuxrouter.org/>

¹⁹¹ Malik. Om. "The Black Box That Would Conquer Telecom". Business 2.0. Feb. 22, 2006. http://money.cnn.com/magazines/business2/business2_archive/2006/03/01/8370567/index.htm

¹⁹² "Ewing, James. Linux on Linksys Wi-Fi Routers". Linux Journal. Aug 1, 2004. <http://www.linuxjournal.com/article/7322>

¹⁹³ Cisco-Centric Open Source Exchange Community homepage. May 5, 2006. <http://cosi-nms.sourceforge.net/>

based on the XORP open-source routing project. Vyatta is funded by Intel and Microsoft, as they both see Vyatta as a method to strategically weaken Cisco and (in the case of Intel) to sell more commodity x86 hardware¹⁹⁴. Vyatta is scheduled to launch its first official non-beta product in the summer of 2006. Such a company could present tremendous problems for Cisco as corporate IT administrators have used open-source operating systems and applications to reduce systems costs, and now realize that they spend significant sums on their proprietary network technologies. Even if Vyatta fails, more startups are undoubtedly going to be created to claim their stake of the networking cash cow. Hence, Cisco has to prepare for this threat and to move in front of it by open-sourcing its software if necessary. How Cisco deals with the open-source community will remain one of the most intriguing challenges in the networking world in the next decade.

6.4 Cisco Business Tactics

Cisco has not relied solely on technology, either hardware or software, to effect its stranglehold on the market. Not by a long shot. Cisco has also used a number of business tactics to obtain and maintain its grip and cement its status as the leading networking platform vendor and defeat all comers. Cisco's business tactics serve to holistically reinforce Cisco's technological advantages and blunt whatever tactics by competitors to obtain a beachhead in the market.

First off, Cisco has managed to wrap a highly positive marketing message around its products. Through the Cisco-Powered Networks marketing promotion, Cisco encouraged service-providers to build networks that were majority Cisco. A provider that adhered to the Cisco-Powered Network requirements was eligible for joint cross-marketing expenditures and promotional partnerships with Cisco, similar to the "Intel Inside" marketing promotion that was used so successfully by Intel with PC vendors. Cisco also pushed a "halo effect" about its networking gear by catering to the notion that its gear is uniquely enterprise-worthy and hence the safe choice. "They are like the early 1960s IBM ... [in that] nobody ever got fired for buying Cisco."¹⁹⁵ Cisco has promoted a mystique about itself such while its gear adheres to the major networking standards and is thus compatible with other vendors, only it holds the true keys to the kingdom. Only Cisco can provide the promised increases in productivity and connectivity that better network spending promises, or so Cisco marketing would have you believe. Cisco even pushed the alluring phrase: "Are you ready?" to increase the anxiety of customers to buy Cisco gear and hence be ready for the whirlwind of technological changes that were purportedly about to befall them.

¹⁹⁴ Salkever, Alex. "Does XORP Have Cisco's Number". Businessweek. Nov 29, 2004.
http://www.businessweek.com/technology/content/nov2004/tc20041129_5206_tc024.htm

¹⁹⁵ Strom, David. "Is Cisco Vulnerable?". VARBusiness. Jun 2, 2003.
http://www.varbusiness.com/sections/news/dailyarchives.jhtml;jsessionid=RKWSK3NGOYWK0QSNDBCSKHSCJUMKJVN?articleId=18830254&_requestid=29855

Cisco used its own business success as an equally important tool in its marketing toolkit. Cisco was the fastest company in history to ever reach \$100 billion, \$200 billion, and \$300 billion of market cap.¹⁹⁶ “At one time, Cisco beat Wall Street’s expectations for 43 straight quarters”¹⁹⁷. Cisco’s advantage was purportedly to its technological acumen – it ran much of its customer service and sales over the Internet and automated much of its operations via advanced information technology. Cisco’s possessed the vaunted ability to perform the ‘virtual close’ through the use of its own home-cooking – large IT infrastructures supported by a robust Cisco network. While Cisco did finally miss earnings in 2001 and while its vaunted information technology capabilities proved to be far less substantial than they seemed, Cisco did enjoy one of the greatest runs in business history. During all that time, Cisco was able to point to its own business success as a reason for customers to invest in IT infrastructure in general and in Cisco networks in particular. By the time Cisco ran into problems, the Cisco had already built a gigantic installed base of customers – many of who had also turned on Cisco-proprietary features – and thus had little choice but to continue to maintain spending on Cisco gear. Moreover, while the benefits may have been overhyped, there were benefits to building a robust IT infrastructure, because doing so did increase productivity. Customers just had to have a reason to invest in that infrastructure in the first place, and during the late 90’s, they were given an excellent reason to do so. They wanted to be like Cisco.

Cisco also developed a large pool of IT technicians and engineers who were trained in Cisco products and therefore had a strong interest in pushing Cisco gear. Cisco developed the Cisco Certified Internetworking Expert (the CCIE) program, which was modeled on the Novell CNE program but had the additional benefit that candidates had to take a hands-on lab exam to prove their knowledge. This proved to be a critical difference between the CCIE program and other IT certification programs like the Novell CNE or the Microsoft MCSE program. It eliminated the problem of “paper-engineers” – those who could pass tests but could not actually perform tasks in the real world, a problem that has plagued the computer certification industry to this day. The CCIE quickly took its place as arguably the most respected and most desired IT certification in the world.¹⁹⁸ Candidates would often times spend up to 5 figures of their own money to purchase lab equipment and books to prepare for the CCIE exam, and might take up to 3 or 4 attempts before passing.¹⁹⁹ Cisco later fleshed out the CCIE certification with lower-level certifications such as the Cisco Certified Networking Associate (CCNA) and Cisco Certified Networking Professional (CCNP) certificates, as well as a host of design and technology-specific certifications. While these certifications suffer from the paper-engineer problem, they are often seen not as endpoints in themselves, but rather as stepping stones to the CCIE. The CCIE is viewed as the real proof of the pudding.

¹⁹⁶ MsFiscallyFit. “Cisco Systems – the Cisco Kid Grows Up”. MsFiscallyFit.com, May 7, 2000. http://www.ms fiscallyfit.com/investing/high_impact/hot_stocks_6/hot_stocks_6.html

¹⁹⁷ Handle name: edelfenbein. “Will Cisco Pay a Dividend?”. Aug 11, 2005. Crossing Wall Street. http://www.crossingwallstreet.com/archives/2005/08/will_cisco_pay.html

¹⁹⁸ Granville, Fred. “The Road to CCIE – Part Three”. Independent Thoughts. April 27, 2006. <http://www.flgnetworking.com/article4.html>

¹⁹⁹ Morris, Scott. “So You Wanna be a CCIE? Part 1”. CertCities. Jun 28, 2005. <http://certcities.com/editorial/features/story.asp?EditorialsID=89>

Furthermore, Cisco has fostered training and knowledge of its products, but explicitly and otherwise. Cisco has founded the Cisco Networking Academy, a program designed to teach high school and college students the basics of configuring and maintaining a Cisco network. Cisco has partnered with states and the Federal government to offer Cisco training courses as options for workers taking advantage of government retraining dollars²⁰⁰. Cisco has also worked with various colleges to shape the Cisco training curriculum in such a way to provide college credits for people taking college courses that primarily teach Cisco networking, such as Cisco Networking Academy courses.²⁰¹ Furthermore, Cisco has deliberately turned a blind eye away from the burgeoning trade of used Cisco gear on Ebay and other Internet commerce sites, despite the fact that much of that gear is used for self-study purposes (especially by those people prepping for their CCIE exams) and is thus never relicensed with Cisco. Strictly speaking, the Cisco software that accompanies any of Cisco's hardware is non-transferable and thus software must be repurchased by anybody who buys used gear²⁰². Furthermore, Cisco has done little to deter the cottage industry of training companies and study sites whose purpose is to sell training and education to help you pass the various Cisco certification exams, especially the CCIE. Cisco probably knows that it indirectly benefits from unlicensed gear being used for self-study and companies who sell easier pathways to the CCIE because they ultimately create a larger pool of Cisco-trained personnel. More Cisco-trained personnel mean more technicians and engineers that will push their employers to purchase Cisco because it is the only gear that they know how to use. This is not dissimilar to the practice of Microsoft for many years of turning a blind eye towards mass pirating of Windows and Office because Microsoft knows that even a pirate copy of Windows/Office means another user is becoming accustomed to the look and feel of Microsoft software instead of learning MacOS or Linux.

Finally, as discussed previously, Cisco implemented numerous initiatives to foster a large ecosystem of systems integrators and resellers. Cisco made it relatively easy to become a licensed Cisco partner, with relatively few requirements. To become an official Cisco Premier Partner, all you needed was to have some of your employees become Cisco-certified, and to maintain a minimum level of customer-service.²⁰³ Higher levels of partnership, such as Silver and Gold partnership, had more stringent requirements, including the employment of CCIE's.²⁰⁴ However, the requirements, especially for the baseline Premier partnership, were still quite reasonable to achieve for many small business owners. Partnership provided a number of benefits, not least of which was access to Cisco marketing and promotional campaigns and, more importantly, access to discounts on Cisco gear upon which you could create a reseller business upon which to

²⁰⁰ Cisco Systems Network Academy "About" homepage. May 1, 2006.

<http://www.cisco.com/web/learning/netacad/academy/About.html>

²⁰¹ ECPI College of Technology Description of Programs. April 30, 2006.

<http://www.ecpi.edu/military/programs/index.cfm?pgNum=16>

²⁰² Broadband Reports. "Cisco Forum FAQ – Smartnet 20.1". Broadband Reports.com, March 16, 2006.

<http://www.dslreports.com/faq/cisco/20.1+Smartnet#13460>

²⁰³ Cisco Systems. "Premier Partner Requirements". 2006.

http://www.cisco.com/web/partners/pr11/pr8/pr51/partners_pgm_requirement_summary.html

²⁰⁴ Cisco Systems. "Gold Partner Requirements". 2006.

http://www.cisco.com/web/partners/pr11/pr8/pr27/partners_pgm_requirements.html

sell professional services such as systems integration and consulting work. Cisco therefore not only encouraged the development of a large number of systems integrators/resellers, but also intertwined them with the education and certification process. All of this served to popularize the Cisco certification tracks even more than usual.

Incidentally, one of Cisco's greatest boons to both the education and the integration industry is what Cisco is not doing. The entire Cisco education and integration industries depend on one thing – that Cisco gear remains difficult to configure. Cisco gear is famous for not only providing a wide array of features, but also for being unusually difficult to learn how to use. Mac's are complicated beasts under the hood too, but Apple goes out of its way to make them easy to use. MacOS X in particular, is a full-blown version of BSD UNIX and therefore retains all of the power of that venerable OS, but that complexity is shielded by the Aqua user interface which is unusually easy to learn. Surely Cisco, with a far larger R&D budget than Apple has, could develop an easy-to-use user interface to configure and maintain its gear. Yet Cisco does not seem to want to do that, as it would surely infuriate the systems integrators and training companies who owe their whole existence to the fact that Cisco gear is so difficult to use.

Cisco's business strategy therefore displays a masterful use of the platform levers. Cisco defined itself as a key information technology vendor that holds the secrets to greater productivity and connectivity that all other companies ought to emulate. Thus, Cisco's scope was to become one of the key "4 horsemen" of the Internet – along with Sun, Oracle, and EMC – where each of the horsemen was king of its particular IT domain²⁰⁵. Cisco would rule networks, Sun would rule servers, Oracle would rule databases, and EMC would rule storage. To be a hip and relevant company was to purchase IT equipment from each of these vendors, or so the marketing would have you believe.

Cisco also encouraged complementors to adhere to the Cisco marketing message through use of the Cisco-Powered-Network rubric. The message was that Cisco would provide the gear, the provider would build a network out of that gear and Cisco would then help that provider with marketing and promotion. Cisco also helped training companies, high schools, and community colleges to sell educational services that would increase the number of Cisco-trained personnel who would be hired by companies and would then recommend Cisco as their vendor of choice. Cisco also created a large ecosystem of systems integrators and resellers who worked to push Cisco equipment.

Cisco's rules of engagement were simple. Cisco would build the gear. The service providers would use that gear to build networks and charge customers to transit them. The training companies and schools would sell classes on how to use that gear. The integrators would sell IT engineering services. Cisco would provide a marketing, training, and integration infrastructure to encourage all of these activities, but would otherwise not interfere in the actions of these other markets. Cisco would not become a service provider on its own. Cisco would not run a training/education department of its

²⁰⁵ Ante, Stephen. "Commentary: The Four Horsemen of the New Economy". Businessweek. Oct. 2, 2000. http://www.businessweek.com/2000/00_40/b3701082.htm

own – all training courses were to be handed off to Cisco education partners. Cisco would not develop its own integration/consulting arm. Hence, Cisco's use of levers 2, and 3 reflected Cisco's deliberate choice of choosing to engage this ecosystem of providers, trainers, and integrators in a highly complementary manner and use them as force multipliers. Cisco by itself may not have been able to take over the networking industry. But Cisco and its army of auxiliaries could and did.

Cisco's marketing message and engagement of training companies and systems integrators served to manipulate lever 5. Via the "Are you Ready?" marketing campaign, Cisco basically tempted customers to find out what networking really was all about and elicited interest in building out networks so that customers could be ready for whatever it was that Cisco was implying customers needed to be ready for. Via expanded training, Cisco unleashed hordes of newly educated and certified technicians who wanted to install networks wherever they could in order to prove their worth (and create some job security). Via an empowered array of systems integrators, Cisco removed the (largely self-inflicted) roadblock that prevented networking gear from penetrating further into the market, namely that networking gear was difficult for customers to use. Hence, the demand for all networking gear in general and Cisco gear in particular was stoked.

Finally, Cisco utilized lever 6 in destroying its competitors through its business practices. Why become Novell certified when becoming a CCIE meant more respect and more money? Why jump through all of the hoops necessary to become a Novell or Vines or DEC integrator and reseller when the Cisco partnership requirements were so easy? Why choose to either learn the technology of or partner with a rival networking vendor like Bay or 3Com when Cisco, because of its seemingly perpetually rising stock price, clearly looked like the winning horse? Hence, the market strongly tipped towards Cisco and away from the other vendors. The other vendors, in losing their integrators and training companies, basically lost their oxygen they needed to breathe. Beating Cisco is difficult enough. But beating Cisco when your allies have abandoned you to join the other team is darn nigh impossible.

7. Summary and Final Thoughts

Cisco has proven to be a fantastically successful company that has demonstrated masterful use of the 6 strategic levers available to networking platform vendors. Cisco's dominance is approaching the dominance that IBM displayed during the glory days of SNA. Nevertheless, a few storm clouds can be seen on Cisco's horizons.

A discussion of Cisco's future would not be complete without a discussion of Juniper Networks, arguably the only company that can match Cisco from a network technology standpoint. Juniper's routers have been widely praised for their innovativeness, reliability, and feature set. Lightreading, a news website that is popular amongst many networking enthusiasts, sponsored a bakeoff between Juniper's (at the time) flagship

product, the M160, against the Cisco GSR 12400 in the year 2001²⁰⁶. The M160 emerged victorious, validating what was already known by many of the networking cognoscenti – that Juniper’s routers were simply faster and more reliable than anything that Cisco had to offer. Juniper maintained its technical lead with the launch of its T640, a router that provided for double the bandwidth in half the rack-size, thereby effectively quadrupling the routing density available,²⁰⁷ and furthered the technological pace via its T-X matrix technology which can bind multiple T routers into a single routing core.²⁰⁸ While Cisco has improved its offerings via the CRS-1 Carrier Routing System that Lightreading found can match the T640 from a ‘speeds and feeds’ basis, and even improves on the Juniper T-series router in certain respects²⁰⁹, the damage had already been done. Juniper developed a reputation for being the vendor that truly provided carrier-class reliability and bandwidth and quickly developed a strong cult following within the network engineering population. As of Feb, 2006, Juniper had taken more than 1/3 market share for core service-provider routers, a market that Cisco used to have all to itself.²¹⁰ Whether network engineers should become Cisco Certified Internetworking Engineers (CCIE) or whether they should follow the Juniper clone certification track, the Juniper Networks Certified Internetworking Engineer (JNCIE) track became a legitimate and heated topic of debate.²¹¹

How Juniper rose from a tiny startup to challenge Cisco is a combination of Cisco’s missteps, Juniper’s laser-focused execution on one market (the service-provider market), and the desires of customers to have a second source. First off, while Cisco was consolidating its dominance in enterprise networks during the late 90’s, Cisco lagged in developing its technology for its service-provider customers, and in particular, for the Internet Service Providers that were screaming for bandwidth to keep pace with the meteoric growth in demand for Internet services. In particular, Cisco was late in developing its OC-192 network interface for its core routers, OC-192 being a specification of optical interface that provides for 10Gbps of bandwidth. By the time that Cisco was ready to launch this interface in 2001, Juniper had already been offering such interfaces on its core routers for nearly a year.^{212 213} Not only did this allow Juniper to win sales from those service-providers whose networks were bursting at the seams, but it gave Juniper great technical cachet that Juniper was the company that really had the best

²⁰⁶ Saunders, Stephen. “Juniper Wins Monster Router Test”. Light Reading. March 12, 2001.

http://www.lightreading.com/document.asp?doc_id=4090

²⁰⁷ Juniper Networks. “T-series Product Guide”. 2006. <http://www.juniper.net/products/tseries/>

²⁰⁸ Haley, Colin. “Juniper Routes Core Objective”. Internetnews.com. Dec. 3, 2004.

<http://www.internetnews.com/infra/article.php/3443571>

²⁰⁹ Matsumoto, Craig. “Cisco’s CRS-1 Passes our Test”. Light Reading. Nov. 30, 2004.

http://www.lightreading.com/document.asp?doc_id=63607

²¹⁰ Moritz, Scott. “Cisco, Juniper Find New Foe”. TheStreet.com, Feb. 24, 2006.

http://www.thestreet.com/_googlen/tech/networking/10270135.html?cm_ven=GOOGL&cm_cat=FREE&cm_ite=NA

²¹¹ Morris. “Should I Become a CCIE or a JNCIS/JNCIE?”. TCPMag.com. May 25, 2004.

<http://tcpmag.com/qanda/article.asp?EditorialsID=281>

²¹² Raynovich, R. Scott. “Cisco Ships OC-192”. Light Reading. Jan 31, 2001.

http://www.lightreading.com/document.asp?doc_id=3455

²¹³ Thomas, Tom M. “Juniper Networks Router Architecture”. Addison-Wesley Professional. Jan. 24, 2003. <http://www.informit.com/articles/article.asp?p=30631&rl=1>

engineering. Juniper positioned itself as the company of “real engineers” as opposed to Cisco which was seen by many as dominated by sales and marketing, but whose ability to meet the technical needs of highly demanding providers was suspect.

Juniper furthered its reputation for technical excellence through JunOS, the Juniper counterpart to Cisco IOS, but which had a number of technical features that were highly appealing to highly technical users, including an interface that strongly resembled the format of the popular C programming language and a full-blown BSD UNIX kernel underneath, which mean that network engineers could dive right into the bowels of a Juniper router using familiar UNIX commands. In short, JunOS provided network engineers with the power and flexibility of familiar technologies. It is the opinion of this author that anybody who is familiar with the UNIX syntax, as most computer engineers are, would find JunOS far more intuitive and flexible than is Cisco IOS.

Finally, Juniper was undoubtedly enhanced by the service-providers strong desire for a second source to avoid becoming beholden to Cisco as a monopoly vendor. One of Juniper’s initial investors was Worldcom, which would eventually become MCIWorldcom and would run, through its UUnet subsidiary, the largest Internet service provider in the world.^{214 215} UUnet also became Juniper’s star customer, serving not only as a test bed for Juniper and an important source of revenue, but more importantly, as a reference customer to prove that it could handle the traffic of the largest IP network in the world. Juniper was able to quickly snap up other large customers, including Verio, Genuity, Cable & Wireless, France Telecom, Cox Communications, and many others. Customers appreciated having Juniper around as a means to prod Cisco on price and on features. Even Don Listwin, Executive Vice President of Cisco acknowledged that “Carriers are always looking for a second source on things...”²¹⁶

Another competitor nipping at the heels of Cisco is Avaya, which is the spun-off and revitalized enterprise telecommunications division of Lucent. Avaya’s claim to fame is its laser-like focus on telecommunications needs of enterprises, coupled with more than a century of accumulated expertise in fulfilling those needs through its Lucent ancestry. Not only could Avaya continue to milk the old Lucent corporate PBX cash cow through service and maintenance fees, but Avaya also made a considerable investment into IP-PBX technology to offer a technically impressive IP telephony solution. Not only that, but Avaya could offer something that Cisco never could through its AVVID IP PBX technology – namely a graceful way to transition from the older technology to the new IP technology. Cisco’s AVVID technology essentially demanded a traumatic rip-and-replace. An established company that wanted to install AVVID had two options. The company could throw away its entire old enterprise PBX infrastructure and replace it with Cisco’s new IP-based gear. However, this ran the substantial risk of introducing

²¹⁴ WHNet. “Gigabit IP Packet Switching Routers”. WHNet.com. July 2002.

<http://www.whnet.com/giga.html>

²¹⁵ ISP-Planet Staff. “ISP Backbone Market Forecast – Flat Through 2002”. ISP-Planet.com. Jan 23, 2002. http://www.isp-planet.com/research/2002/backbone_020123.html

²¹⁶ Heskett and Wong, Wylie. “Juniper Finds its Own Networking Niche”. CNET. Aug 16, 1999. http://news.com.com/Juniper+finds+its+own+networking+niche/2100-1033_3-229831.html

integration bugs into your company, which is especially problematic given how mission-critical the phone system is to most companies. Unsurprisingly, few customers opted for this choice. The other option would be to maintain your old enterprise PBX infrastructure everywhere except in one section, which would use the Cisco IP solution. This would allow you to slowly introduce the Cisco solution into your network. The problem with that is that you then had to translate between the border of the old and new phone technologies to ensure that every phone in the company could still contact every other phone in the company. These translations were fraught with technical difficulties and often times resulted in Cisco and the incumbent PBX vendor pointing fingers at each other, a situation exacerbated by the fact that Cisco had minimal track record with phone technology.

Avaya provided a way to avoid these problems. Avaya integrated IP technology right into its existing PBX's. That way, translation problems were minimized, as the translation would happen within the Avaya PBX itself. While this solution obviously availed itself only if the enterprise was running Avaya/Lucent PBX's, the fact is, Lucent had left a large installed base of customers who would clearly find it easier to transition to IP by following the Avaya roadmap than the Cisco roadmap. Even those customers who were not using Avaya/Lucent PBX's often times still found it easier to follow Avaya. Avaya, through its Lucent patrimony, had been building translation technologies that work amongst different PBX's from different vendors for many decades. Hence, what was needed to properly translate to and from a Nortel PBX or a Siemens PBX was far better understood by Avaya than it was by Cisco. Finally, Avaya clearly benefited from the Lucent heritage and the comfortable rapport that many corporate telecom engineers still felt for Lucent. Cisco's reputation was in datacom, and had therefore inherited some of the more disreputable traits of data networks, especially its reputation for unreliability. Say what you will about the public telephone network, but the fact is, it is to this day still far more reliable than the Internet is. You pick up the phone, and you know you will get a dial-tone. It just works.

The Avaya strategy seems to have worked so far. Despite a full-court press from Cisco, Avaya has led Cisco in market share in IP telephony since 2003.^{217 218} Avaya has also partnered with Juniper to offer a turnkey IP telephony and network bundle. Avaya also continues to leverage its deep knowledge of enterprise telecom needs by developing call-center and call-distributor software packages that will embed Avaya telephony technologies deep into its customers' IT infrastructures.

So it would seem that Cisco has serious competitors within the service-provider router and telephony submarkets. However, that statement needs to be put in perspective because those submarkets are just that – submarkets. Even if Cisco were to lose utterly in both the service-provider router and the telephony submarkets, Cisco would still be a fantastically profitable company that still dominates the corporate networking cash cow.

²¹⁷ Burrows, Peter. "Can Cisco Settle for Anything Less than Sizzling?" *Businessweek*. Feb. 21, 2005. http://www.businessweek.com/magazine/content/05_08/b3921093_mz063.htm

²¹⁸ Synergy Research Group. "Enterprise IP Telephony Posts Another Strong Year in 2005". *Marketwire*. Feb. 27, 2006. http://www.marketwire.com/mw/release_html_b1?release_id=111128

While Cisco would certainly lose face if it were to lose the service-provider market to Juniper, for it would therefore be conceding that Juniper does in fact have faster and more reliable networking technology, the truth is, Cisco competes in the corporate networking space on neither speed nor reliability, and never has. Rather, Cisco competes in that space through better marketing, better understanding of corporate needs, and top-notch technical support. Cisco also competes with superior multi-protocol networking technology. Juniper is a pure IP company, and has made no indications that it wants to support IPX, SNA, and any of the other legacy protocols that enterprises still find themselves using. While IP has certainly won in the core of the Internet, and is ultimately destined to win on the Internet's edges, total victory will still take time as enterprises are always loath to eliminate installed bases of infrastructure. Hence, while Cisco would be wounded if Juniper were to win the service-provider, Cisco would be far from dead.

Cisco's only danger of really losing to Juniper would be if Juniper were to successfully enter the enterprise market and take enterprise share from Cisco. Given Cisco's powerful and well-developed corporate sales force and customer lists, coupled with a well-developed ecosystem of resellers and training companies that cater to the enterprise market, this would seem to be highly unlikely. In fact, Juniper is essentially a one-trick pony. Their only reason for existence is that they make a technically superior IP service-provider router than does Cisco. If Cisco can ever make a router that is better than Juniper's, then Juniper will inevitably fall by the wayside. Juniper has little to fall back to, despite acquisitions such as Unisphere and Netscreen to broaden its portfolio. Furthermore, Cisco has the huge corporate networking business to fund such an R&D assault upon Juniper if that's what it wanted to do. Hence, Cisco is clearly far more of a mortal threat to Juniper than vice versa.

The same could be said about Avaya, along with the dynamic that, as John Chambers once said, "Voice will be free". Hence, given the direction that voice metering is going, it is actually rather questionable just how much of a business will continue to exist in selling corporate voice solutions. Cisco and Avaya currently sell their IP telephones for several hundred dollars each. However, consumers have shown increasing comfort with pure software IP voice clients such as Skype and voice-enabled instant messaging software, meaning that the consumer interface in the future may just be a piece of software on their PC, along with a microphone/headphone set. In fact, Cisco and Avaya currently offer "Soft-phones" which are basically emulations of their IP telephony hardware within a software skin. Hence, the telephony hardware business is probably destined to shrink. In fact, it is the opinion of this author that the entire corporate telephony business will accrue to neither Avaya nor to Cisco, but to the company that owns both the client OS itself as well as the corporate directory that serves as the central store of all identity data within a company. In other words, it will accrue to Microsoft. Microsoft offers (among other things) a tremendously powerful corporate identity database through Active Directory. Microsoft already offers voice-enabled Instant Messenger bundled in Windows. Microsoft already offers a technically advanced corporate Instant Messenger Server, dubbed Live Communications Server, that is fully voice enabled. Unified messaging, which is the integration of voicemail and email, can

be offered only through tight integration of telephony with email systems, and Microsoft sells Exchange, the most popular corporate email server application in the world. Hence, many pundits predict that Microsoft will become an increasingly important player in the corporate telephony space²¹⁹. Not only does Microsoft bring a suite of technologies to the table that Cisco will find difficult to match, but Microsoft is one of the few companies that can clearly outmuscle Cisco from a financial standpoint. Hence, Cisco should be far more fearful of Microsoft than of Avaya.

Finally, a word should be said about the threat from open-source. Numerous open-source routing projects such as XORP and Zebra exist, and as mentioned previously, one VC-funded company has already decided to build products based on those projects. More are sure to come. Cisco has been lucky to escape the fury of the open-source evangelists so far, as they have chosen to instead vent their wrath on Sun and Microsoft, but one should not expect this détente to last for much longer. The fact is, corporate networking infrastructure represent a conspicuously large chunk of the budget of an IT department, and is therefore a prime candidate for cost rationalization.

Cisco should probably follow the script laid out by IBM in dealing with the open-source community in engaging them heavily to create technologies while reserving the crown jewels for itself, as well as be cognizant of the six strategic levers available to it. Cisco should first analyze lever one to determine what are the core functions it wishes to keep for itself, and what can be handed to complementors in the open-source community. Corporate routing and switching, as governed by IOS, is Cisco's core strength and other functions are probably extraneous. Cisco could, for example, release certain software applications to the open-source community of which it is not doing particularly well anyway, such as the Ciscoworks management software which is an also-ran to offerings from other vendors such as HP Openview. Cisco should also certainly try to convert more of its regular applications software to work with Linux. Cisco launched a version of Callmanager that is compatible with Linux in March 2006, which is certainly a good first step.²²⁰ Cisco could choose to release all of the current software that runs on servers, but maintain careful control over the right to integrate that software with IOS, especially any software that runs on Cisco routers or switches themselves. Callmanager running on a server will inevitably be a commodity as that market will most likely be swallowed by Microsoft's LCS product. However, Callmanager that is embedded on a router (the so-called "Callmanager Express" upgrade to IOS) adds significant value to the router as it moves the telephone signaling intelligence away from the server space, where Cisco is uncomfortable, to the router, where Cisco is quite comfortable.

Levers two and three could then be utilized to allow for IOS to interact with outside software, but retain control over IOS for now, and then to identify complementors who might create this outside software. For example, if an open source party develops software that enhances the management of IOS. Cisco should allow for such parties to

²¹⁹ Salkever. "Microsoft: Your Next Phone Company?" Businessweek. Mar 2, 2004.
http://www.businessweek.com/technology/content/mar2004/tc2004032_6587_tc076.htm

²²⁰ Keating, Tom. "Cisco Unified Communications Launches". VoIP & Gadget Blog. Mar 5, 2006.
<http://blog.tmcnet.com/blog/tom-keating/voip/cisco-unified-communications-changes-everything.asp>

create such software and profit from it, and perhaps even allow for a method for that software to be loaded directly onto routers themselves as an add-on feature. As described previously, Cisco already provides scripting hooks to IOS through the TCL interface. Cisco could simply codify and standardize these hooks and provide royalty-free licenses for research and developmental purposes to the open-source community to utilize these hooks, as well as expose more hooks and API's.

Cisco should reorganize its software division to maneuver lever four. For reasons stated above, Cisco should do this whether the open-source threat existed or not, simply to standardize and centralize IOS development across its entire product line and eliminate software inconsistencies. However, an additional benefit is that it would allow the Cisco IOS division to speak with one voice to the open-source community as opposed to presenting myriad different hooks and different interfaces, some of them contradictory. It would also allow Cisco to move towards becoming a pure software vendor should it find that necessary to meet the open-source challenge. Such a step is certainly not recommended anytime in the near future, but if Cisco's business does commoditize from competition from open-source routing vendors, then turning into a software business may be what Cisco needs to do eventually. Most customers buy Cisco to get access to IOS, not because they are impressed with Cisco's hardware.

Finally, Cisco can use lever five by engaging the open-source community to develop new and innovative uses for Cisco networks. Cisco would increase the popularity of its routers by allowing open-source community members to hook into the capabilities of IOS to create integrated Linux/Cisco networks that could be dynamically tuned to handle Linux server traffic or that would feature fully automated management of both the servers and the routers. Game enthusiasts who are playing on a Linux system at a LAN party might be able to tweak their Cisco network to provide high levels of responsiveness for the data packets that correspond to the movements of their players in the game. Linux systems might be able to provide specialized call-processing applications to a Cisco IP telephony network. Any or all of these initiatives would serve to increase the value of a Cisco network to end-customers and would therefore increase the total size of the pie.

Finally, Cisco could use lever six to strike back at its competitors. Juniper currently carries an aura of being technically savvy – a high profile association with the open-source community would blunt that advantage. Furthermore, a close association with the open-source community could take away much of the wind from the sails from open-source competitors like Vyatta. Such an association would certainly serve as a final blow to Lucent and Nortel, whose technological development in the routing space has been lagging, and which certainly would have great difficulty in matching the quick pace of development of the open-source community.

Obviously the great danger of getting too close to the open-source community is that somebody will figure out how to come up with a true open-source version of IOS. IOS is an extraordinarily complicated piece of software, and the communities that exist around the XORP or the Zebra project are nowhere near as popular as, say, the Linux project.

Hence, it would take significant time for the open-source community to build software that could approximate IOS, and if Cisco sees this coming, then Cisco would be well-advised to improve IOS significantly to stay one step ahead. However, it is the opinion of this author that such an outcome, even if such an event were to happen, it would not be a serious blow to Cisco. Such a piece of software might work on a third-party PC-based chassis, but not on Cisco's own hardware. Cisco's routers and switches have extensive proprietary hardware chips such as ASIC's and network processors whose interfaces are not only secret, but are also licensed to Cisco. So even if the open-source community were able to figure out how to emulate these licenses, Cisco could simply change the code on the ASIC's or the network processors at will, which would break the interfaces. The upshot is that even without IOS, Cisco holds substantial proprietary control over its platform. Couple that with the fact that Cisco can just implement a policy of not selling any new router or switch without an accompanying IOS license (hence, no 'headless' sales), and won't support any unlicensed product, and this will serve to preserve all but the most daring of customers. Clearly, Cisco would be hurt by such an outcome, but the result would not be fatal.

8. Conclusion

The networking platform industry can be treated as an entity that shares traits with the IT industry as a whole but also displays attributes that differentiate it from the greater IT industry. In particular, while the networking industry strategy can be dissected with the same 4-lever strategic platform leadership analyses that one can use to understand the greater IT industry to analyze the scope of the firm, the interfaces and architecture of the technology of the firm's products, the interrelationships between the firm and its complementors/partners, and the firm's internal organization, the networking industry also displays a fifth and sixth lever as well. These two additional levers are the firm's ability to take advantage of network effects to increase overall demand, and the firm's ability to outmaneuver competitors. It is this 6-pronged strategy and the firms' ability to utilize the mechanisms provided via these 6 levers, or lack thereof, that determines the firm's success.

The first true networking platform industry leader was clearly IBM through its SNA networking product line. IBM promptly dominated the networking industry for almost two decades not only because of the technical elegance of the SNA technology, but also because of its monopoly-style dominance in mainframe technology. Essentially, if you wanted to access your mainframe remotely, or have various mainframes that were in different locations share resources, SNA was the only game in town, and IBM was the dominant vendor. IBM did tolerate a large SNA-compatible industry to develop under its pricing umbrella, not only because IBM treated such an industry as a cheap extension of its R&D and sales arm, but also to avoid unnecessary antitrust scrutiny. However, IBM clearly maintained a hammer lock on the steering wheel. IBM dictated the terms of SNA licensing, set the path of technological development, and sold the highest-margin networking equipment. IBM engaged telecom companies to develop WAN offerings

such as X.25 and Frame Relay that furthered the reach of SNA networks, an arrangement that proved to be highly successful for all parties.

However, IBM fell victim to myopia. IBM was too wedded to the mainframe and saw little reason to develop networking technology that had nothing to do with transmitting remote mainframe communication. As a result, IBM chose not to forcefully engage in the various markets for networking technologies that were developing at the time. IBM chose not to use Token Ring to make a serious bid to capture the entire LAN market, despite a number of technical advantages that Token Ring enjoyed, thus ceding the market to Ethernet. Most fatally of all, IBM never made a strong push to capture the IP router networking market. IBM slowly but surely allowed SNA to be subsumed within IP until it eventually became just another protocol that was to be bridged over tunneled over an IP core. IBM then let other vendors dominate the market for SNA/IP gateway technology. Nowadays, SNA has been completely marginalized in favor of IP-based Internet traffic, and IBM itself exited the market for networking gear in 1999, selling its entire networking division, including the SNA patents, to Cisco.

While one might say that the victory of IP over SNA was inevitable, what was not inevitable was IBM's complete surrender of the IP networking market. IBM could surely have made a stronger showing if had chosen to do so. Most intriguing of all was that the victory of IP over SNA was perhaps not intriguing. IBM, with some forethought, could have killed the Internet baby in its crib by popularizing SNA as the protocol suite of choice for worldwide open networking communication. After all, the first online services such as CompuServe and Minitel were mainframe communications services running on SNA. With help from IBM, these services could have grown to become what the present-day ISP's are today. Perhaps more importantly, if IBM had provided low-cost and open licensing for SNA to the government and to academia, then they might have chosen to standardize onto SNA rather than IP, and the DARPA net might have been built on SNA foundations, not IP foundations. The founders of the DARPA net were simply looking for a packet-based network protocol that would allow for the sharing of remote computer resources and later for resiliency in the face of nuclear attack. SNA is a networking protocol that allows for remote sharing of computer resources and is also highly resilient. Hence, SNA was a tried-and-trust protocol that could have provided everything that IP did, and more. A tremendous opportunity for IBM was therefore lost. Instead of IBM dominating the commanding heights of the networking industry, IBM was left with nothing at all.

A number of other vendors challenged for the throne, and some even managed to control the throne for awhile. DEC developed the DECnet protocol suite. Banyan developed VINES and StreetTalk. Apple developed the Appletalk technology suite. More seriously of all, Novell created Netware and the IPX protocol and dominated PC networking until the early 90's, and UNIX harnessed the burgeoning popularity of IP and dominated the networking of the nascent Internet. However, ultimately all of these contenders fell by the wayside. All of them wedded to either an operating system, or to hardware, or to both. They therefore saw their networking technology as nothing more than an adjunct technology to allow its core business to function and not as a standalone business in its

own right. Novell and UNIX also suffered from problems with competition – Novell with cutthroat and possibly illegal competition from Microsoft, and the UNIX vendors with each other.

Then along came Cisco. Cisco started from humble beginnings and won the router wars on the back of a strong multiprotocol story coupled with sharp acquisitions that played to the complementarities of the various networking technologies. Cisco then rode the Internet wave, and superseded SNA by converging it with IP to marginalize IBM and snatch away one of the most lucrative markets of corporate networking. Cisco continues to converge other lucrative networking markets like voice and video onto IP. Cisco thus hopes to bring the networking story to its final chapter in which all network communications all converge onto IP, all onto a Cisco dominated network.

Cisco's platform strategy can be broken into 3 components – hardware, software, and management. From a hardware component standpoint, Cisco has become quite open. Cisco is utilizing a significant number of off-the-shelf components and technologies and has handed off most hardware manufacturing to outside parties. Cisco has internally organized itself to emphasize the importance of outsiders to source the components and manufacturing capacity that it needs. Cisco has modularized its hardware design where components can be mixed and matched as necessary, and has engaged in deep long-standing partnerships with suppliers and contract manufacturers.

However, from a customer standpoint, Cisco's hardware platform is completely closed. Cisco relies on a chassis strategy to deter customers from migrating to other vendors' gear by reducing the costs of incremental Cisco upgrades. However, Cisco insists on selling the entire solution. No 3rd party expansion cards can fit into a Cisco router or switch chassis. No 3rd party can manufacture the chassis. While Cisco gear will interoperate with other vendors' gear from a networking standpoint, it will not do so from a chassis standpoint. Cisco's role is clear – while Cisco will happily source components and factory time from numerous 3rd party vendors to create the hardware platform, Cisco wants to maintain complete control of that platform when it is sold to the customer. While such a strategy allows Cisco to maintain high profit margins and control over the technology, it denies Cisco the ability to broaden the extensibility of the hardware by allowing innovative startups to develop new card functionality – functionality that Cisco might later wish to acquire. Cisco may obtain greater power by opening the chassis to a limited number of small vendors while maintaining control over the overall chassis technology direction. Cisco may also choose to abandon the development of low-margin cards that offer simple commodity features like basic Ethernet, and instead let other vendors take up the slack.

Cisco's software strategy, including not only how it manages IOS, but also CatOS, Callmanager, and all of the other software technologies that Cisco controls, demonstrates the full extent of Cisco's mastery of the various platform levers. Cisco's software is open, but not really open. Cisco's software technical configuration guides and command references are open for all to read. Cisco provides open scripting access to IOS. Cisco broadly licenses IOS. However, Cisco maintains complete change control over its

software. While Cisco software is compatible from a networking sense with the software of other vendors, Cisco has developed a number of proprietary features within its software that improve the operation of a pure Cisco network. Cisco has also pushed its software to be the de facto network software such that competing vendors feel the need to emulate the Cisco software command-line interface in order to gain traction with their products. Once somebody learns how to use Microsoft Office, he/she doesn't want to spend the time to learn anything else, and similarly, once a network technician learns how to configure Cisco software, he/she doesn't want to spend the time to learn another user interface. IOS and other Cisco software have therefore become the lingua franca of the networking world, and, just like all other lingua francas throughout history, has served to ease communication. Finally, Cisco actively uses its software as a weapon by constantly bundling features into it that will marginalize its competition, most notably by bundling SNA features to kill IBM, and now bundling voice and video features to capture those markets.

The one weakness of Cisco's software platform strategy is that Cisco may be making the same mistake that Apple made in focusing too much on its integrated hardware and software solution, and not realizing that software is its true *raison d'être*. Cisco may be playing with fire here. Cisco should prioritize software development and pull all of its software groups amongst its various product lines into one centralized division. This would eliminate the frustrating inconsistencies that exist across the supposedly standard IOS software packages and would centralize bug-tracking and feature development.

Finally, Cisco's business strategy serves to complement Cisco's hardware and software strategy. Cisco's marketing has been nothing short of stunning. Cisco has used its marketing to get in the heads of CIO's and Directors of IT everywhere to the point that Cisco is widely considered to be the safe default choice for networking products. Cisco has also stoked demand by being seen for many years as the paragon of information technology and a leader of Internet technology. The unspoken pitch was that if you wanted your business to enjoy the success of Cisco, you have to emulate Cisco, which meant that you had to revamp your IT infrastructure to be able to provide a virtual close and Internet-enabled sales and customer service. Cisco also incubated powerful allies among the training and systems integration/reseller industries by, on the one hand, creating the powerfully branded and popular CCIE and related certification programs, and on the other hand, by making resale partnerships easy and lucrative. Cisco has also coaxed the growth of a large number of Cisco-trained personnel who will inevitably advise their employers to purchase Cisco. Cisco has also deliberately chosen not to make its products easier to use, which would alienate its training and sys-integration allies. Cisco therefore got powerful allies that cemented its status as the king of the hill. It would be a daunting task for any competitor to topple Cisco anytime soon.

To summarize, Cisco has deftly adapted its strategy to utilize the levers of the network platform strategy to capture markets and eliminate competitors. Its hardware, software, and business strategies complement each other and present an interlocking series of challenges to any company who wants to topple Cisco to become the next king of the networking mountain. Cisco has demonstrated the ability to morph its strategies and the

platform levers to further its grip on the networking market. However, Cisco has elected not to open certain parts of its technology – and in particular, has chosen to keep its hardware chassis strategy closed and has chosen not to decouple its software from its hardware – and that may ultimately prove to be Cisco's undoing.

Certain strong competitors, notably Juniper and Avaya, exist in niche markets. However, at this time, they are not a serious existential threat to Cisco. Even if Cisco were to lose those niche markets totally, Cisco would still be a tremendously strong player with enough financial resources to re-enter those markets almost at will. The IP telephony market may eventually disappear anyway, to be dominated by Microsoft and/or to be drained of all profit from consumer VoIP vendors such as Skype. The open-source community is a more serious threat, but is also an opportunity. True, the open-source community could eviscerate Cisco's software position. However, managed properly, and in particular, through deft use of the six levers, Cisco may be able to benefit from the open-source community the way that IBM currently does. We should therefore expect Cisco to dominate in the near future, but should carefully examine how the company utilizes its levers to deal with inevitable changes in the market. Cisco has certainly played its hand brilliantly, and in particular, has learned important lessons from its vanquished foes to dominate its current markets, but these markets are among the most competitive and dynamic in the world so Cisco must constantly fine-tune its strategy to meet the constant new threats. So far so good. It will be interesting to see how long Cisco can stay on top.

9. Bibliography and References

- "Cisco Acquires Dallas-based Selsius Systems". San Francisco Business Times. Oct. 14, 1998.
<http://www.bizjournals.com/sanfrancisco/stories/1998/10/12/daily9.html>
- "Ewing, James. Linux on Linksys Wi-Fi Routers". Linux Journal. Aug 1, 2004.
<http://www.linuxjournal.com/article/7322>
- Ante, Stephen. "Commentary: The Four Horsemen of the New Economy". Businessweek. Oct. 2, 2000.
http://www.businessweek.com/2000/00_40/b3701082.htm
- Anthes, Gary. "Changing Fortunes". Computerworld. Nov. 12, 2001.
<http://www.computerworld.com/networkingtopics/networking/story/0,10801,65534,00.html>
- Artesyn Technologies. "Teledatcom Glossary: X25". March 4, 2006.
<http://www.artesynpc.com/resources/glossary/x-25.html>
- Bacon, Chet. "Distance Vector Concept". March 3, 2006.
http://www.chetbacon.com/lessonplans/sem3/images/5_d.gif
- Bacon, Chet. "IPX Networking". March 2, 2006.
http://www.chetbacon.com/lessonplans/sem3/images/7_c.gif
- Bell System Memorial. "AT&T Divestiture". March 18, 2006.
http://www.bellsystemmemorial.com/att_divestiture.html
- Biran, Gil. "Voice over Frame Relay, IP and ATM". Protocols. May 1, 2006.
<http://www.protocols.com/papers/voe.htm>
- Blanchard, Eugene. "Introduction to Data Communications". 1999.
http://www.techbooksforfree.com/intro_to_data_com/page176.html
- Boardman, Bruce. "Picky Picky: SNA over Frame Relay". Feb. 27, 1996.
<http://www.networkcomputing.com/703/703work3.html>
- Brisbin, Shelly. "Appletalk Chattness: Fact or Fiction". Mac Publishing. 1998. <http://mac-guild.org/appletalk.html>
- Broadband Reports. "Cisco Forum FAQ – Smartnet 20.1". Broadband Reports.com, March 16, 2006.
<http://www.dslreports.com/faq/cisco/20.1+Smartnet#13460>
- Broesma, Matthew. "At Web Sites, Windows Outpaces Linux". CNET. July 18, 2003.
http://news.com.com/At+Web+sites,+Windows+outpaces+Linux/2100-1016_3-1027188.html
- Bunnell, David & Bate, Adam. . "Making the Cisco Connection." John Wiley & Sons. 2000.
- Burek, Joan. "The Frame Relay Primer". E-server magazine, Vol. 12, #2, Nov. 1996.
<http://www.tug.on.ca/articles/Volume12/V12N2/V12N2FrameRelayPrimer.html>
- Burrows, Peter. "Can Cisco Settle for Anything Less than Sizzling?" Businessweek. Feb. 21, 2005.
http://www.businessweek.com/magazine/content/05_08/b3921093_mz063.htm
- Caslon Analytics. "The Dotcom bubble". Marc 20, 2006. <http://www.caslon.com.au/boomprofile10.htm>
- Chappell, Laura. "Migrating to Pure IP with Netware 5". Netware Connection. Sept 1998.
http://support.novell.com/techcenter/articles/nc1998_09c.html
- Cheshire, Stuart. "Appletalk NBP (Name Binding Protocol). Sep. 1997.
<http://www.stuartcheshire.org/rants/NBP.html>
- Christensen, Clayton. "The Innovator's Dilemma". Harvard Business School Press. 1997. Chapter 2.
- Chun, David, editor. "IBM 1956 Consent Decree". US Department of Justice Document Group. Jan. 25, 1956. <http://www.cptech.org/at/ibm/ibm1956cd.html>
- Cisco Press. "About Us". 2006. <http://www.ciscopress.com/about/index.asp>
- Cisco Systems "Cisco ONS 15400 Series Command References". May 1, 2006.
http://www.cisco.com/en/US/products/hw/optical/ps2006/prod_command_reference_list.html
- Cisco Systems CCIE website. May 5, 2006.
http://www.cisco.com/web/learning/le3/ccie/certified_ccies/worldwide.html
- Cisco Systems documentation. "Internetworking Technology: IBM Systems Network Architecture (SNA) Protocols". March 15, 2006. http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ibmsna.htm
- Cisco Systems documentation. "Internetworking Technology: Token Ring". March 15, 2006.
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/tokenrng.htm
- Cisco Systems Network Academy "About" homepage. May 1, 2006.
<http://www.cisco.com/web/learning/netacad/academy/About.html>
- Cisco Systems Partner Website. May 1, 2006.
http://www.cisco.com/warp/public/765/partner_programs/specialization/spec_partner_points.shtml

Cisco Systems. "Cisco 6500 Catalyst Switch". http://www.asl-cisco.co.uk/images/screenshots/cdccont_0900aecd8006994b.gif

Cisco Systems. "Cisco Catalyst 6500 and 6500E Series Switch Data Sheet". April 2, 2006. http://www.cisco.com/en/US/products/hw/switches/ps708/products_data_sheet09186a00800ff916.html

Cisco Systems. "Cisco Catalyst 6500 Series Switches". Feb. 15, 2006. <http://www.cisco.com/en/US/products/hw/switches/ps708/>

Cisco Systems. "Cisco IOS Bridging and IBM Networking Configuration Guide, Release 12.1" – part 2, configuring NCIA server.. March 15, 2006.

Cisco Systems. "Cisco IOS Software Releases 12.0 Mainline: The Show Processes Command" Document ID 15102. Cisco IOS documentation. May 6, 2006. http://www.cisco.com/en/US/products/sw/iosswrel/ps1828/products_tech_note09186a00800a65d0.shtml#s_howproc

Cisco Systems. "Cisco MGX 8900 Series Switches Command References". May 1, 2006. http://www.cisco.com/en/US/products/hw/switches/ps3873/prod_command_reference_list.html

Cisco Systems. "Cisco Systems to Acquire Token Ring Switch Innovator Nashoba Networks". News@Cisco. Aug 6, 1996. http://newsroom.cisco.com/dlls/1996/corp_080696.html

Cisco Systems. "Cisco Transaction Connection Overview". Nov. 5, 2002. <http://www.cisco.com/univercd/cc/td/doc/product/ibd/ctrc/ctrcin.htm#xtocid3>

Cisco Systems. "Flexible Migration Paths for IBM Networking". Document ID 12305. Cisco Technology White Papers. . Sep. 9, 2005. http://www.cisco.com/en/US/tech/tk331/tk336/technologies_white_paper09186a0080114a38.shtml

Cisco Systems. "Internetworking Technology Handbook: Appletalk". March 20, 2006. http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/applet.htm#xtocid6

Cisco Systems. "Internetworking Technology Handbook: Chapter 26, Banyan VINES". March 20, 2006. http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/vines.pdf

Cisco Systems. "Network Management Product Catalog". 2006. <http://www.cisco.com/en/US/products/sw/netmgts/index.html>

Cisco Systems. "Product Overview – the Cisco 7010 Router". 1997. <http://www.cisco.com/univercd/cc/td/doc/product/core/cis7010/7010ug/prodolog.htm>

Cisco Systems. "Cisco Interface Processors: Introducing SNA on the CMCC". March 18, 2006. http://www.cisco.com/en/US/products/hw/modules/ps2643/prod_technical_reference09186a00802381e9.html#wp813935

Cisco Systems. "Cisco IOS Software Release 12.3 New Features and Platforms". 2006. http://www.cisco.com/en/US/products/sw/iosswrel/ps5187/prod_bulletin09186a0080199900.html

Cisco Systems. "Gold Partner Requirements". 2006. http://www.cisco.com/web/partners/pr11/pr8/pr27/partners_pgm_requirements.html

Cisco Systems. "IBM Internetworking". Document ID 12305. March 14, 2006. http://www.cisco.com/en/US/tech/tk331/tk336/technologies_white_paper09186a0080114a38.shtml

Cisco Systems. "Premier Partner Requirements". 2006. http://www.cisco.com/web/partners/pr11/pr8/pr51/partners_pgm_requirement_summary.html

Cisco Systems. "SDLLC Direct Configuration Example". Document ID 12281. Cisco Configuration Examples and Technotes. Sep, 9, 2005. http://www.cisco.com/en/US/tech/tk331/tk336/technologies_configuration_example09186a0080093d9a.shtml

Cisco Systems. "STUN Basic with Multiple Tunnels". Document ID 12275. Cisco Configuration Examples and Technotes. Sep 9, 2005. http://www.cisco.com/en/US/tech/tk827/tk369/technologies_configuration_example09186a00801241f6.shtml

Cisco-Centric Open Source Exchange Community homepage. May 5, 2006. <http://cosi-nms.sourceforge.net/>

Commweb. "VoIP/IP Telephony Statistics". Oct 15, 2002. <http://www.techweb.com/article/COM20021015S0002>

Computerhope.com. "Ethernet". March 25, 2006. <http://www.computerhope.com/jargon/e/ethernet.htm>

Cooney, Michael. "Determined but Battered IBM in Fight of its Token Ring Lift". Network World. May 13, 1996. http://www.findarticles.com/p/articles/mi_qa3649/is_199605/ai_n8745983

Core Competence. "Appletalk Networking: A Lower Layers Primer". March 1, 2005.
<http://www.corecom.com/html/appletalk.html>

Cringely, Robert X. "Breaking the Bank". I, Cringely. May 9, 2002.
<http://www.pbs.org/cringely/pulpit/pulpit20020509.html>

Cringely, Robert X. "Valley of the Nerds: Who Really Invented the Multiprotocol Router, and Why Should We Care?". I, Cringely. Dec. 10, 1998. <http://www.pbs.org/cringely/pulpit/pulpit19981210.html>

Cusumano, Michael and Gawer, Annabelle.. "Platform Leadership: How Intel, Microsoft, and Cisco Drive Industry Innovation". Harvard Business School Press. 2002.

D'Allesandro, Paul; deLeon, Jeff; Littman, Daniel; and Quane, Stephen. "Case 8: Intel's Gigabit Ethernet Strategy". March 4, 2006. <http://www-personal.umich.edu/~afuah/cases/case8.html>

Data Network Resource. "ISO CLNP, IS-IS, and ISO IGRP". March 15, 2006.
<http://www.rhyshaden.com/isis.htm>

Dawson, Fred. "A Role for ATM in Managing Local Traffic?". Communications Engineering & Design. April 1997. <http://testced.cahners1.com/ced/9704/9704c.htm>

Demarter, Richard, editor. "IBM Antitrust Suit Records 1950-1982". 1980.
<http://www.hagley.lib.de.us/1980.htm>

Dix, John. "A Timeline of Network Developments". Network World. March 26, 2001.
<http://www.networkworld.com/anniversary/line/line.html>

Duffy, Jim. "Juniper Gains Corporate Network Ground". Network World. Oct 10, 2005.
<http://www.computerworld.com.au/index.php/id;1133614807;relcomp;1>

E-consultancy.com. "AT&T to Acquire IBM's Global Network Business for \$5 billion" (press release). Dec. 8, 1998. <http://www.e-consultancy.com/newsfeatures/19595/at-t-to-acquire-ibm-s-global-network-business-for-5-billion.html>

ECPI College of Technology Description of Programs. April 30, 2006.
<http://www.ecpi.edu/military/programs/index.cfm?pgNum=16>

Falkotimme. "Mail Server Survey". April 2004.
http://www.falkotimme.com/projects/survey_smtp.php?id=170

Ferguson, Charles. "High Stakes, No Prisoners". Times Books. 1999.

Fortune Magazine: "100 Best Companies to Work For".
<http://money.cnn.com/magazines/fortune/bestcompanies/>

Frame Relay Forum. "The Basic Guide to Frame Relay Networking". 1998.
http://www.tti.unipa.it/~alcuri/Materiale_Telematica/FrameRelay.pdf

Fried, Ina & Kanellos, Michael. "Dell Calls for Blade Server Standards". CNET. Sep. 8, 2003.
http://news.zdnet.com/2100-9584_22-5072603.html?tag=nl

Gibney, Frank. "You Name It, We'll Make It". Time Global Business. August 2001.
<http://www.time.com/time/global/august/cover.html>

Gittlen, Sandra. "IBM to Charge for QoS Patents". Network World. Nov. 9, 1998.
<http://www.networkworld.com/news/1109mpls.html>

Goodlin, Dan. "Microsoft Emails Focus on Dr. DOS Threats". CNET. April 28, 1999.
http://news.com.com/Microsoft+emails+focus+on+DR-DOS+threat/2100-1001_3-225129.html

Grant, Todd. "Will it be Lobster or Linguini?". Novell Cool Solutions Feature. Apr. 25, 2001.
<http://www.novell.com/coolsolutions/feature/90.html>

Granville, Fred. "The Road to CCIE – Part Three". Independent Thoughts. April 27, 2006.
<http://www.flgnetworking.com/article4.html>

Greene, Thomas C. "Sun's Linux Killer Shows Promise". The Register. Aug 16, 2005.
http://www.theregister.com/2005/08/16/solaris_x86_not_too_shabby/

Greene, Tim & Rohde, David. "MCI Worldcom's Frame Net Face Lift". Network World. Nov. 9, 1998.
<http://www.networkworld.com/news/1109mciframe.html>

Grygus, Andrew. "Novell Netware: The King Returns from the Dead". March 2, 2006.
<http://www.aaxnet.com/design/novell.html>

Grygus. "SCO – Death Without Dignity". AAXnet.com. Jun 2003.

Haley, Colin. "Juniper Routes Core Objective". Internetnews.com. Dec. 3, 2004.
<http://www.internetnews.com/infra/article.php/3443571>

Handle name: edelfenbein. "Will Cisco Pay a Dividend?". Aug 11, 2005. Crossing Wall Street.
http://www.crossingwallstreet.com/archives/2005/08/will_cisco_pay.html

Hart, David M. "Antitrust and Technological Innovation". Issues in Science and Technology, Winter 1998 issue. <http://www.issues.org/issues/15.2/hart.htm>

Heard, Kevin. "UNIX Tutorial: Section 12 Viewing Processes". Berkeley SIMS. Jan. 27, 2006. <http://www.sims.berkeley.edu/~kevin/unix-tutorial/section12.html>

Hermes Group. "SNA: A Technological Overview". March 1, 2006. <http://www.hermesgroup.com/whitepapers/SNA/sna.html>

Heskett and Wong, Wylie. "Juniper Finds its Own Networking Niche". CNET. Aug 16, 1999. http://news.com.com/Juniper+finds+its+own+networking+niche/2100-1033_3-229831.html

Heskett, Ben. "A Fresh Vision for Novell?". CNET. March 1, 2002. <http://news.com.com/2100-1001-848816.html>

Heskett, Ben. "Cisco Raises the Voice Market Stakes". CNET. April 13, 1999. http://news.com.com/Cisco+raises+the+voice+market+stakes/2100-1033_3-224348.html

Hochmuth, Phil & Duffy, Jim. "Challenges Dog Cisco as Company's Role Expands". Network World. May 26, 2003. http://www.findarticles.com/p/articles/mi_qa3649/is_200305/ai_n9264446

Hochmuth, Phil & Duffy, Jim. "Cisco to Juice 6500 Switch". Network World. Aug 15, 2005. <http://www.networkworld.com/news/2005/081505-cisco.html>

Hochmuth, Phil. "Cisco Drops Suit Against Huawei". Network World. July 28, 2004. <http://www.networkworld.com/edge/news/2004/0728huawei.html>

http://www.cisco.com/en/US/products/sw/iosswrel/ps1831/products_configuration_guide_chapter09186a00800d982b.html

Huitema, Christian. "Routing and the Internet, 2ed.". Prentice Hall. 2000.

IBM. "Cisco and IBM Collaborate to Design and Build the World's Most Sophisticated, High-Performance 40 Gbps Custom Chip" IBM Press Release. June 9, 2004. http://www-03.ibm.com/chips/news/2004/0609_cisco.html

IEEE 802.3u1 Fast Ethernet Specification

IEEE 802.5 Basic Token Ring specification

IEEE 802.5 Basic Token Ring Specification.

IEEE 802.5r Dedicated Token Ring Specification.

IEEE 802.5t 100 Mbps Token Ring specification

IEEE 802.5v 1000 Mbps Token Ring specification

InetDaemon. "Bus Topology". Feb. 2, 2006. http://www.inetdaemon.com/img/topology_bus.png

Intel Inc. "Intel to Acquire Shiva Corporation". Intel Press Release. Oct 19, 1998. <http://www.intel.com/pressroom/archive/releases/HN101998.HTM>

Intel Inc. "Network Connectivity". Intel Support document. March 20, 2003. <http://support.intel.com/support/network/sb/cs-009528.htm>

ISP-Planet Staff. "ISP Backbone Market Forecast – Flat Through 2002". ISP-Planet.com. Jan 23, 2002. http://www.isp-planet.com/research/2002/backbone_020123.html

Juniper Network Company Profile. May 20, 2006. <http://www.juniper.net/company/profile/leadership.html>

Juniper Networks. "Juniper T-Series". May 10, 2006. <http://www.juniper.net/products/tseries/>

Juniper Networks. "T-series Product Guide". 2006. <http://www.juniper.net/products/tseries/>

Keating, Tom. "Cisco Unified Communications Launches". VoIP & Gadget Blog. Mar 5, 2006. <http://blog.tmcnet.com/blog/tom-keating/voip/cisco-unified-communications-changes-everything.asp>

Krapf, Eric. "Cisco Folds IP-PBX into IOS". Business Communications Review. Nov 1, 2003. <http://www.bcr.com/bcsmag/2003/11/p56.php>

Krapf, Eric. "IBM-Cisco: End of an Era for Big Blue". Business Communications Review. Oct 1, 1999. <http://www.bcr.com/bcsmag/1999/10/p16.php>

Lawson, Stephen. "Network Processors Enter New Generation". Network World. Jun 19, 2002. <http://www.networkworld.com/news/2002/0619processors.html>

Lee, Ron & Major, Drew. "Netware on One CPU Outperforms Windows NT Server on Four CPU's". Novell Appnotes. Jun 1, 1996. <http://support.novell.com/techcenter/articles/ana19960601.html>

Library noi.net. "Routing Protocols Reference Guide, Chapter 9, Appletalk". March 1, 2006. <http://library.noi.net/networking/protocols/ro-routing-refg/ch9-apl.html>

Lightreading.com. "RAD aids Ethernet over ATM". Aug 29, 2005. http://www.lightreading.com/document.asp?doc_id=79683

Linux Router homepage. May 1, 2006. <http://www.linuxrouter.org/>

Linux-DECnet Sourceforge Project. "DECnet for Linux". March 15, 2006. <http://linux-decnet.sourceforge.net/lat.html>

Malhotra, Ravi. "IP Routing – Chapter 4, EIGRP". O'Reilly Publishing. Jan. 2002. <http://www.oreilly.com/catalog/iprouting/chapter/ch04.html>

Malik, Om. "The Black Box That Would Conquer Telecom". Business 2.0. Feb. 22, 2006. http://money.cnn.com/magazines/business2/business2_archive/2006/03/01/8370567/index.htm

Matsumoto, Craig. "Azanda Lands a Buyer". Heavy Reading. Jan 20, 2005. http://www.heavyreading.com/document.asp?doc_id=66312&site=lightreading

Matsumoto, Craig. "Cisco's CRS-1 Passes our Test". Light Reading. Nov. 30, 2004. http://www.lightreading.com/document.asp?doc_id=63607

Mayer, David. "Ecosystems and Acquisition Management: Understanding Cisco's Strategy". <http://hcd.ucdavis.edu/faculty/kenney/articles/siliconvalley/cisco%20ver.%203.doc>, Industrial and Corporate Change (forthcoming), Dec 22, 2003.

Microsoft Inc. "Windows 2000 DNS". Microsoft Technet. May 1, 2006. <http://www.microsoft.com/technet/prodtechnol/windows2000serv/plan/w2kdns2.mspx>

Microsoft Inc. "Windows NT Server Resource Kit" Appendix D – Windows Sockets. May 1, 2006. http://www.microsoft.com/resources/documentation/windowsnt/4/server/reskit/en-us/net/sur_tsok.mspx

Morettini, Phil. "The Rise and Fall of Novell". PJM Consulting blog. Nov. 16, 2005. <http://www.pjmconsult.com/2005/11/rise-and-fall-of-novell.html>

Morgan, Timothy Prickett. "Sun to Buy Supercomputer-Maker Cray?" IT Jungle. June 30, 2004. <http://www.itjungle.com/breaking/bn063004-story02.html>

Moritz, Scott. "Cisco, Juniper Find New Foe". TheStreet.com, Feb. 24, 2006. http://www.thestreet.com/_googlen/tech/networking/10270135.html?cm_ven=GOOGL&cm_cat=FREE&cm_ite=NA

Morris, Scott. "So You Wanna be a CCIE? Part 1". CertCities. Jun 28, 2005. <http://certcities.com/editorial/features/story.asp?EditorialsID=89>

Morris, Scott. "Should I Become a CCIE or a JNCIS/JNCIE?" TCPMag.com. May 25, 2004. <http://tcpmag.com/qanda/article.asp?EditorialsID=281>

Morrison, Gale. "Outsourcing Comes to Network OEM's". Electronic News. Aug 27, 2001. http://findarticles.com/p/articles/mi_m0EKF/is_35_47/ai_77673033

MsFiscallyFit. "Cisco Systems – the Cisco Kid Grows Up". MsFiscallyFit.com, May 7, 2000. http://www.ms FiscallyFit.com/investing/high_impact/hot_stocks_6/hot_stocks_6.html

MundoPC. "Star topology". March 15, 2006. <http://www.mundopc.net/cursos/redes/images/star.gif>

Murphy, Paul. "Goodbye SuSE, SuSE Goodbye". LinuxInsider. Nov. 13, 2003. <http://www.linuxinsider.com/story/32137.html>

Musthaler, Linda. "Netware: It Just Works:". Network World. July 15, 2002. <http://www.networkworld.com/columnists/2002/0715musthaler.html>

Natal, Frank. Editorial comment within "Innovation Insights, A Look Back",. Oct. 18, 2004. <http://www.informationweek.com/showArticle.jhtml?articleID=50500311&pgno=5>

Network Dictionary. "Peer to Peer Networking". March 18, 2006. <http://www.networkdictionary.com/networking/peer.php>

Network World. "Response to RFP: Altiga". May 10, 1999. <http://www.networkworld.com/reviews/0510rfpaltiga.html>

Newman, David. "Enterasys 10G Switch Goes its Own Way – Fast". Network World. Nov 19, 2001. <http://www.networkworld.com/reviews/2001/1119rev.html>

Novell Inc. "eDirectory Positioning". Novell Cool Solutions. Jan 18, 2001. <http://www.novell.com/cool solutions/feature/5318.html>

Novell Inc. "Netware SDK for Java Platform". Novell Appnotes. Jun 1, 1996. <http://support.novell.com/techcenter/articles/dnd19960607.html>

Novell Inc. "Novell Reports Third Fiscal Quarter 2002 Results". Novell Press release. Aug 22, 2002. <http://www.trainingpressreleases.com/newsstory.asp?NewsID=392>

Novell Inc.. "Novell Settles One Antitrust Claim with Microsoft for \$536 million, Plans to File Suit for Second Claim". Novell Press Release. Nov 8, 2004. <http://www.novell.com/news/press/archive/2004/11/pr04076.html>

Online Books Library. "VINES diagram". March 15, 2006.
<http://www.emu.edu.tr/english/facilitieservices/computercenter/bookslib/MS%20BackOffice%20Administrator's%20Survival%20Guide%202nd%20Ed/24mba02.gif>

Oppenheimer, Allan. "A History of Mac networking". Jan 7, 2006.
<http://macslash.org/article.pl?sid=04/01/07/1555239&mode=thread>

Packet Attack. "FrankenPix 101". Jan 5, 2006. <http://www.packetattack.com/frankenpix.html>

Passmore, David. "A Modern IOS". Business Communications Review. Oct. 3, 2005.
<http://www.bcr.com/bcramag/2005/10/p16.php>

PC Tech Guide. "Token Ring". Dec. 1, 2002. http://www.pctechguide.com/29network_Token_Ring.htm

Perlman, Radia. "Interconnections 2ed: Bridges, Routers, Switches and Interconnection Protocols". Addison-Wesley. 2000. Chapter 7.2.

Protocols.com. "SNA Protocols". March 3, 2006. <http://www.protocols.com/pbook/sna.htm>

Pulse. "What is OSI Routing?" 2000. http://www.pulsewan.com/data101/osi_routing_basics.htm

Rash, Wayne & Rist, Oliver. "10Gig Slugfest". InfoWorld. April 25, 2003.
http://www.infoworld.com/article/03/04/25/17Fetengig_4.html

Raymond, Eric Steven. "Origin and History of UNIX, 1969-1995". Chapter 2: History. 2003.
<http://www.faqs.org/docs/artu/ch02s01.html>

Raynovich, R. Scott. "Cisco Ships OC-192". Light Reading. Jan 31, 2001.
http://www.lightreading.com/document.asp?doc_id=3455

Reardon, Marguerite. "The Skinny on Cisco's Product Strategy". CNET. June 9, 2004.
http://news.com.com/The+skinny+on+Ciscos+product+strategy/2008-1033_3-5223552.html

Reference.com. "Daemon (mythology)". [http://www.reference.com/browse/wiki/Daemon_\(mythology\)](http://www.reference.com/browse/wiki/Daemon_(mythology))

Reinhardt, Andy. "John Chambers". Businessweek Online. May 15, 2000.
http://www.businessweek.com/2000/00_20/b3681028.htm

Reinhardt. "Meet Cisco's Mr. Internet". Businessweek. Sep. 13, 1999.
http://www.businessweek.com/1999/99_37/b3646001.htm

RFC 1041. <http://www.faqs.org/rfcs/rfc1041.html>

RFC 1131. <http://www.faqs.org/rfcs/rfc1131.html>

RFC 1195. <http://www.faqs.org/rfcs/rfc1195.html>

RFC 2136. <http://rfc.sunsite.dk/rfc/rfc2136.html>

RFC 2338. <http://www.faqs.org/rfcs/rfc2338.html>

RFC 2784. <http://www.faqs.org/rfcs/rfc2784.html>

Rudich, Joe. "Ethernet Accelerates". Aug. 23, 2001.
<http://techupdate.zdnet.com/techupdate/stories/main/0,14179,2807260,00.html>

Salkever, Alex. "Does XORP Have Cisco's Number". Businessweek. Nov 29, 2004.
http://www.businessweek.com/technology/content/nov2004/tc20041129_5206_tc024.htm

Salkever, Alex. "Microsoft: Your Next Phone Company?" Businessweek. Mar 2, 2004.
http://www.businessweek.com/technology/content/mar2004/tc2004032_6587_tc076.htm

Saunders, Stephen. "Juniper Wins Monster Router Test". Light Reading. March 12, 2001.
http://www.lightreading.com/document.asp?doc_id=4090

Silwa, Carol. "Q&A: Novell CEO touts Linux, Identity Products". Computerworld. April 27, 2005.
<http://www.techworld.com/opsys/features/index.cfm?featureid=1385&Page=1&pagePos=17>

Slater, Robert. "The Eye of the Storm. How John Chambers Steered Cisco Through the Technology Collapse." Collins Publishing. 2003.

Software Magazine. "IBM gathers Moss & Co.". Software Magazine. March 1996.
http://www.findarticles.com/p/articles/mi_m0SMG/is_n3_v16/ai_18030924

Songini, Marc. "IBM Has All But Thrown in the Towel on its Networking Hardware Division". Network World. Aug 31, 1999. <http://www.networkworld.com/news/1999/0831ibmcisco.html>

Stauffer, David. "Nothing but Net. Business the Cisco Way". Capstone Publishing. 2001.

Strom, David. "Is Cisco Vulnerable?". VARBusiness. Jun 2, 2003.
<http://www.varbusiness.com/sections/news/dailyarchives.jhtml;jsessionid=RKWSK3NGOYWK0QSNDBCSKHSCJUMKJVN?articleId=18830254&requestid=29855>

Sun Microsystems. "Sun Trademarks". Sun Trademarks Homepage. <http://www.sun.com/suntrademarks/>

Symonds, Matthew. "Softwar: An Intimate Portray of Larry Ellison and Oracle." Simon & Schuster. 2003. P. 46-50.

Synergy Research Group. "Enterprise IP Telephony Posts Another Strong Year in 2005". Marketwire. Feb. 27, 2006. http://www.marketwire.com/mw/release_html_b1?release_id=111128

Sysprog. "Quotations -> Operating Systems". May 15, 2006. <http://www.sysprog.net/quotos.html>

Takahashi, Dean. "Crunching the Numbers". ESM. Nov 15, 2004. <http://www.my-esm.com/printableArticle.jhtml?articleID=51202436>

Techiwarehouse. "Instant Messaging". May 12, 2003. http://www.techiwarehouse.com/cms/engine.php?page_id=1746eedd

The Linux Review. Token Ring diagram. Jan. 5, 2006. http://www.thelinuxreview.com/howto/intro_to_networking/stdimages/page176a.gif

Thomas, Tom M. "Juniper Networks Router Architecture". Addison-Wesley Professional. Jan. 24, 2003. <http://www.informit.com/articles/article.asp?p=30631&rl=1>

Unisys. "Unisys completes Acquisition of ePresence Security Services Business" – Press release. June 3, 2004. http://www.unisys.com/services/news_a_events/all_news/06038416.htm

UPS. "UPS Supply Chain Solutions Awards for Business Excellence". UPS Fact Sheets. May 15, 2006. <http://www.pressroom.ups.com/mediakits/factsheet/0,2305,1053,00.html>

US Patent 5237566. "Network Hub for Maintaining node bandwidth in a Single Node Network". <http://www.freepatentsonline.com/5237566.html>

VoIP-info.org. "Asterisk Call-Manager Integration". May 27, 2006. <http://www.voip-info.org/wiki-Asterisk+Cisco+CallManager+Integration>

Webconsult. The 1980's (Divestiture and a Whole New Ballgame". March 20, 2006. <http://www.webbconsult.com/1980.html>

Weiss, Todd. "SCO pulls IBM's AIX license in UNIX dispute". Computerworld. Jun 16, 2003. <http://www.computerworld.com/softwaretopics/os/linux/story/0,10801,82200,00.html>

Welcher, Peter. "TCL'ing your Cisco Router". Chesapeake NetCraftsman. Jan 3, 2005. <http://www.netcraftsmen.net/welcher/papers/iostcl01.html>

Westnet Learning Technologies and Digital Creators. "LAN Architectures". March 3, 2006. http://ih.esuhsd.org/staff/aielloj/westnet/westnet/units/unit4_sec2.html

WHNet. "Gigabit IP Packet Switching Routers". WHNet.com. July 2002. <http://www.whnet.com/giga.html>

Wikipedia Encyclopedia Entry on Metcalfe's Law: http://en.wikipedia.org/wiki/Metcalfe's_law

Wikipedia. "Banyan VINES". March 20, 2006. http://en.wikipedia.org/wiki/Banyan_VINES

Wikipedia. "Netware". March 3, 2006. <http://en.wikipedia.org/wiki/Netware>

Wollack, Todd. "Cisco Passes Microsoft in Worth". San Francisco Chronicle. March 28, 2000. <http://www.sfgate.com/cgi-bin/article.cgi?file=/chronicle/archive/2000/03/28/MN1065CH.DTL>

Wood, Robert S. "A Network Administrator's View of Multiservice Networks". Courtesy of Cisco Press. Dec. 9, 2005. <http://www.informit.com/articles/article.asp?p=426645&rl=1>

Woodie, Alex. "IBM Rents Linux Partitions Under Utility Sales Model. The Four Hundred: iSeries and AS/400 Insight. Vol 11, #28, July 22, 2002. <http://www.itjungle.com/tfh/tfh072202-story06.html>

Xerox PARC. "About PARC: History". March 25, 2006. <http://www.parc.com/about/history/default.html>

Xerox System Integration Standard - Internet Transport Protocols

Zebra Project homepage. May 1, 2006. <http://www.zebra.org/>

Zycko Networking Products. "Product Catalog – Cisco Memory". 2005. http://www.zycko.com/networking/products/networking_memory.asp