Using Network Search Engine Technology to Accelerate Directory Services

Directory Services

A key part of the provisioning of any network service is a method to resolve the symbolic names used in applications and by people to the coded network and host addresses that are used within the network itself. Each application and protocol has developed their own techniques for addressing this problem and all of these techniques are collectively referred to as 'Directory Services'. The most popular of all these services is the DNS provided by the TCP/IP protocol. This paper concentrates on this particular service by way of an example as to how Network Search Engine (NSE) technology can be deployed to accelerate the provision of this service and eliminate the DNS lookup overhead inherent in all IP networks and in particular in the 'Public Internet'.

DNS is an Application Layer protocol that provides service to other Application Layer services in order to resolve the underlying transport mechanism. By providing DNS within networking devices such as routers and switches and accelerating the operation using NSE technology, it is possible to simplify the Application Layer thus reducing the load on the network and the Application Servers. This makes 'Service Based Routing' possible which in turn simplifies the tasks of 'Load Balancing' and 'Service Resilience' in Internet (e-commerce) Systems.

LDAP is an emerging standard for generic directory services and is now being deployed in a diverse range of end user directory application such as 'Directory Enabled Networking', 'NDS' and 'Windows 2000 Active Directory'. NSE technology can be applied to the development of such a generic LDAP directory.

DNS

The Domain Name Service (DNS) is an application which has the function of resolving registered host names to their respective IP addresses. Other IP enabled applications make requests to the DNS server using the DNS protocol which is transported to the DNS server as an IP packet, and the reply is then forwarded back to the requester as another IP packet.

Traditionally DNS servers are implemented as a software application residing on an industry standard computer system, and their performance is improved by caching commonly accessed addresses. In a local area environment where each client only accesses a small number of discrete hosts or services this works well, however the Internet has extended the reach of the average desktop beyond the enterprise to a wide range of hosts and services that must be resolved each time a request is made by a public DNS service. Enabling DNS resolution in network devices improves the end-to-end performance at the application layer. This sort of resolution is requested regularly by TCP/IP applications such as web browsers and email programs, where application users enter meaningful service names rather than the networking addresses of the hosts contacted. Providing DNS resolution in network devices enables a range of further technologies such as server load balancing, service resilience and service based routing that will further improve performance and enhance the "Internet Experience".

DNS resolution can be implemented in a network device to operate at wire speed, through the use of Network Search Engine technology by using a cascaded table. DNS entries are of variable length and may be up to 255 bytes long. Each entry will resolve to a single IP address. The table structure that is required to implement this design is shown later in this paper.

Network Search Engine (NSE) Technology

NSE technology uses silicon search engines to provide lookup functionality as a co-processor to an existing Microprocessor or Network Processor design. The silicon search engine implements a table (or number of tables) using Associative Memory, and therefore NSE co-processors are able to provide the results of a query to the host processor a few CPU cycles after the request has been made. The LNI7040/LNI8010 co-processor from Cypress Semiconductor Corporation is capable of providing one result every three cycles for a 68bit wide record, at a clock speed of up to 100MHz. The alternative to using an NSE co-processor is to implement the lookup tables required in software. NSE provides the following benefits over software based solutions:

- Faster and less CPU cycles until a match is achieved
- Offloads processing – the host CPU is free to perform other tasks
- Deterministic, i.e, performance is constant – even as table sizes grow past 1M records!

- Simpler Design – no complex software algorithms to debug

Logically, an NSE co-processor can be considered as made up of a number of tables of records with the configured width. The LNI7040 NSE device from Cypress Semiconductor may be configured to create tables either 36, 72, 144, or 288 bits wide. Each of the tables can be considered to be independent and the format of the record within the table is determined by the controlling software. The record structure described in this paper is only an example for this application and should be modified to suit the precise requirements of the user.

Each record in the tables also has an associated mask record, the same width as the record. This can be used to mark any bit of the record as 'don't care' for the purpose of the match process. This allows the tables to be further divided into rules that are generic and those that are more specific.

In an associative memory search it is only possible for one entry in the table to be returned as the match. Due to the mask capabilities it is possible for multiple matches to occur when the search is carried out. This is handled by the built in priority encoder which ensures that the match with the lowest address is returned in the case of multiple matches. If the tables are organised such that the more specific rules occupy the lower addresses in the table, then this process ensures that specific rules override more generic entries.

Using NSE to Implement DNS in Network Devices

Cascaded Table Technique

The widest table that can be supported by current NSE co-processors is 288 bits. This allows for up to 36 ASCII characters to be encoded. While most domain names that are registered are shorter than 36 characters, it is necessary to support longer strings, in particular in countries where double-byte character sets are used. In order to implement a flexible lookup system that is able to provide name resolution for short names in one search operation, a table is implemented with a variable record structure as shown in figure 1.

Figure 1

NSE record*				**Related SSRAM record**		
Level 8 bits	Entry type 8 bits	Record 34 bytes		Resolved / extend flag	IP Address / Hash	Spare
0x00		34 characters match data		1 bit	32 bits	31 bits
0x01 – 0xFF		Hash 32 bits	Additional 30 characters match data			

*The NSE is configured to provide a 288-bit wide search table.

The software disassembles the incoming DNS request packet and extracts the lookup record. At this point the software calculates the length of the lookup record and from that works out the depth of searching that is required. (This table implementation will allow for search strings of up to 7682 bytes (255*30 + 32). This is well in excess of the requirements of a DNS server where the name length is limited to 255 characters.)

The first match is always preformed with the first 34 characters of the lookup record. If the lookup record is less than 34 characters, the returned value will either be a miss, or the required IP address.

If the lookup record is longer than 32 characters, the returned value will be either a miss, or the hash value that needs to be used in the level 1 search to resolve the lookup record. If a value is returned, then this is assembled into a search string made up of the first byte as 0x01, the next four bytes the hash value from the first lookup, and the next 30 characters of the lookup record.

This process is continued until a miss is encountered, or the previously calculated query depth has been reached (which is then considered to be a miss) or the lookup record is resolved to an IP address.

The entry type field allows separate tables to be implemented for different protocols such as IPV6, etc.

Reverse Lookup

DNS functionality also requires the ability to resolve the registered name(s) for services from the IP address to which they are associated. The lookup function can be implemented using a table structure as shown in figure 2. This can be co-located in the NSE co-processor used for the Cascaded Table lookup due to the multi-table capability of the LNI70XX devices.

Figure 2

NSE record*	**Related SSRAM record**
IP Address	Pointer for host list
32 bits	64 bits

*The NSE is configured to provide a 36-bit wide search table.

The returned value from the lookup is then used as a pointer to the base address of the list of hosts associated to the IP address. This list is held in the system memory.

Table Management

The NSE must be configured into two blocks to implement the tables, one 36 bits wide and the other 288 bits wide. These tables can then be populated through the 'add record' process, as new DNS entries are added. DNS entries where the host name is longer than 34 characters need to be split into segments, the first of which is 34 characters and is added to the table with the Level set to 0, and the remainder 30 characters with their Level value set in sequence from 1 upwards. The hash value that is stored in the SSRAM record is simply a count of the number of entries that have been added at that level so far. This would be created as an array of 256 host variables, one of which is incremented each time a new record is added with a particular level. If a duplicate entry is encountered at any level, then the hash value should be read for the existing record and used in the following record. There is then no need to add the duplicate record as the existing record will provide the required lookup at the next level.

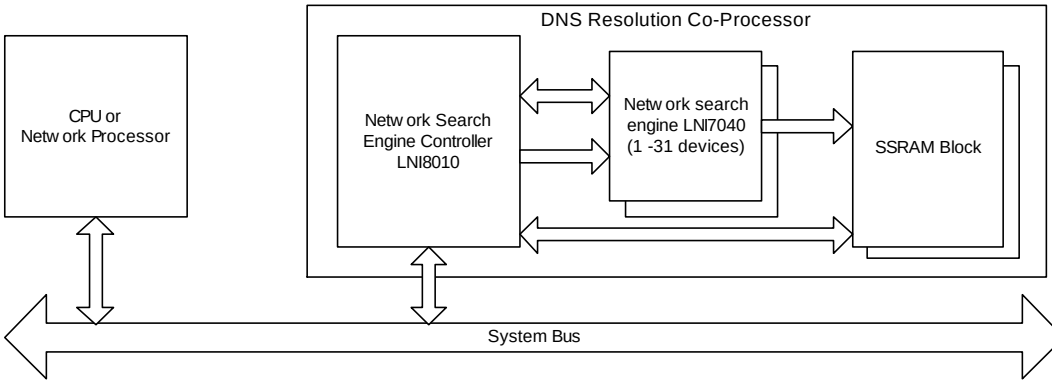
As duplicate records are not allowed in this design, there is no concept of order within the tables, new entries can simply be added at the first available free space in the table.

Implementation

Architecture

To implement a 'DNS Resolution Co-Processor', Network Search Engine technology (such as the LNI7040 and LNI8010 products from Cypress Semiconductor) may be deployed, either in conjunction with a Network Processor, or with a standard RISC microprocessor. A typical system architecture is shown in Figure 3.

Figure 3



Performance

In order to determine the resolution performance that the "DNS Resolution Co-Processor" will achieve, it is necessary to look at the query throughput that can be achieved by the design. The LNI8010 Network Co-Processor may be connected to a 64 bit, 100MHz system bus. Three bus cycles are required to perform a 36-bit search, and six cycles are required to perform the 288-bit search.

Figure 4

Operation	36 bit search	288 bit search
Write Search Word	10ns	40ns
Write Command	10ns	10ns
Read Result Word	10ns	10ns
Total response time	**30ns**	**60ns**

Resolution is performed for each DNS request received and is resolved in the form of a returned packet containing the required address. As this DNS co-processor is implemented in the core of the network device, multiple packets may be received simultaneously and will need to be queued. The majority of DNS resolutions are forwarded and will therefore require 70ns per search cycle. A statistical sample of the

registered domain names has been used to provide an estimate of the average number of search cycles required.

Figure 5

Name Length	% of Names	Cycles Required	Period
<34	92	1	60ns
35<length<60	7	2	120ns
61<	1	3	180ns

This provides us with an average resolution rate of 15 million resolutions per second.

The number of registered hosts in the public Internet is continually growing and passed one million in 1996. The public DNS system consists of a number of distributed databases that are locally managed at the ISP or organization level. These services share information and have part of their resolution table dedicated to local hosts that they manage, and part dedicated to provide cached lookup to improve performance when remote host lookups are requested. NSE systems can provide tables in excess of one million records which is easily capable of supporting the requirements of even the largest ISP as well as providing adequate space to cache entries from other ISPs for resolution with local clients.

Each round trip between a client application using DNS and the host server requires at least two packet trips. One to resolve the IP address of the host and another to request the required page or data. If the host cannot be resolved locally then further requests need to be made to the other known root DNS resolvers. Given the capabilities of NSE based DNS, a protocol could be devised where the destination IP address portion of the data request packet is filled in on the fly by the DNS co-processor by interpreting the data held in a DNS request wrapper. End to end performance of the network layer is enhanced as only one round trip is required per data request.

This implements a variant of service based routing where the requesting client does not need to have knowledge of the network address of the host contacted as this is resolved by a device within the network itself. This technology provides a number of operational services to such a network as it enables packet to be routed to different hosts based on loading characteristics to provide dynamic load balancing. Packets could be routed to the same host but through different network paths, to provide network service resilience and load balancing, and transparent failover could be provided to e-commerce applications without the client being affected in the event of maintenance or failure of a particular host.

LDAP

The Lightweight Directory Access Protocol (LDAP) provides a standard for the access to a directory entry resolution service. To implement LDAP, the directory service needs to be configurable and be able to service request based on a hierarchy of keys and their record values.

The cascaded NSE architecture that was deployed in the DNS co-processor implementation can be extended to provide a generic lookup facility that could be used to implement and LDAP compliant directory service. As the resolution process is implemented in hardware the resolution performance will exceed existing solutions by many orders of magnitude.

Conclusion

Directory Services are a key component of all Network Systems in use, and the most widespread is the Domain Name Service implemented using the IP protocol. Network Search Engine technology can be employed to accelerate the resolution process performed by Directory Services by many orders of magnitude over traditional software techniques. These performance improvements can be employed to enable a number of emerging technologies such as 'Service Based Routing" and "Server Load Balancing", by providing wire speed name resolution within Networking Devices. NSE based solutions are simpler to implement and provide a cost effective alternative to traditional designs of networking devices as the same hardware co-processor can be implemented to provide a number of different network services without degradation in performance through the use of multi-table databases.