**I CLAIM:**

1. A method for achieving seamless transition of IPv4 addresses to IPv6 addresses, the method comprising:

deploying, by an Internet Service Provider (ISP), at least an IPv6 address holding an IPv4 address and the IPv6 address simultaneously for allocating at least the IPv4 address or the IPv6 address to at least one customer in the network, wherein the IPv6 address maintains an IPv4 address register holding the IPv4 address along with an IPv6 address register holding the IPv6 address, and a software library call available for the IPv4 address is configured to utilize the IPv6 address register;

migrating at least an application, used by the customer, utilizing the software library call available for the IPv4 address to utilize the software library call available for the IPv6 address;

enabling, after migration, transition of the IPv4 address to the IPv6 address and routing of the data and calls based on the transition achieved.

2. The method as claimed in claim 1, wherein the software library call for IPv4 address comprises "gethostbyaddress32" call, and the software library call for the IPv6 address comprises "gethostbyaddress128" call.

3. The method as claimed in claim 2, wherein "gethostbyaddress128" call and "gethostbyaddress32" call utilizes the IPv6 register.

4. The method as claimed in claim 1, wherein upon migration, the application is configured to use both the IPv4 register and the IPv6 register.

5. The method as claimed in claim 1, wherein upon migration, the IPv4 address is available for re-use and/or reallocation to at least a new customer available in the network.

6. The method as claimed in claim 1, wherein upon migration, further comprises using both "gethostbyaddress32" (IPv4) and "gethostbyaddress128" (IPv6) library call.

7. The method as claimed in claim 1, wherein the IPv6 address holds the IPv4 address register and the IPv6 address register simultaneously.

8. The method as claimed in claim 1, further comprises routing the data / calls based on the transition achieved.

9. The method as claimed in claim 1, before migrating, the method further comprises:

differentiating which applications or databases hosted by the customer based on utilization of the IPv4 software library call or the IPv6 software library call to service calls to provide a host name by address; and

providing a list of applications or databases to the customer that highlight which applications or databases may be transitioned to use the IPv6 address.

10. A method for achieving seamless transition of IPv4 addresses to IPv6 addresses, the method comprising:

detecting, by an Internet Service Provider (ISP), one or more IPv4 addresses and one or more IPv6 addresses available for allocation to at least one customer in the network;

maintaining an IPv4 address register and an IPv6 address register for the IPv4 addresses and the IPv6 addresses detected;

allocating the IPv4 address and the IPv6 address at a defined portion of an address field of an IPv6 packet header;

matching the IPv4 address registers and IPv6 address registers, available on the network at the same time, based on a software library call available on the IPv6 address thereby enabling the IPv4 address registers to use the IPv6 address;

migrating at least an application utilizing a software library call available in the IPv4 address to utilize the IPv6 address and a software library call available in the

IPv6 address, to achieve transition of the IPv4 addresses to the IPv6 address; and routing the data and/or calls based on the transition achieved.

11. The method of claim 10, wherein, matching further comprises allocating a software library call available in the IPv4 address to the lPv6 address register matched and thereby removing the IPv4 address register from the IPv4 address.

12. The method of claim 10, wherein matching the IPv4 address registers and IPv6 address registers is based on "gethostbyname32" call.

13. The method of claim 10, wherein, migrating further comprises the software library call, "gethostbyaddress32" call, available in the IPv4 address to utilize the IPv6 address and the software library call, "gethostbyaddress 128" call available in the IPv6 address.

14. The method of claim 10, wherein, migrating further comprises allocating, based at least one a routing information, the software library call, "gethostbyaddress32" call, of the IPv4 address to the lPv6 address register, and thereby de-allocating the IPv4 address register.

15. The method of claim 10, wherein, migrating further comprises recompiling at least a source code of the application to utilize the software library call, "gethostbyaddress128" call, available on the IPv6 address.

16. The method of claim 10, further comprises routing based on at least a routing information provided by broadcasting permissible routes to at least an autonomous system (AS) numbers, wherein the permissible routes utilizes a private instance of a Resource Key Public Infrastructure to broadcast the routing information pertaining to the transition achieved to a Regional Internet Registrar.

17. The method of claim 10, wherein, migrating the application to utilize the software library call available on the IPv6 address, comprises:

recompiling a source code of the application to the software library call available on the IPv6 address provided by an operating system (OS); and substituting using a direct binary substitution.

18. The method of claim 10, further comprises:

using the IPv4 software library call to provide the host name by address for applications that only support IPv4 addressing; and

using the IPv6 software library call to provide the host name by address for applications that support IPv6 addressing.

19. The method as claimed in claim 10, before migrating, the method further comprises:

differentiating which applications or databases hosted by the customer based on utilization of the IPv4 software library call or the IPv6 software library call to service calls to provide a host name by address; and

providing a list of applications or databases to the customer that highlight which applications or databases may be transitioned to use the IPv6 address.

20. A method for achieving seamless transition of IPv4 addresses to IPv6-addresses, the method comprising:

allocating one or more IPv4 addresses and one or more IPv6 addresses at same time with a gethostbyname32 pointing to an IPv4 register and a gethostbyname128 pointing to an IPv6 register;

changing the gethostbyname32 library calls to specifically access the IPv6 register and deallocating the IPv4 register; and

migrating at least an application utilizing the gethostbyname32, a software library call, to utilize the gethostbyname128, a software library call, to achieve transition of the IPv4 addresses to the IPv6 addresses.

21. The method as claimed in claim 20, further comprises

allocating the software library call to the IPv4 address register and the IPv6 address register by pointing a gethostbyname32, the software library call, to the IPv4 register, and agethostbyname128, the software library call, to the IPv6 register.

22. The method as claimed in claim 20, wherein after deallocating the IPv4 register, the method further comprises reusing the lPv4 address associated with the IPv4 deallocated.

23. The method as claimed in claim 20, further comprise routing the data / calls based on the transition achieved.

24. The method as claimed in claim 20, before migrating, the method further comprises:

differentiating which applications or databases hosted by the customer based on utilization of the IPv4 software library call or the IPv6 software library call to service calls to provide a host name by address; and

providing a list of applications or databases to the customer that highlight which applications or databases may be transitioned to use the IPv6 address.

25. The method as claimed in claim 20, further comprises ceasing to utilize the IPv4 register after the software library has migrated to use the IPv6 address, wherein the ceasing includes re-allocating an IPv4 software library call to use the IPv4 address located within the IPv6 register.