464XLAT Optimization for CDNs/Caches

draft-palet-v6ops-464xlat-opt-cdn-caches

Jordi Palet (jordi.palet@theipv6company.com)
Alejandro D'Egidio (adegidio@telecentro.net.ar)

Problem Statement

- In IPv6-only networks using NAT46 (464XLAT, MAP-T), IPv4-only devices flows to dual-stack CDNs/Caches/services are terminated as IPv4, which means extra translations and the subsequent unnecessary overload

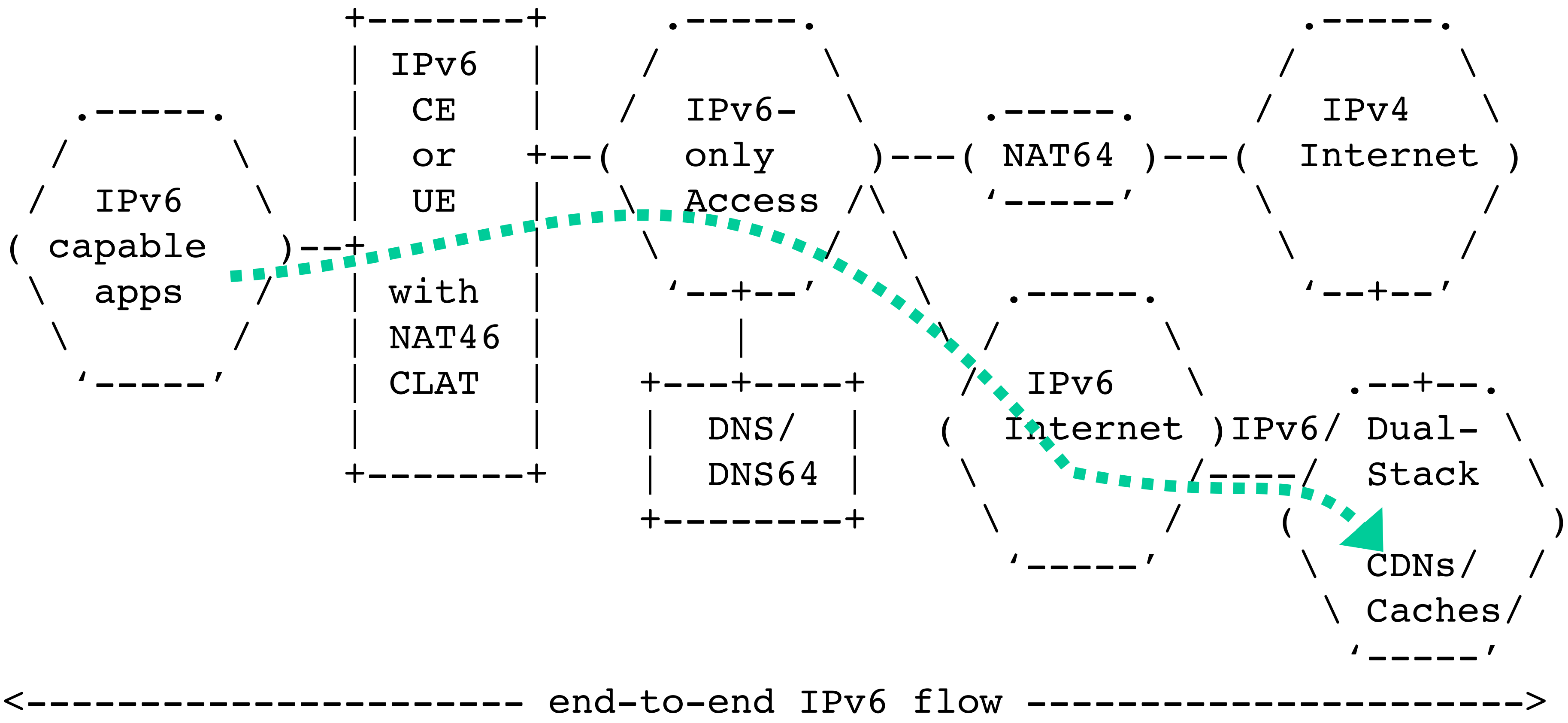
- In equivalent IPv4-only CGN use cases, the CDNs provide "private" addresses (typically 100.64.0.0/10) to avoid exactly the same issues

Typical 464XLAT Deployment

```
               +-------+     .-----.                     .-----.
               | IPv6  |    /       \                   /       \
   .-----.     |  CE   |   /  IPv6-  \     .-----.     /  IPv4   \
  /       \    |  or   +--(   only    )---( NAT64 )---(  Internet )
 /  LAN's  \   |  UE   |   \  Access /\    '-----'     \         /
(   Dual-   )--+       |    \       /  \                \       /
 \  Stack  /   | with  |     '--+--'    \   .-----.      '-----'
  \       /    | NAT46 |        |        \ /       \
   '-----'     | CLAT  |    +---+----+    /  IPv6   \
               |       |    |  DNS/  |   (  Internet )
               +-------+    |  DNS64 |    \         /
                            +--------+     \       /
                                            '-----'
```

Figure 1: Typical 464XLAT Deployment

IPv6-Capable device

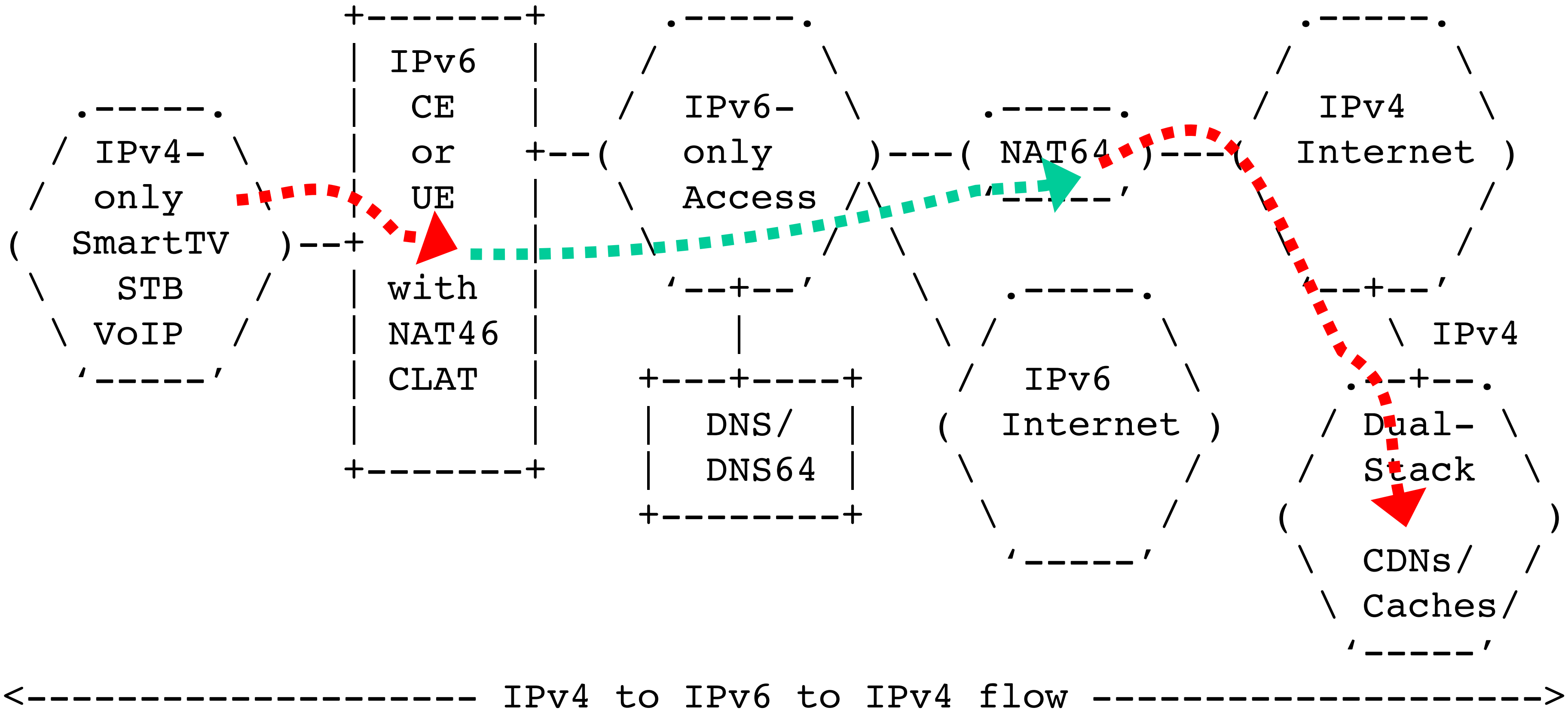


```
                +--------+     .------.                     .------.
                |  IPv6  |    /        \                   /        \
   .------.     |   CE   |   /  IPv6-   \     .------.    /   IPv4   \
  /        \    |   or   +--(   only     )---( NAT64 )---(  Internet  )
 /  IPv6    \   |   UE   |   \  Access  /\    ‘------’    \          /
(  capable   )--+        |    \        /  \                \        /
 \  apps    /   |  with  |     ‘--+--’     \   .------.     ‘--+--’
  \        /    | NAT46  |        |         \ /        \
   ‘------’     | CLAT   |    +---+-----+    /   IPv6   \     .--+--.
                |        |    |  DNS/   |   (  Internet )IPv6/ Dual- \
                +--------+    | DNS64   |    \         /----/  Stack  \
                              +---------+     \       /    (           )
                                               ‘------’     \  CDNs/  /
                                                             \ Caches/
                                                              ‘------’

<---------------------- end-to-end IPv6 flow ---------------------->
```

Figure 3: 464XLAT access to CDNs/Caches by IPv6-capable apps

IPv4-only device



```
                  +--------+     .------.                     .------.
                  |  IPv6  |    /        \                   /        \
    .------.      |   CE   |   /  IPv6-   \     .------.    /  IPv4    \
   / IPv4-  \     |   or   +--(   only     )---( NAT64  )---(  Internet  )
  /  only    \    |   UE   |   \  Access  /\    `------'    \          /
 (  SmartTV   )---+        |    \        /  \                \        /
  \   STB    /    |  with  |     `--+---'    \    .------.    `--+---'
   \ VoIP   /     | NAT46  |        |         \  /        \       \ IPv4
    `------'      | CLAT   |   +----+-----+    /  IPv6    \    .--+---.
                  |        |   |  DNS/    |   (  Internet  )  / Dual-  \
                  +--------+   |  DNS64   |    \          /  /  Stack   \
                               +----------+     \        /  (            )
                                                 `------'    \  CDNs/   /
                                                              \ Caches/
                                                               `------'
<-------------------- IPv4 to IPv6 to IPv4 flow -------------------->
```

Figure 4: 464XLAT access to CDNs/Caches by IPv4-only apps

IPv4-only device (optimized)

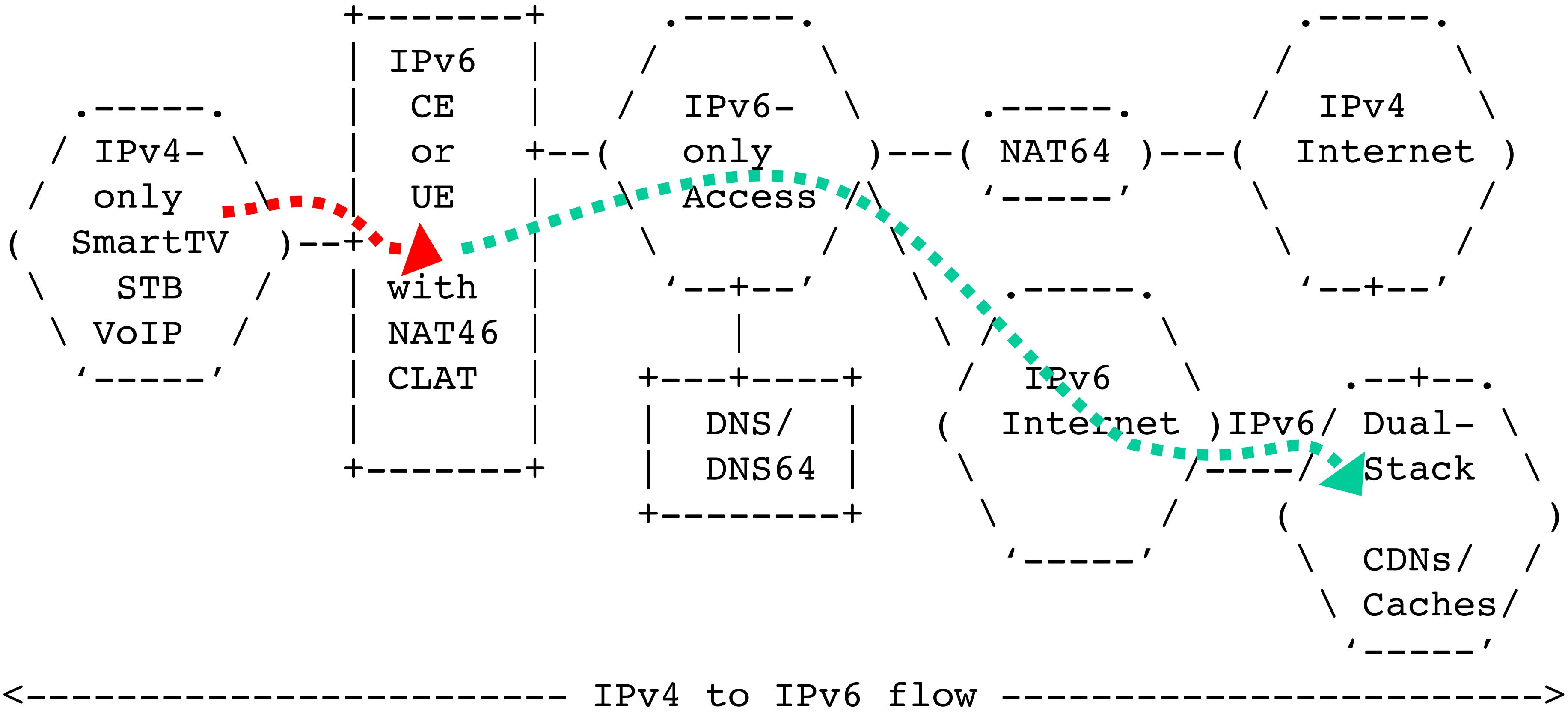
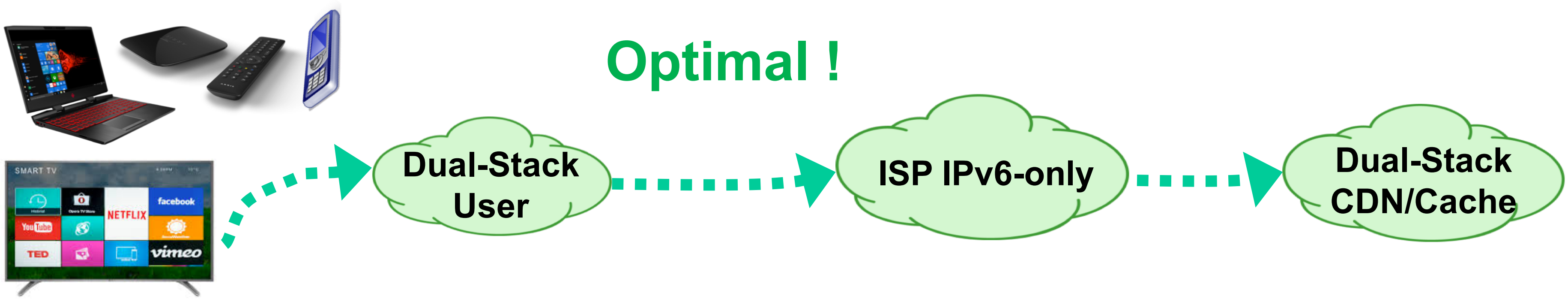


Figure 5: Optimized 464XLAT access to CDNs/Caches by IPv4-only apps

In Summary



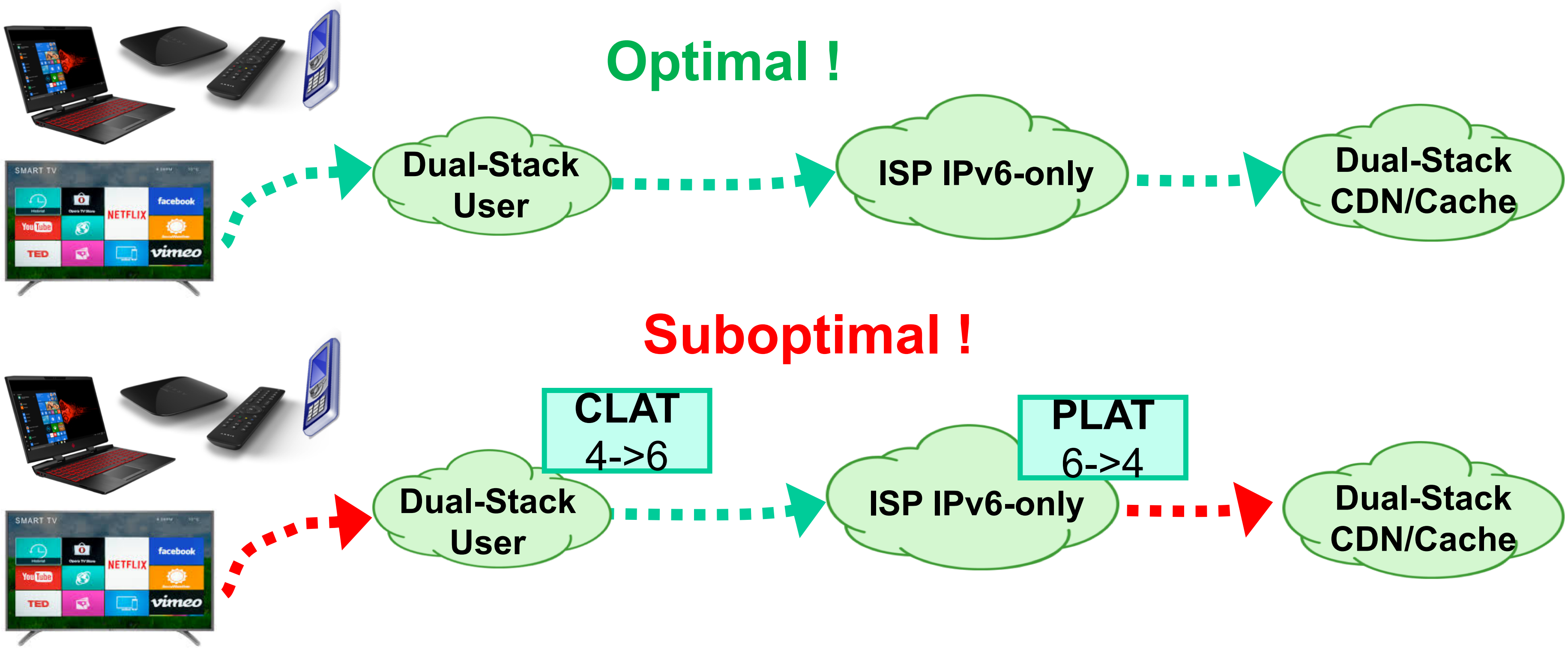
Optimal !

Dual-Stack User

ISP IPv6-only

Dual-Stack CDN/Cache

In Summary



Optimal !

Dual-Stack User

ISP IPv6-only

Dual-Stack CDN/Cache

Suboptimal !

CLAT 4->6

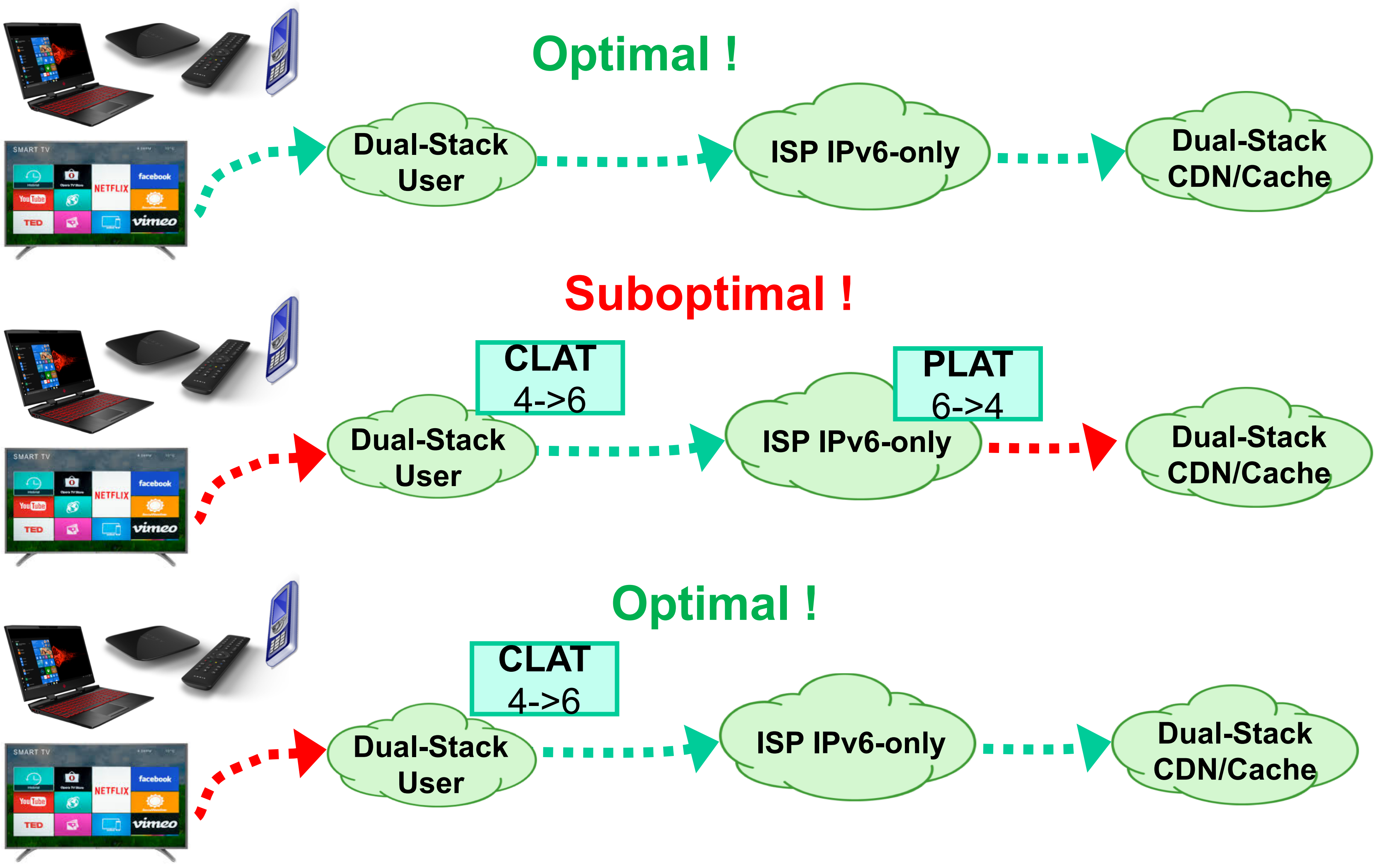
PLAT 6->4

Dual-Stack User

ISP IPv6-only

Dual-Stack CDN/Cache

In Summary



Optimal !

Dual-Stack User

ISP IPv6-only

Dual-Stack CDN/Cache

Suboptimal !

CLAT 4->6

PLAT 6->4

Dual-Stack User

ISP IPv6-only

Dual-Stack CDN/Cache

Optimal !

CLAT 4->6

Dual-Stack User

ISP IPv6-only

Dual-Stack CDN/Cache

Approach 1: DNS/Routing-based

- CLAT translate A records into AAAA:
  - – WKP::A or NSP::A
- CDN/Cache provider configures dedicated interfaces to match WKP::A or NSP::A

www.example.com	A	192.0.2.1
CLAT translated to		64:ff9b::192.0.2.1
CDN IPv6 interface must be		64:ff9b::192.0.2.1
Operator must have a specific route to		64:ff9b::192.0.2.1

- Issues:
  - – Only works if “local/private” connectivity
  - – CDN/Cache provider needs to do “something”

Approach 2: CLAT/DNS-proxy-EAMT

- NAT64/CLAT/CE is also a DNS proxy/stub resolver, so an internal interaction can be created.
- This approach uses existing IPv4 and IPv6 addresses (A, AAAA RRs), so no additional complexity for services.
- Steps:
  - Detection of IPv4-only devices or apps
  - Detection of IPv6-enabled service
  - Creation of EAMT entries
  - Forwarding path for existing EAMT entries
  - Maintenance of the EAMT entries

Approach 2 Example

- Example

www.example.com	A	192.0.2.1
	AAAA	2001:db8::a:b:c:d
EAMT entry	192.0.2.1	2001:db8::a:b:c:d
NAT64/CLAT translated to		2001:db8::a:b:c:d
CDN IPv6 interface already is		2001:db8::a:b:c:d
Operator already has specific route to		2001:db8::a:b:c:d

1. A query for www.another-example.com A RR is received
2. www.another-example.com A 192.0.2.1
3. www.another-example.com AAAA 2001:db8::e:e:f:f
4. A conflict has been detected
5. The existing EAMT entry for 192.0.2.1 is set as invalid

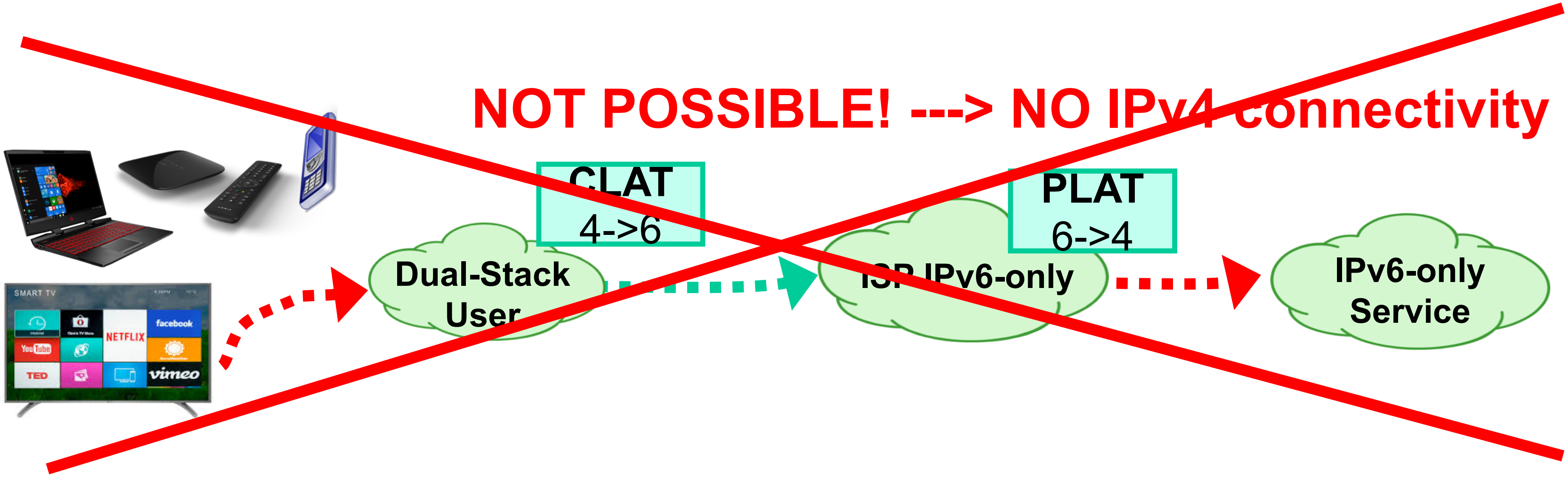
Approach 2: Additional Considerations

- Behavior in case of multiple A/AAAA RRs
- Behavior in case of presence/absence of DNS64
- Behavior when using literal addresses or non IPv6-APIs
- False detection of a dual-stack host as IPv4-only
- Behavior in presence of HE
- Behavior in case of Foreign DNS
    - Devices/apps using other DNS
    - DNS privacy/encryption
    - DNS modified by user in OS
    - DNS modified by user in CE
    - Combinations of above

Approach 3: CLAT-provider-EAMT

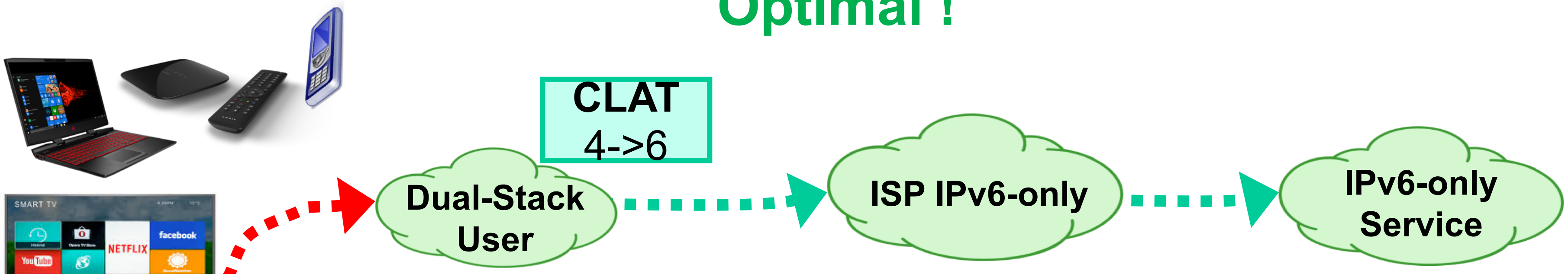
- Similar to previous one, but no ”automated” EAMT
- Operator must push or CE must pull the table
- It will work even if user change DNS for STB, SmartTV, …
- More control from the operator
  - EAMT pairs may be built “apart” from DNS
- Issues:
  - Increase complexity
    - Is the benefit worth for it?
  - Need to add TTL (from DNS) to EAMT

Solution for IPv6-only Services?



NOT POSSIBLE! ---> NO IPv4 connectivity

Optimal !



Add A RRs even if IPv4 is not available

Next steps

- Become a WG item ?

- Title change ?

- New inputs ? CDN/cache providers, others