DNS/DHCP & NetWare 5 @ Novell
A Beigepaper

by Grettir Asmundarson (grettir@neticus.com)
Last Revised: June 30, 1999

A Service Of Novell's Information Services & Technology Global Technical Architecture Group

About The "@ Novell" Series

Most documentation starts as hastily scrawled notes from sleep-deprived developers who weren't necessarily hired for their keen communication skills. Those notes are then fleshed out by recently graduated English majors who have spent their last four years immersed in works of fiction. The results are then passed on to the marketing department whose job it is to make sure that no word or phrase, even if it's true, will reflect unfavorably on the product ("I don't think that the word 'Basic' properly communicates the exciting nature of the product. Why don't we call it 'Visual Zesty!?!'"). It is then beset by lawyers who finish the job by making sure that they haven't explicitly promised that the product will actually do anything.

By the time the documentation gets into your hands, it has been so sanitized for your protection and generalized beyond recognition that you usually have to go out and buy a 3rd-party manual (that was, more likely than not, written by the same non-technical technical writer who wrote the original documentation) in a vain attempt to get an unbiased, unexpurgated, and/or unfiltered view of just how you're really supposed to use the stuff.

That's where the "@ Novell" series comes in. Rather than the vague, generalized, and wholly fictional examples found in most documentation, we're going to tell you exactly how we use our own products to run our own company. After all, we are not a small, tidy computing environment suitable for documentation. We are a big, sprawling, untidy environment made up of over 500 production servers and 20,000 workstations in 130 locations throughout the world. In other words, we're probably an awful lot like you.

And it's not that we're necessarily any smarter than you are, we just have a distinct advantage. By the time you get your hands on one of our released products, we've already been using it to run our business for quite some time. For instance, a month before NetWare 5 shipped, well over half of our 500 production file servers had already been upgraded to NetWare 5. (Keep in mind that these were production servers. These were not test servers that we had safely tucked away in antiseptic labs. These were real-world servers in a real-world environment solving real-world problems.) And two months before NetWare 5 shipped, we'd already converted one of our buildings to IP Only. That means that we've probably gained some insights into implementing our products in a big, sprawling, untidy environment, and this paper is an attempt to share those big, sprawling, untidy insights with our customers.

But keep in mind that this document may be a little rough. It wasn't conceived by a committee, written by a committee, or approved by a committee, so it hasn't been edited, re-edited, tidied up, sanitized, and whitewashed. Don't think of this as an official whitepaper. It's more like a beigepaper.

What's In A Name?

Names can provide us with insights into the purpose or nature of things and people. For instance, a company with a name like Lucent Technologies automatically benefits from the

images conjured up by the word "lucent": clarity, luminescence, etc. Amazon.com elicits images of a vast river of knowledge, full of mystery and wonders to be discovered. Cisco benefits from being named after any of a variety of whitefish (genus *Coregonus*); especially Lake Herring. And Intel combined the words "celery" (an edible plant in the parsley family) and "Oberon" (the fairy king in Shakespeare's *A Midsummer Night's Dream*) to come up with Celeron, "The Bard's Relish Plate" of microprocessors.

Naming also has profound psychological importance. In *The Psychological Attitude of Early Buddhist Philosophy*, Anagarika B. Govinda wrote, "…things that could be named had lost their secret power over man, the horror of the unknown. To know the name of a force, a being or an object was (to primitive man) identical with the mastery over it."

A Rose By Any Other Name…

And naming is becoming even more important with NetWare 5. In the IPX world, you could afford to be lazy. If you wanted to attach to a particular server, you didn't need to know where that server was located, its IP address, its domain, or the NDS context in which the server resided. You could simply refer to the server by its short name. The same is not true in the world of Pure IP. In a Pure IP environment, naming (something you never really had to think about before) suddenly becomes a big issue. Because...

"A rose by any other name…
would have a different IP address."

- Grettir Asmundarson

With NetWare 5, any number of "names" can refer to the same server. If I want to use the Novell GUI Map utility to map a drive to the server that I have here at my desk, what "name" should I use?

Fully-Qualified DNS Name	\\prv-botanica.provo.novell.com\sys
Fully-Qualified NDS Name	\\novell_inc\.prv-botanica_sys.gta.prv.novell
Short Name	\\prv-botanica\sys
IP Address	\\192.108.102.1\sys

DNS: The Namespace Of The Rose

Here @ Novell, we've standardized on fully-qualified DNS names. Why? Because DNS is the only naming standard that will work no matter where I am. Whether I'm mapping a drive locally, via a VPN client across the Internet, or dialing in with RADIUS, fully-qualified DNS names work everywhere. And if I ever have to change the IP address of the server, I simply point the DNS entry to the new IP address and everyone's still working.

DNS makes the TCP/IP world a much nicer place because it allows you to refer to things by user-friendly DNS name (hvannadalshnukur.hafnarfjordur.com) rather than obscure IP address (192.108.102.1). It also allows us organize things into logical, geographic subnets that are easily understandable to the average user. For instance, our three main sites in the U.S. are:

Location	DNS Subnet Name
Provo, Utah	provo.novell.com
Orem, Utah	orem.novell.com
San Jose, California	sjf.novell.com[1]

We've also standardized the DNS names of particular servers and/or services at most sites. For instance:

DNS Entry	Record Type	Description
mail.*site*.novell.com	A	MX record for *site*.novell.com to reference GroupWise IP connection.
ns.*site*.novell.com	NS	Site DNS Server
news.*site*.novell.com	CNAME	Site Newsgroup/Collaboration Server
proxy.*site*.novell.com	CNAME	HTTP and FTP Proxy/Cache Server
www.*site*.novell.com	CNAME	Site Web Werver
da.*site*.novell.com	CNAME	Site Directory Agent
ma.*site*.novell.com	CNAME	Site Migration Agent
ngwnameserver.*site*.novell.com	CNAME	Site Post Office Agent
webaccess.*site*.novell.com	CNAME	Site GroupWise WebAccess Service

Standardizing names allows us to cheat by configuring clients in shorthand. For instance, we configure all browsers with the hostname "proxy" as the proxy server. We could hard code all of the browsers in Provo to use "proxy.provo.novell.com" but that means that if you go to Hong Kong and forget to reconfigure your browser, you'd be pulling information from a proxy server across a 56k WAN link, which sort of defeats the purpose. By simply specifying the hostname "proxy," the client will append your DHCP-assigned sub-domain name to that hostname to come up with the fully-qualified DNS name. For instance, if you were in Hong Kong, the hostname

[1] "sjf" used to stand for "San Jose Fortune," Fortune Drive being the location of our headquarters at the time they standardized our subnet names. When it was announced a few years ago that we would be moving to a new location in San Jose, I think our DNS administrator started making offerings to Cecil, the Nordic god of DNS, in hopes that they wouldn't mess up our DNS naming scheme by moving to a street who's named started with a "Q" or something silly like that. Our new headquarters is located on North ***F***irst Street, which was probably close enough to make our DNS administrator happy…but I personally think we need to change the subnet name to "***nfs***.novell.com."

"proxy" along with the DHCP-assigned domain name "hkg.novell.com" would become "proxy.hkg.novell.com." which just happens to be the name of the local proxy server.

We could jump through a number of different hoops attempting to reconfigure all of our clients/browsers/applications when someone travels to a new location, but why bother? Contrary to the current philosophy of a large portion of the industry, the simplest solution is almost always the best solution.

DHCP: Automatically Keeping Your Ducks In Rows

Anyone who has had to manage IP addresses on a large scale should already be familiar with the sanity-saving benefits of DHCP. But DHCP is especially useful with NetWare 5 because it not only simplifies the assignment and administration of client IP addresses, it also allows you to hand out Novell client configuration information such as Preferred Tree, Preferred Server, Directory and Migration Agent addresses, etc.

We have at least one DHCP server running at each site. At our major sites we use two servers (for redundancy) and split the available range of IP addresses between the two. And at those sites with multiple segments, we use forwarding to route DHCP packets to a local DHCP server.

In addition to the IP address, local DNS servers, and local subnet (*site*.novell.com), we also hand out the following NDS information via DHCP:

DHCP Option #	DHCP Option Name	Setting/Explanation
85	NDS Servers	The addresses of at least two servers that contain replicas to which the user can connect to log in.
86	NDS Tree Name	"Novell_Inc," in our case…

This allows me to fly out to Des Moines, plug my notebook in, and automatically be off and running without having to piddle around with my IP settings or Novell Client configuration.

Lease times are one aspect of DHCP that we're still tweaking. By default we have an 8 hour lease time, with renew times at 50% and 85%. This means that a workstation's DHCP address is good for at least 8 hours. After 50% of that time (4 hours) the workstation will go out and renew the lease if it can. If the workstation hasn't been able to renew the lease with the original DHCP server after 85% of the lease time has expired, it will start broadcasting for another DHCP server.

This hasn't worked for some people who expect to have the same IP address when they come back every day. In some areas of Development and Testing, where a semi-static IP address is a

Very Useful Thing, we've set the lease times anywhere from 21 to 25 hours. So, we're still experimenting with the right balance of convenience for the user vs. ease of administration[2].

Dynamic DNS: Updating The Name Of The Rose

We use Dynamic DNS to automatically create a DNS entry for a device when it is assigned an IP address from a DHCP server. We separate our static DNS entries from our dynamic ones by putting them in different DNS domains. We keep all of our devices with static addresses in the "*site*.novell.com" domain, while our Dynamic DNS entries are kept in the "dnsdhcp.*site*.novell.com" domain. This logically separates the company-wide, static devices from the local, dynamic devices and generally keeps things tidy.

We use the name of the workstation when creating the Dynamic DNS entry. For example, let's say that a person has a workstation named "BabeMagnet[3]." When a Dynamic DNS entry is created for that workstation it would be "babemagnet.dnsdhcp.*site*.novell.com." Another nice aspect of dynamic DNS is that I never really need to know the IP address of my workstation. Since the name of my notebook is "Ananas," I know that my workstation can always be found at "ananas.dnsdhcp.*site*.novell.com."

DNS & NDS: How Does Your Rose Garden Grow?

Exponentially. Implementing DNS/DHCP can add an enormous number of objects into your tree. If you were to set up your entire Class B address (approx. 65,000 addresses), once you got done with all of the in-addr.arpas, you could have well over 200,000 new objects in your tree. And if you're using Dynamic DNS those objects can get updated quite often. So if you want to optimize your NDS synchronization, you might want to seriously consider segregating DNS/DHCP data from the rest of your NDS data.

Here @ Novell, we create a DNSDHCP container below every site container and create both DNS and DHCP containers under that. Like this:

dnsdhcp.*site*.novell
dns.dnsdhcp.*site*.novell
dhcp.dnsdhcp.*site*.novell

We partition the DNS/DHCP data off in their own containers, with their own replication ring dedicated solely to DNS/DHCP. That way the synchronization of regular NDS data doesn't get

[2] If we *do* discover the right balance between convenience for the user and ease of administration, I think we'd automatically be awarded a Nobel Prize.

[3] I'm not making this one up…I saw it just the other day. And it's another good example of the how the name of an object can provide stunning insights into a person's nature. When I see a workstation with a DNS entry like "babemagnet.dnsdhcp.provo.novell.com," I can already tell you that the owner of that workstation is under 5' 10", has experienced some degree of hair loss, drives a Pontiac, and (like all magnets) is repellant to at least 50% of the population.

bogged down if 10,000 workstations all decide to release and renew their DHCP-assigned addresses at the same time.

If you're particularly anal-retentive about having a tidy tree, you could even deploy DNS/DHCP in its own separate tree. There are obviously disadvantages to this (additional hardware requirements, maintaining two, separate administrator accounts in both trees, etc.), but it's an option.

We've delegated DNS/DHCP management to administrators in the various regions around the world, but we've only given them rights to manage their own particular regions. This is a good idea in general when it comes to DNS and, with our configuration, there's little reason that the administrator in Singapore would need to change the DNS entry of the Web server at the Chicago office.

And speaking of regional administration, keep in mind that when you start the DNS/DHCP Management Console, it reads all of the sites in the locator object and then goes to each of those sites to see if you have the necessary rights to manage those objects. Needless to say, at remote sites with slow links, this can take a while. Remote administrators should take advantage of the "-s" command line option of the Management Console in order to limit the scope of the search. For instance, an administrator in Sydney can use a "-s asia-pac.novell" command line option to limit the search to only those objects in the "asia-pac.novell" container.

Acknowledgments

Nothing in this beigepaper represents original thought on my part. I couldn't have written a word without the generous help and input from everyone in Novell's IS&T Global Technical Architecture group. (I'd name them all individually but they'd probably get spammed.)

Send all comments, questions, corrections, and/or complaints to:

grettir@neticus.com

Tasty baked goods can be sent to:

Grettir Asmundarson
PRV-C-122
122 E. 1700 S.
Provo, UT 84606

And please note that Grettir Asmundarson is just a ridiculous pseudonym, so don't bother trying to call. You'll only confuse our receptionist.