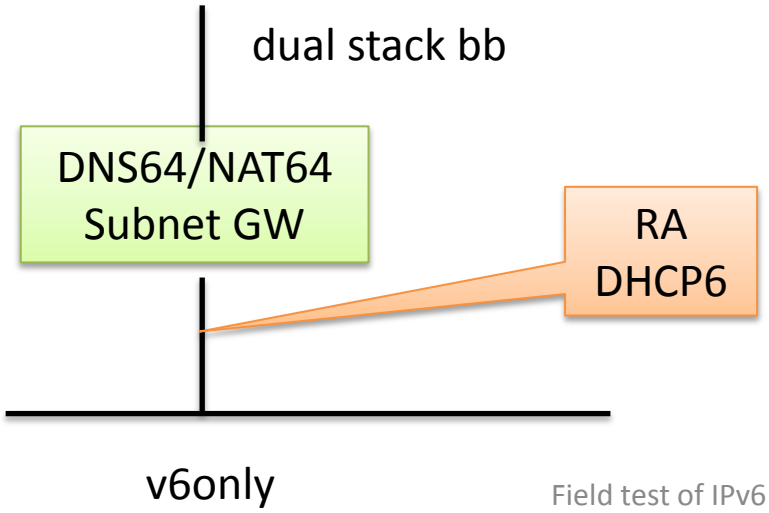
draft-hiromi-sunset4-ipv4-termination-00.txt

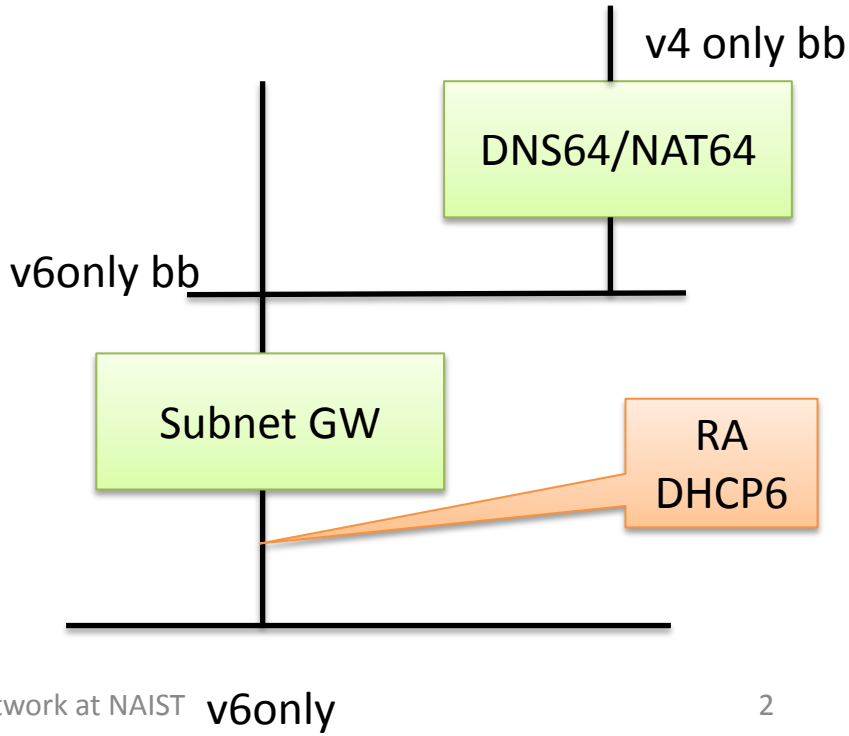
draft-hiromi-sunset4-ipv4-termination-00.txt

Ruri Hiromi, Hiroaki Hazeyama,
Atsushi Onoe, Osamu Nakamura

IPv6 only with DNS64/NAT64

- DNS64/NAT64: Map all IPv4 on Internet into IPv6 RA: Enable Other Config (for DHCP6)
- DHCP6: Distribute DNS64 address
- DHCP4: No DHCP4 running!





Assumption

- People run their devices in dual stack mode
  - Some of them fully support DHCP6 features
  - Some of them support only RA
  - Some of them have happy eyeball features

- People, except for geeks, cannot set up IPv6 configuration in manual
  - So, people usually let OSes set up automatic configuration

Tested OSes

- Windows Vista, 7, 8, XP
- MacOS X (Leopard, Snow Leopard, Lion, Mountain Lion)
- Apple iOS 4, 5
- Android 2.3.4, 4.1

- All of them can get IPv6 address through RA

Issues on OS in IPv6 only network

- Windows Vista, 7, 8
  - No problem except for waiting time out of DHCP4
- Windows XP, MacOS X (~Snow Leopard)
  - Need manual settings of DNS
  - Except for geeks, people cannot set up it
- MacOS X
  - Waiting for IPv4 connection timeout for Chrome, Firefox. (Safari has no problem).
  - Many Mac OS users met nightmare !
- iOS5
  - In some 3G carrier, network Settings was not completed ,and continuously failed due to retry
  - In other 3G carrier, there are no problem
- Android 2.3.4, 4.1
  - DNS was not usable (DNS queries employed IPv4 function).
  - No manual configuration was available.

Issues in detail

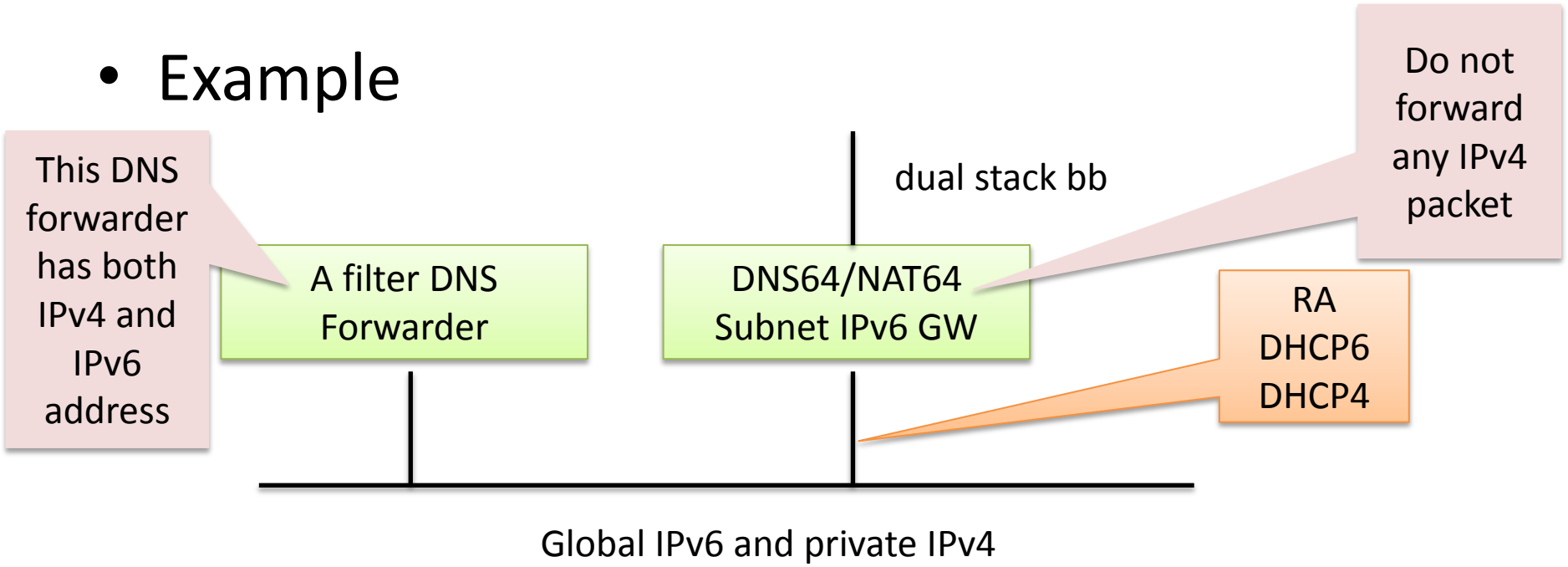
- DNS is not available (WinXP, MacOS Snow Leopard, Android)
  - DNS information should be got via DHCP6, but these Oses did not have DHCP6 function.
  - Android 2.x cannot be configured to use DNS over IPv6 even in manual configuration.
- Network Settings is not completed in IPv6 only network (iOS5 in Some 3G carrier)
  - “Network Settings” will be completed only if IPv4 address, IPv4 router, and IPv4 DNS can be retrieved via DHCPv4 or manually configured all of these 3.
- Waiting for IPv4 connection timeout (MacOS X)
  - MacOS X implements RFC3927 3.3 (Interaction with Hosts with Routable Addresses), which assumes all IPv4 address are on-link at Link-Local configuration.
  - MacOS X’s implementation of getaddrinfo() prefer IPv4 over IPv6, while Happy-Eyeball is implemented on Objective-C API. So Safari has no timeout problem, but Chrome, Firefox and most other application uses C API faced the problem.

DNS forwarder with “A” filter

- Just convert “AAAA” filter to “A” filter on bind
  - Quick hack patch by Jinmei of ISC
  - This DNS forwarder forwards every queries to DNS64 and filters out A records from replied records
  - We provided this DNS forwarder through DHCP4 and stateless DHCP6 in 3rd exp (September 2012) and in 4th exp (March 2013) of WIDE camp
    - DHCP4 does not tell IPv4 gateway or an IPv4 gateway does not forward any IPv4 packets
- “A” filter DNS forwarder can cease troubles on OS
- This DNS forwarder should be run with DHCP4 that provides only private IPv4 address
  - See Appendix

DNS forwarder with “A” filter

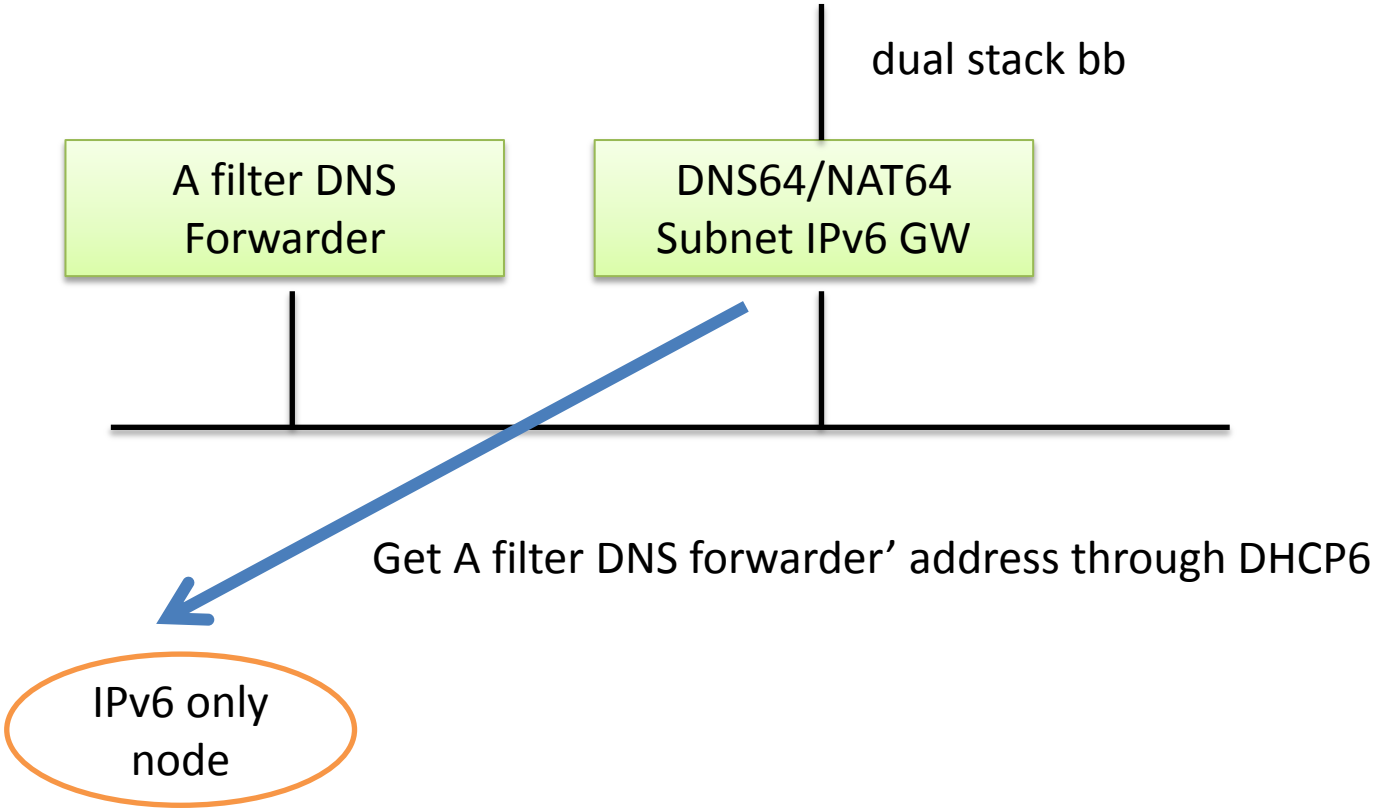
- Example



Note: Both DHCP4 and DHCP6 announce DNS forwarder’s IP(v4 or v6) address

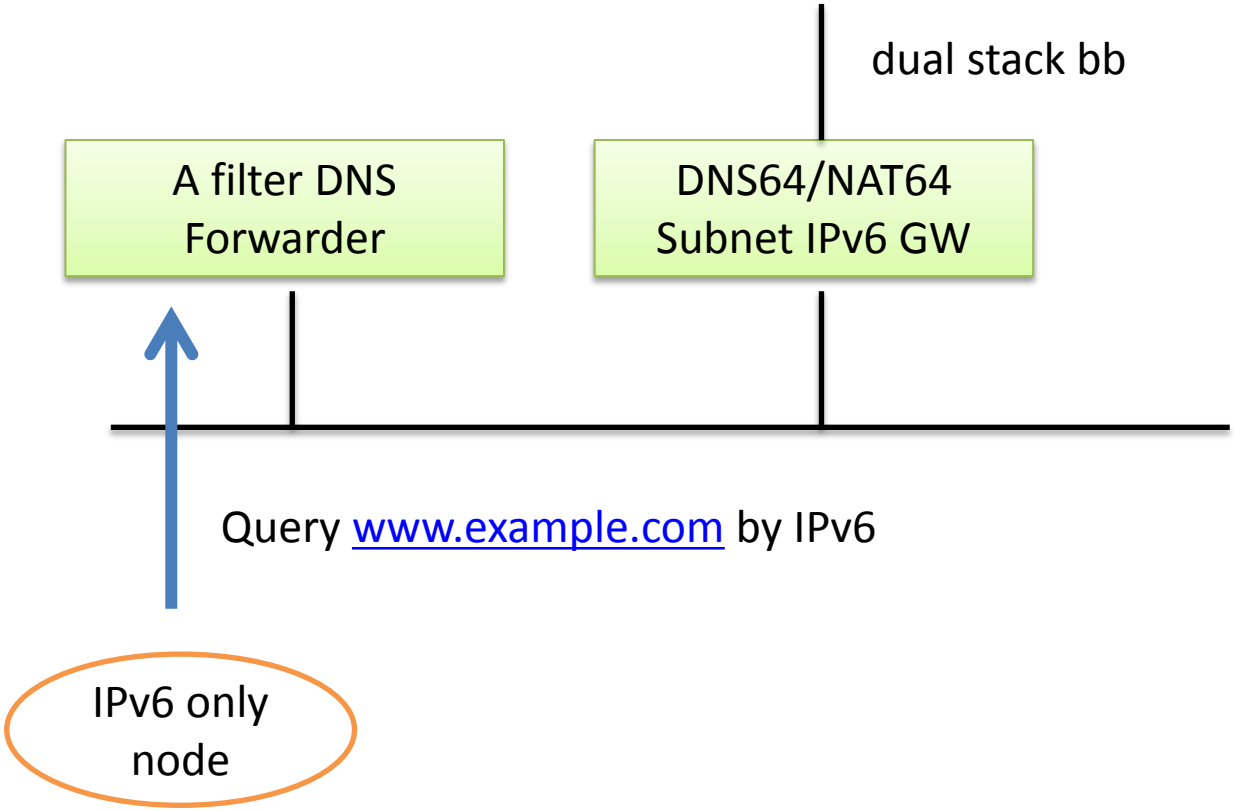
DNS forwarder with “A” filter

- IPv6 only node



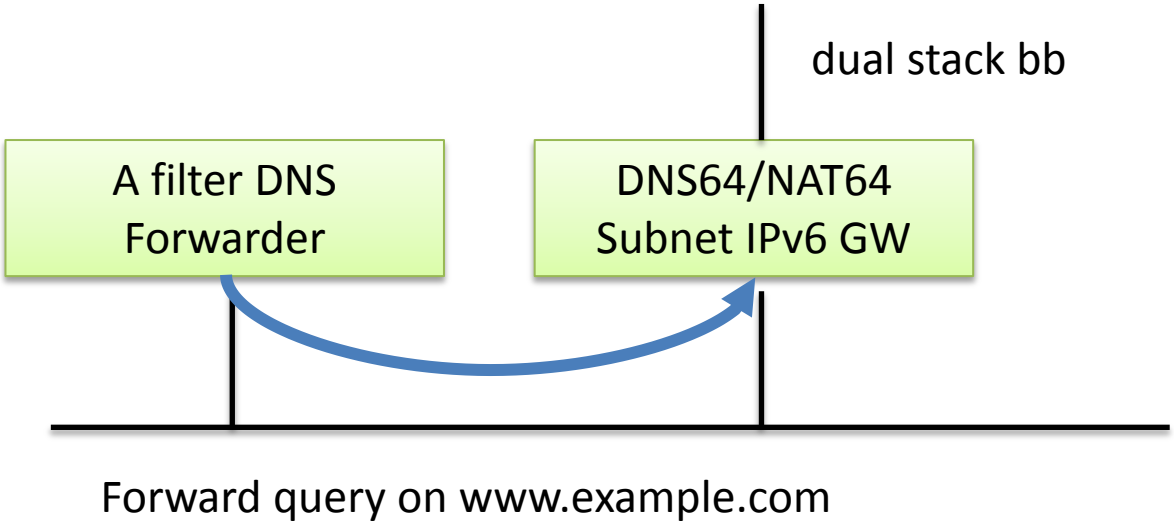
DNS forwarder with “A” filter

- IPv6 only node



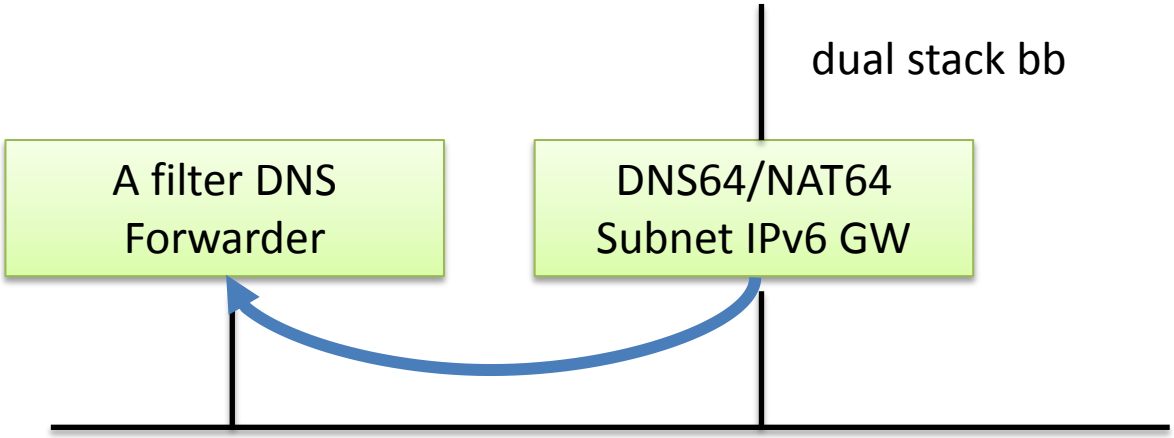
DNS forwarder with “A” filter

- IPv6 only node



DNS forwarder with “A” filter

- IPv6 only node

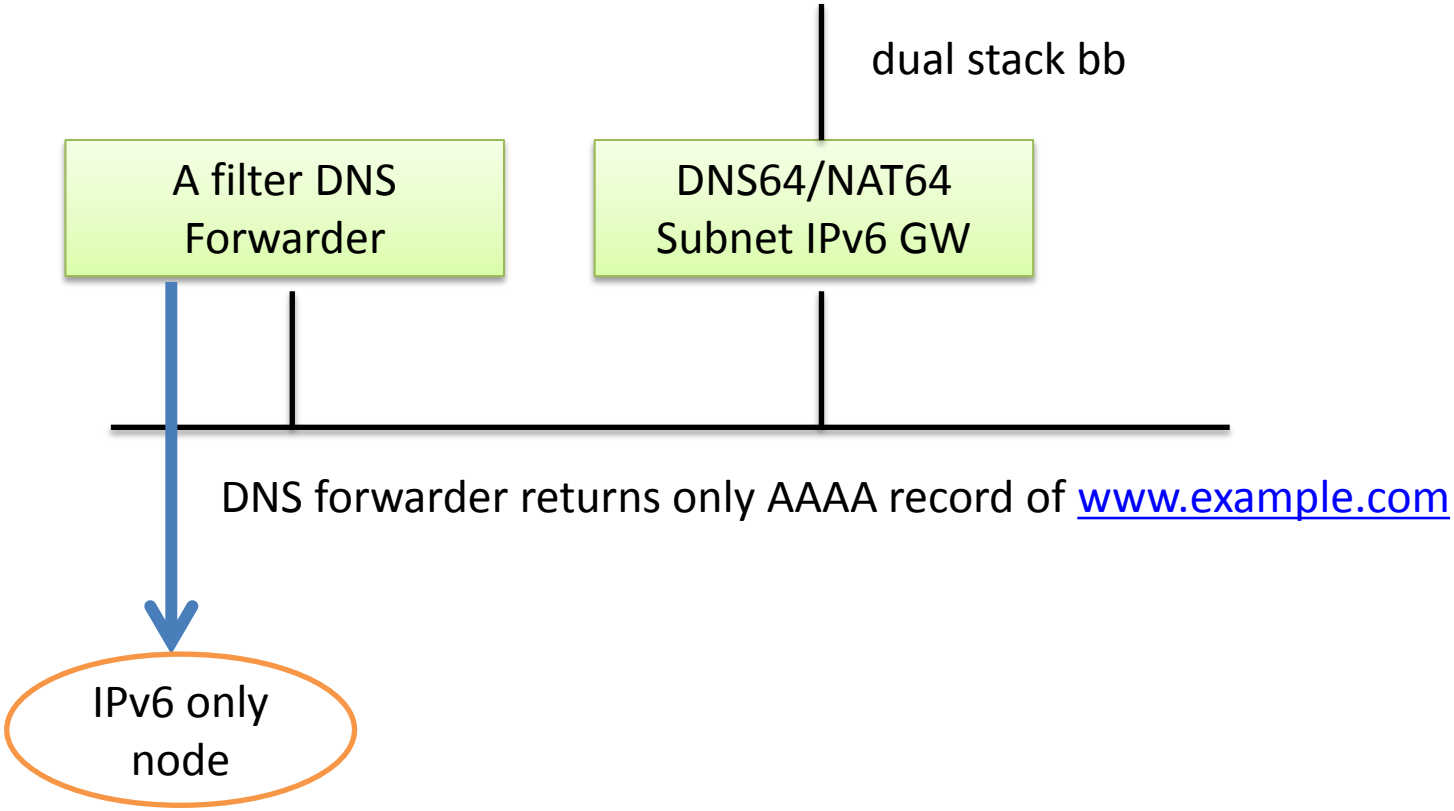


DNS64 may return A and AAAA records on www.example.com
The AAAA record may be NAT64 prefix-mapped IPv4 address

IPv6 only node

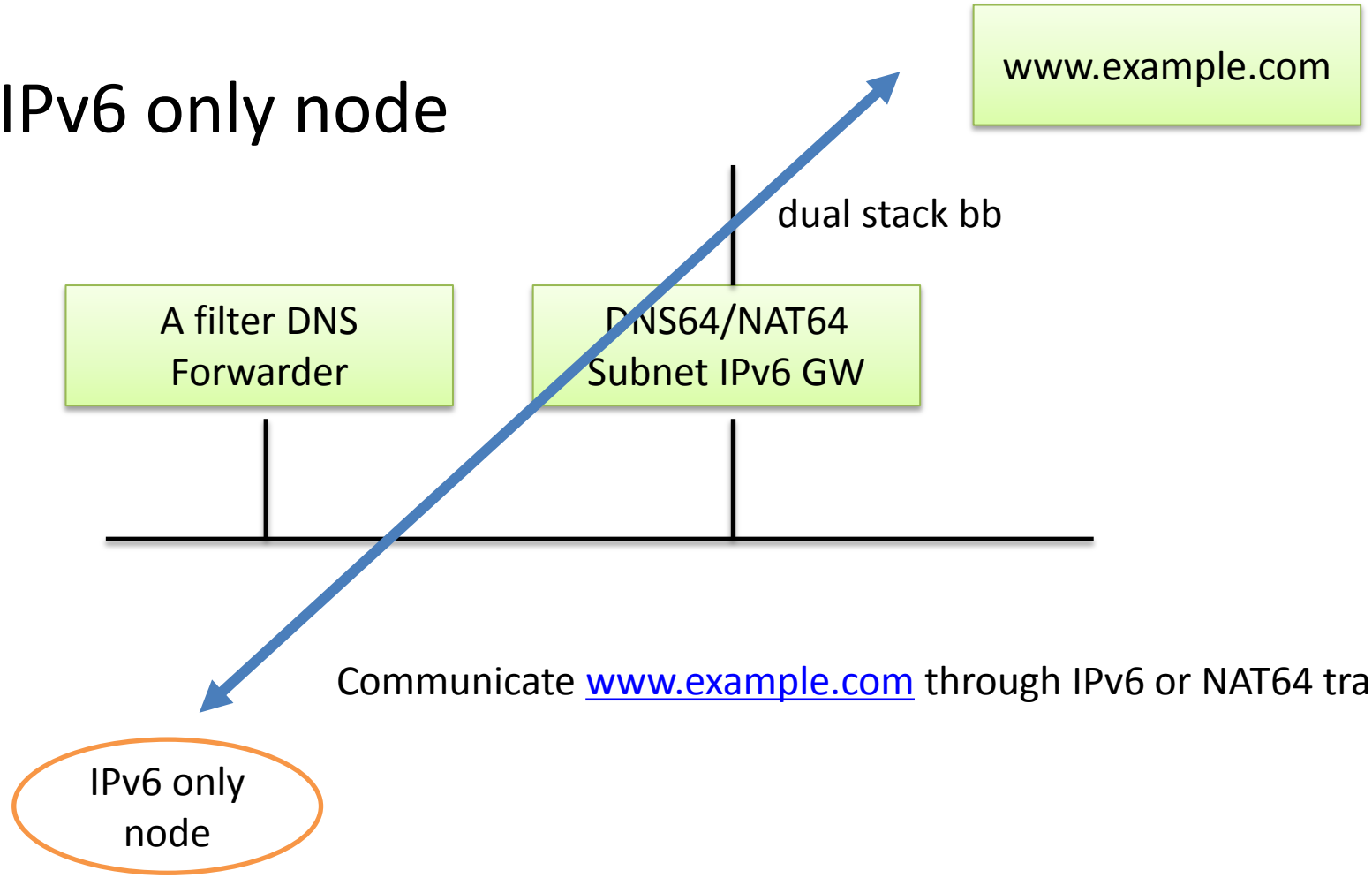
DNS forwarder with “A” filter

- IPv6 only node



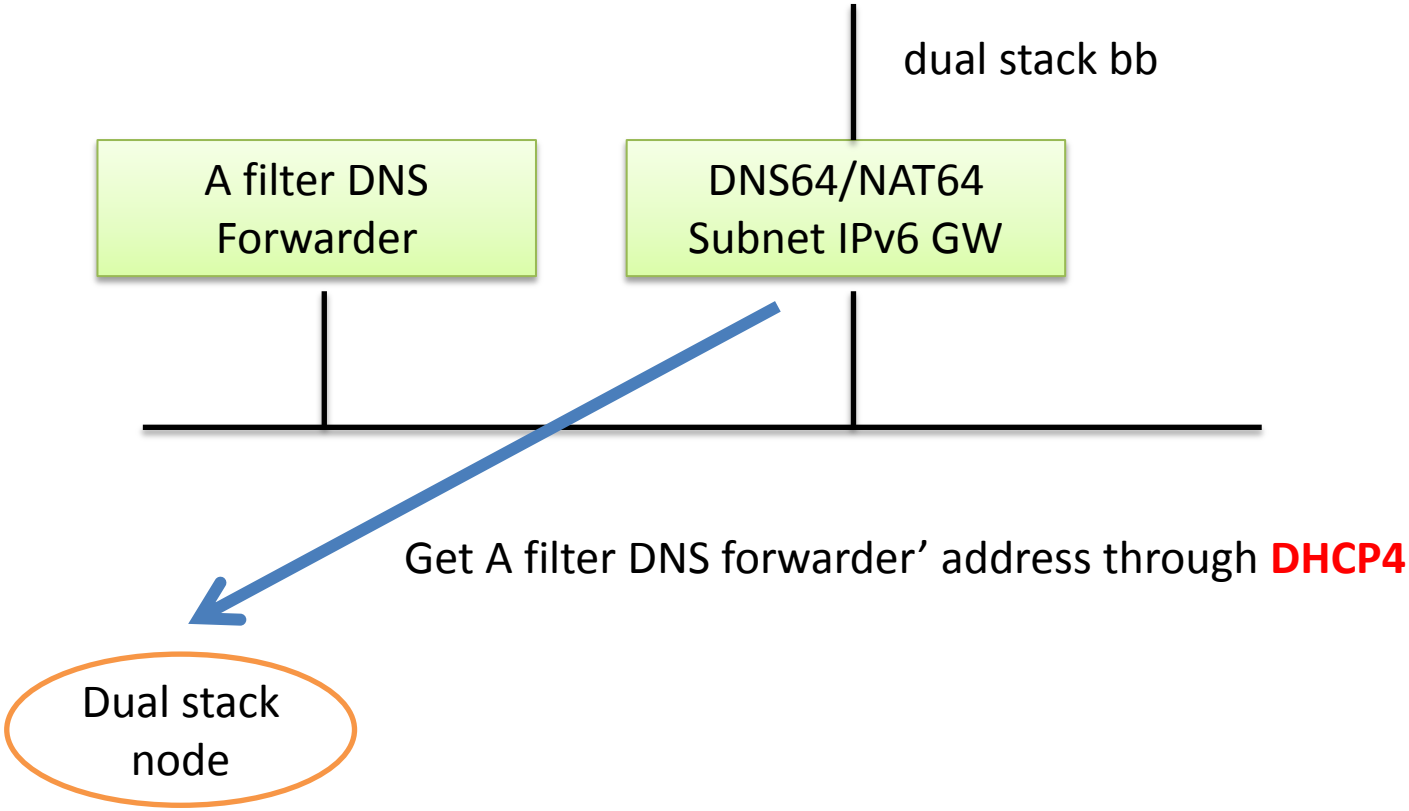
DNS forwarder with “A” filter

- IPv6 only node



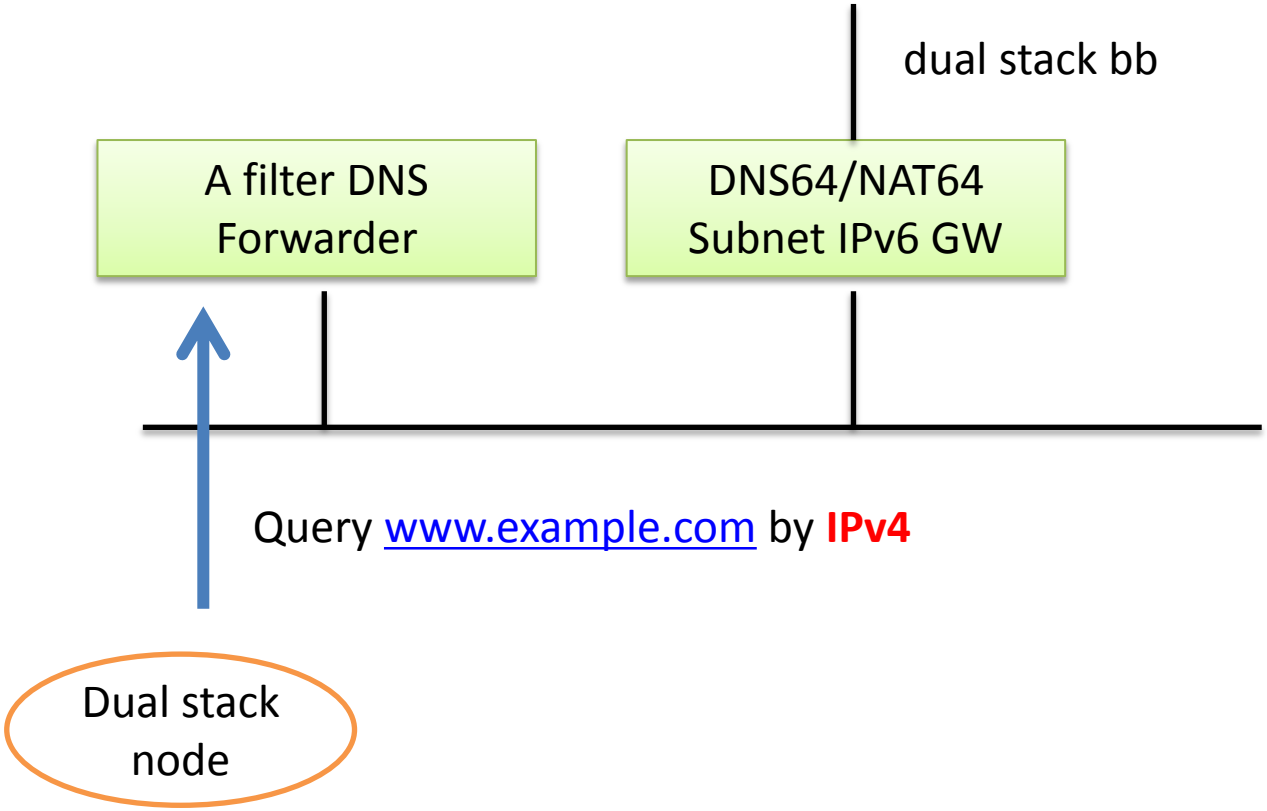
DNS forwarder with “A” filter

- Dual stack node



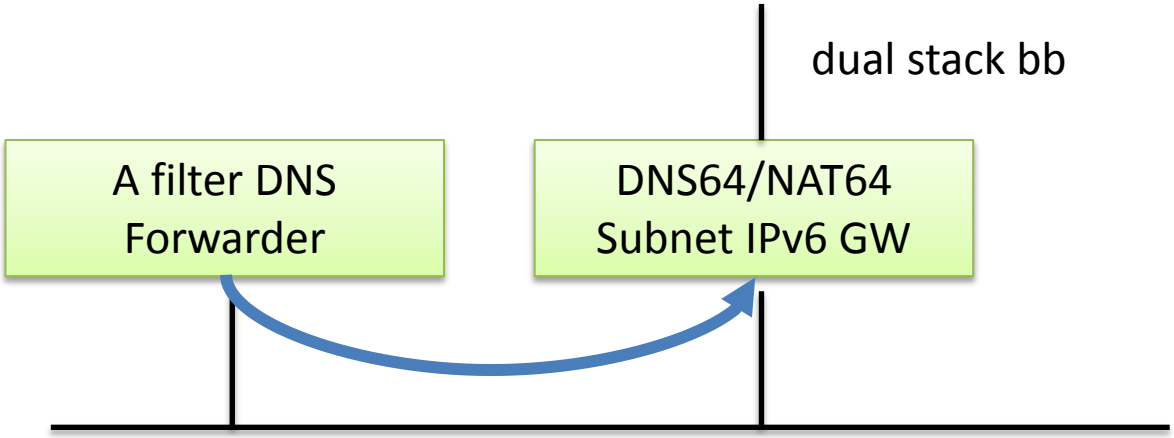
DNS forwarder with “A” filter

- Dual stack node

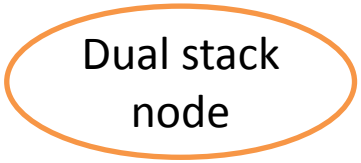


DNS forwarder with “A” filter

- Dual stack node

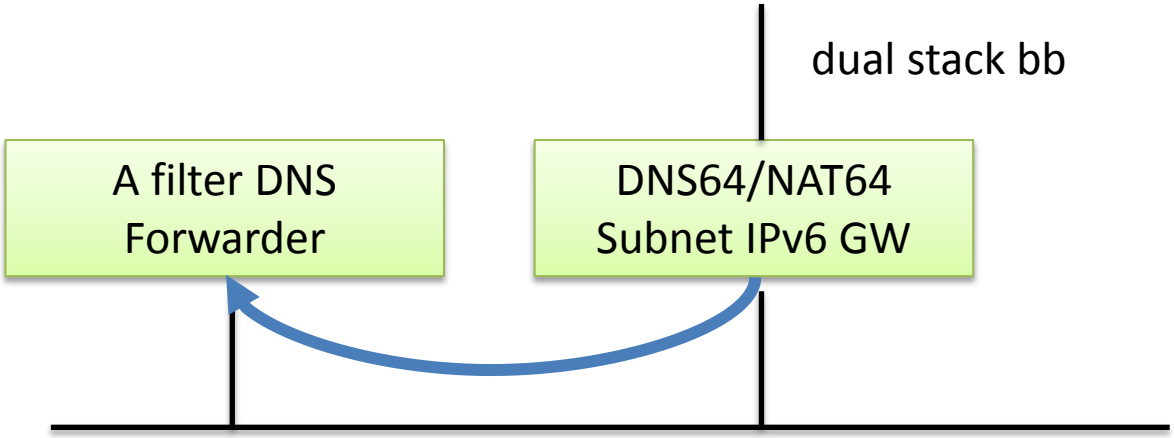


Forward query on www.example.com to DNS64 or other cache servers



DNS forwarder with “A” filter

- Dual stack node

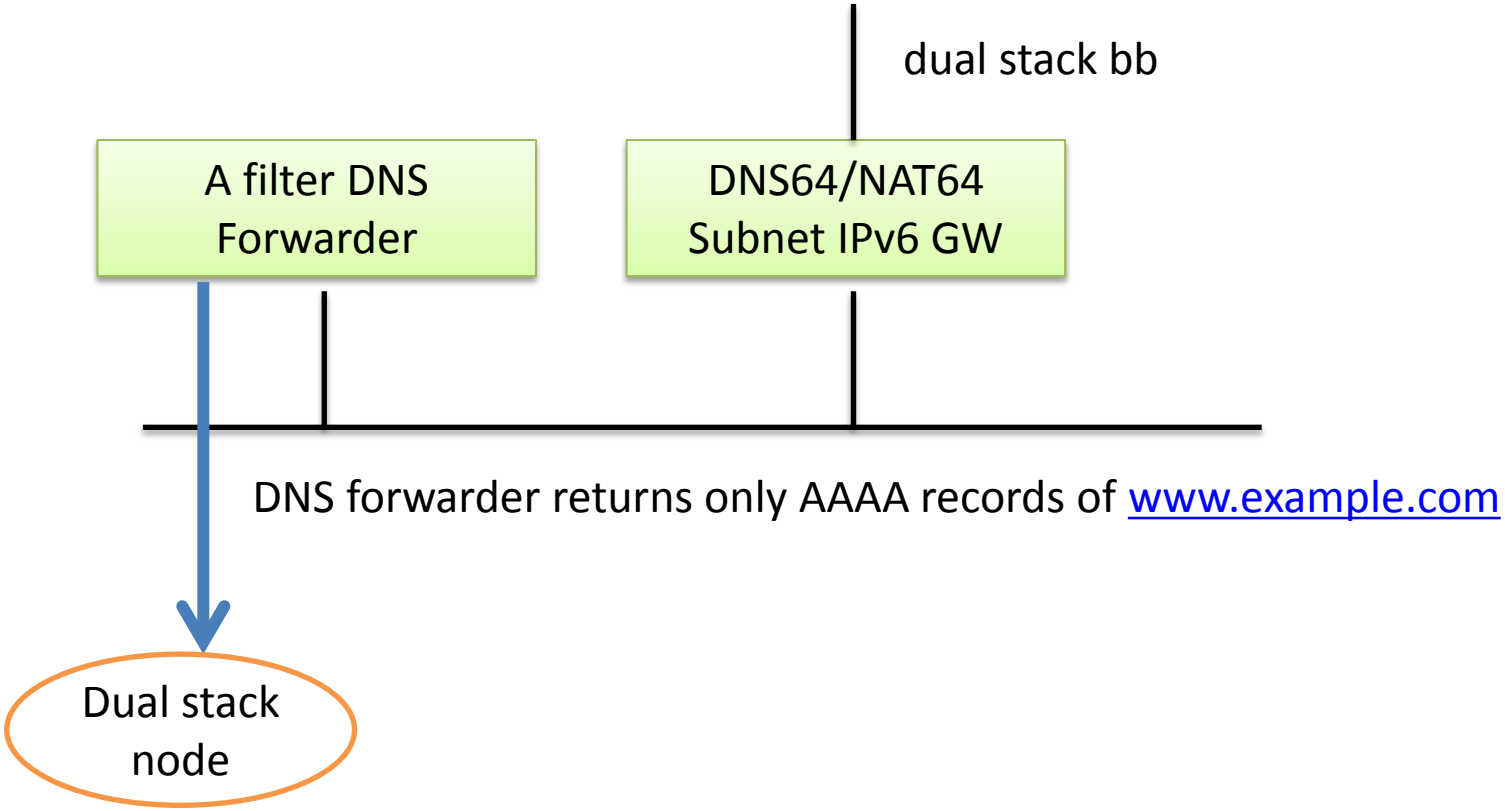


DNS64 may return A and AAAA records on www.example.com
The AAAA record may be NAT64 prefix-mapped IPv4 address

Dual stack
node

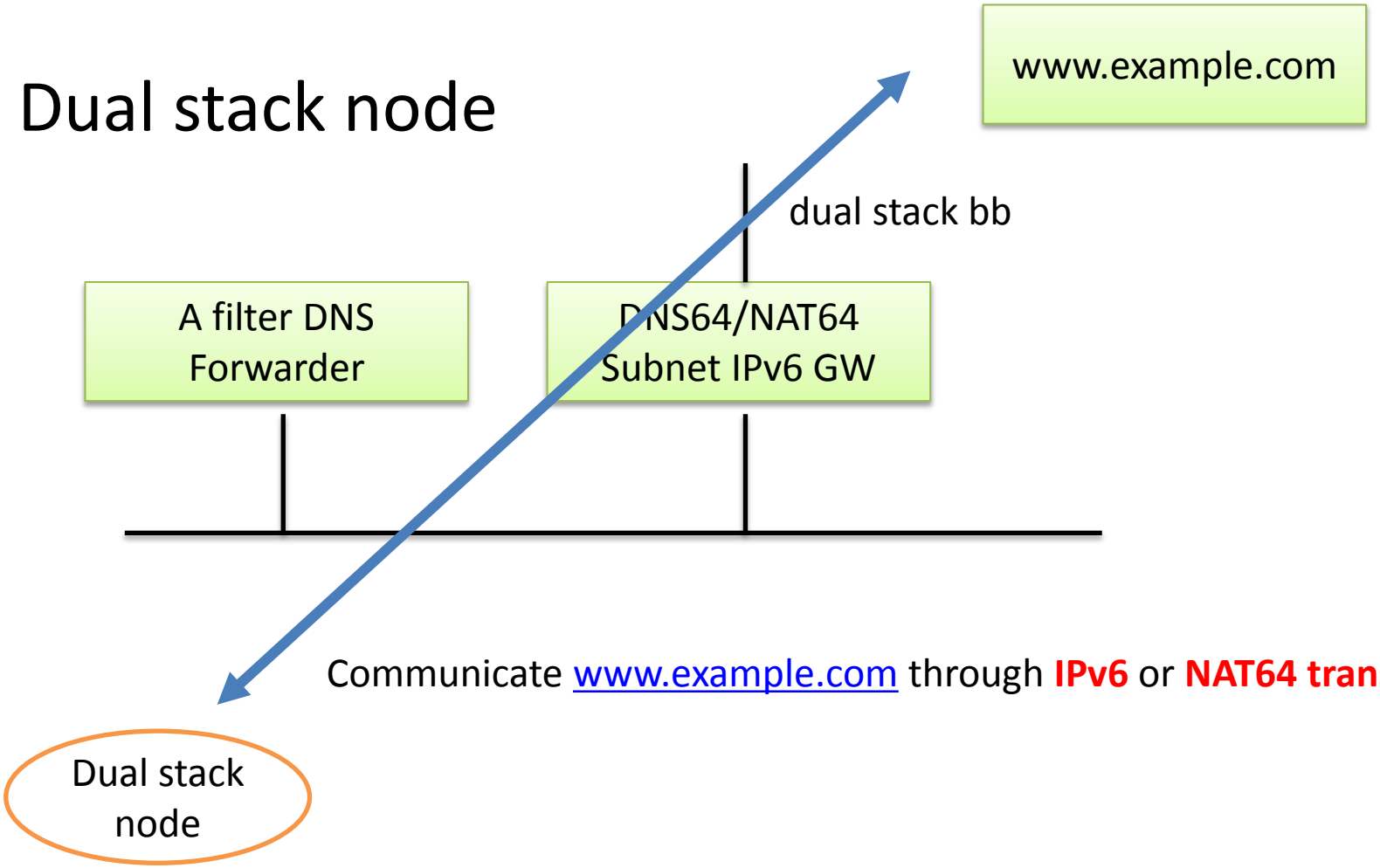
DNS forwarder with “A” filter

- Dual stack node



DNS forwarder with “A” filter

- Dual stack node



DNS forwarder with “A” filter

- Benefit
  - Enable dual stack in name resolution, however, force single stack (v6 only) in IP forwarding
  - Most OSes can reach happy IPv6 only life
    - if WiFi network is stable

BIND 9.9.2P1 Patch

- See appendix of a WIDE technical report
  - http://member.wide.ad.jp/tr/wide-tr-live-with-ipv6-wg-ipv6-depolyment-in-japan-01.pdf

Issues on apps in IPv6 only networks

- We cannot use IPv4 literals
- We cannot use IPv4 only applications
- Inappropriate DNS mentioned in RFC4074
  - ServFail stops DNS64's address mapping

Is WG interested in this proposal?

- Is our experience and “A” filter DNS forwarder suited for WG draft on sunset4 ?

Appendix: Field test of IPv6 only network at NAIST

KAKIUCHI Masatoshi

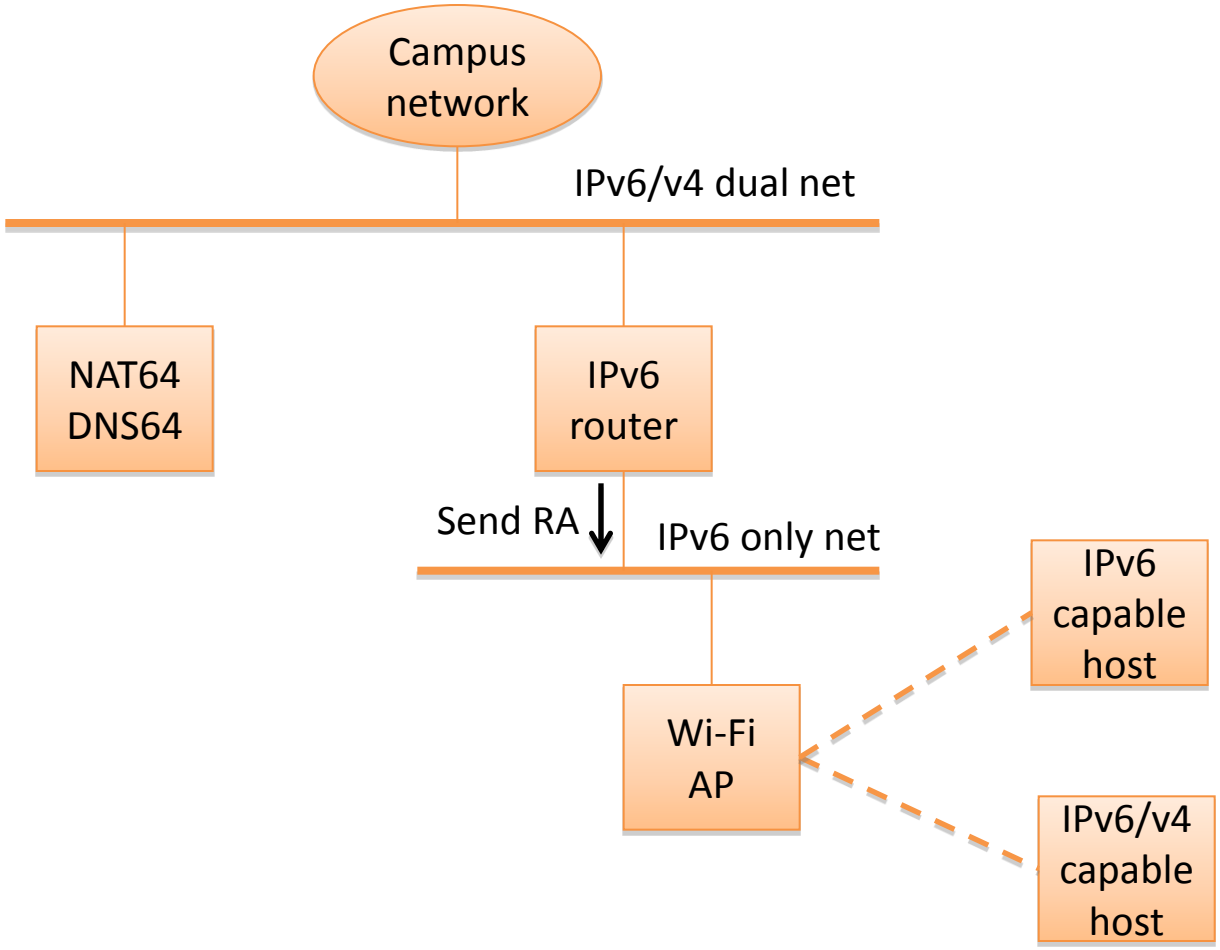
HAZEYAMA Hiroaki

Nara Institute of Science and Technology

Try work around in campus wi-fi nework

- We tried IPv6 only network settings according to draft-hazeyama-widecamp-ipv6-only-experience-02
  - Phase 1
    - Just placed DNS64/NAT64 in our campus wi-fi network
  - Phase 2 (current)
    - Added DHCP4 server
      - Same as experiment 1 setting of draft-hazeyama-02
  - Phase 3 (under construction)
    - DNS forwarder with A filter solution

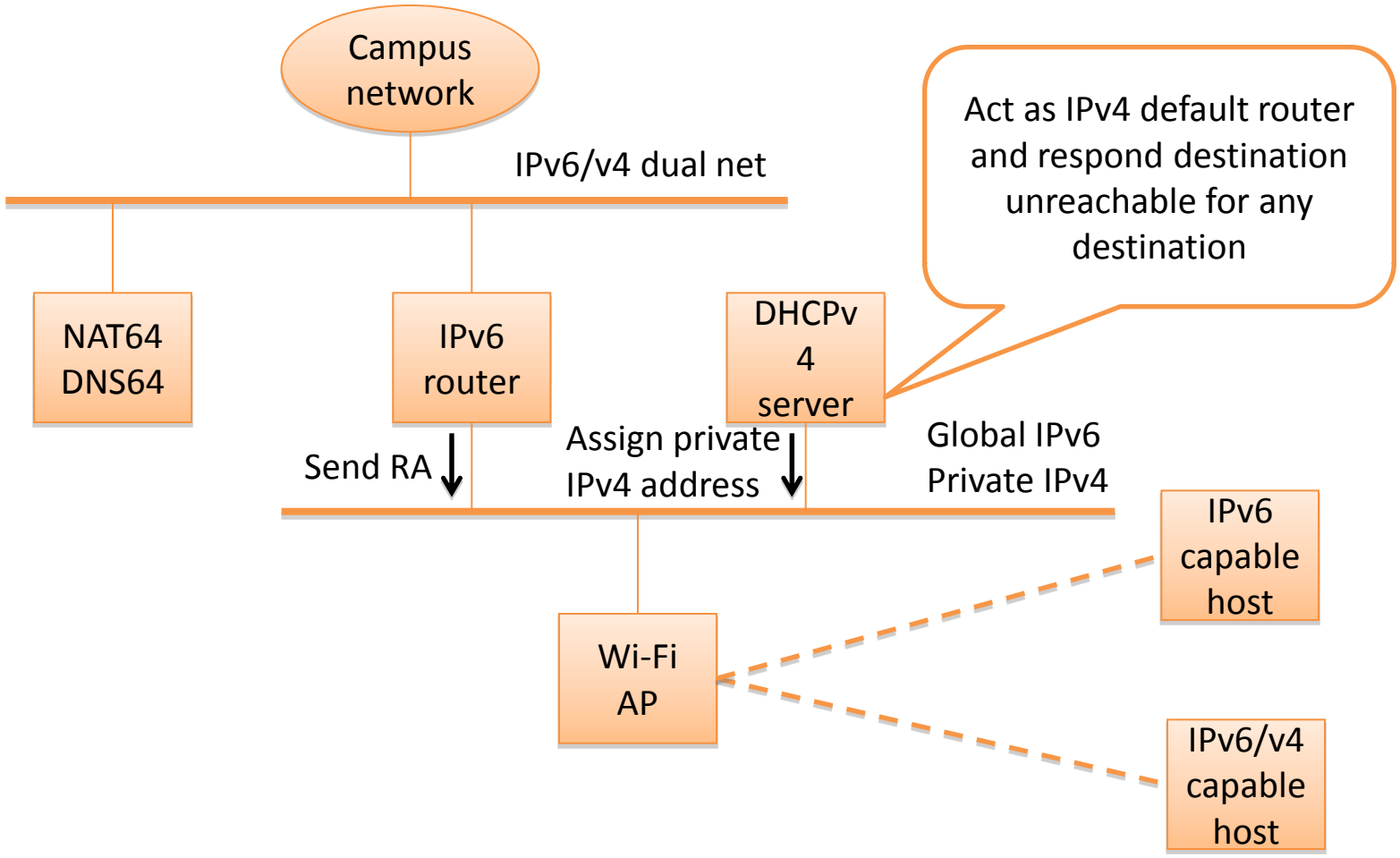
Configuration of phase 1



Results of phase 1

- IPv6 capable host
  - Works well for almost applications
  - NAT64/DNS64 provide connectivity between IPv6 client and IPv4 server
  - Temporary address issue
- IPv6/v4 capable host
  - Poor experience
  - RFC 3927 capable hosts activate IPv4 link-local address and it brings IPv4 black hole network to realize IPv4/IPv6 fallback issue

Configuration of phase 2



Results of phase 2

- IPv6 capable host
  - Works well for almost applications the same as phase 1
- IPv6/v4 capable host
  - Works well for almost applications against phase 1
  - Quickly fallback to IPv6 on ICMP destination unreachable response to any IPv4 destination

Temporary address issue

- RFC 4941 capable hosts generate temporary addresses whenever they connect to the network and receive RA
- The hosts generates many temporary addresses on unstable network connection (e.g. Wi-Fi roaming, note PC suspend/resume)
- Some security gateways reject IPv6 communication because they limit number of IP addresses for each node
- One solution is stateful address auto-configuration by DHCPv6

Key mechanisms

- NAT64 / DNS64
- IPv4 private address assignment by DHCPv4 and destination unreachable response for any IPv4 destination
- stateful address autoconfiguration by DHCPv6