

464XLAT/MAP-T Optimization

**draft-ietf-v6ops-464xlat-
optimization-06**

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 many cases this may become a show-stopper
- In equivalent IPv4-only CGN use cases, the CDNs accept “private” addresses (typically 100.64.0.0/10) to avoid exactly the same issues

Typical 464XLAT Deployment

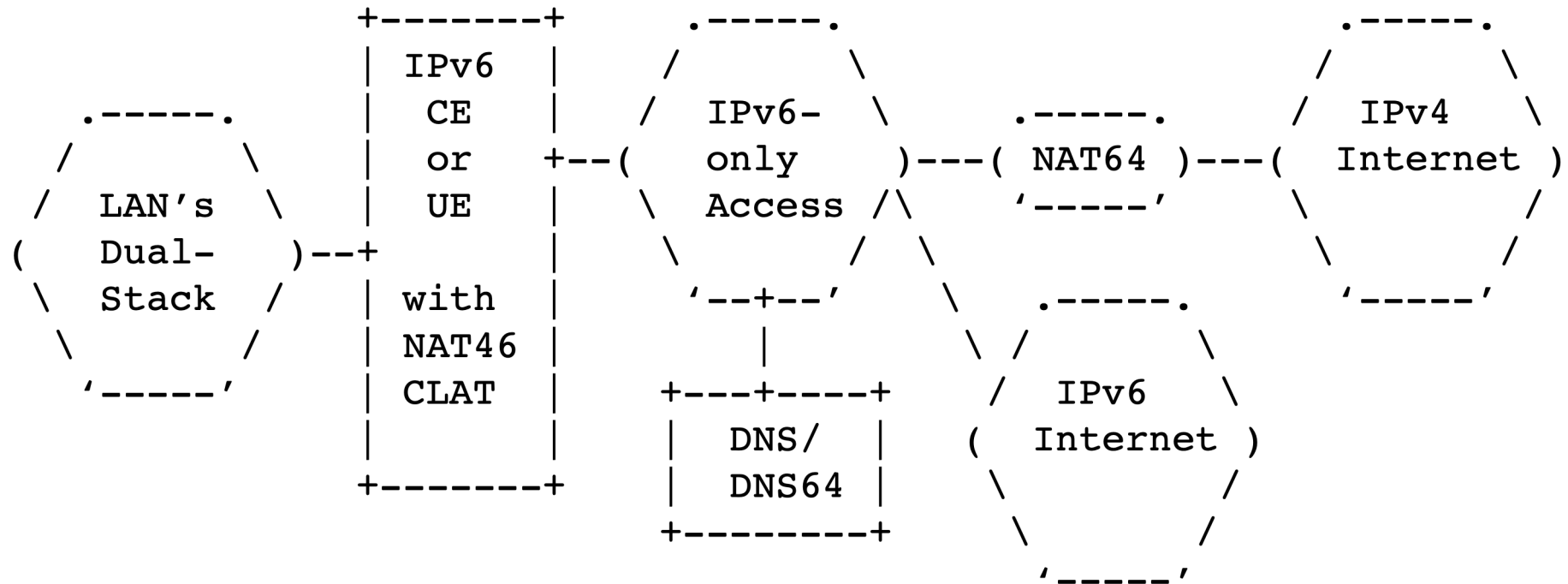


Figure 1: Typical 464XLAT Deployment

IPv6-Capable device

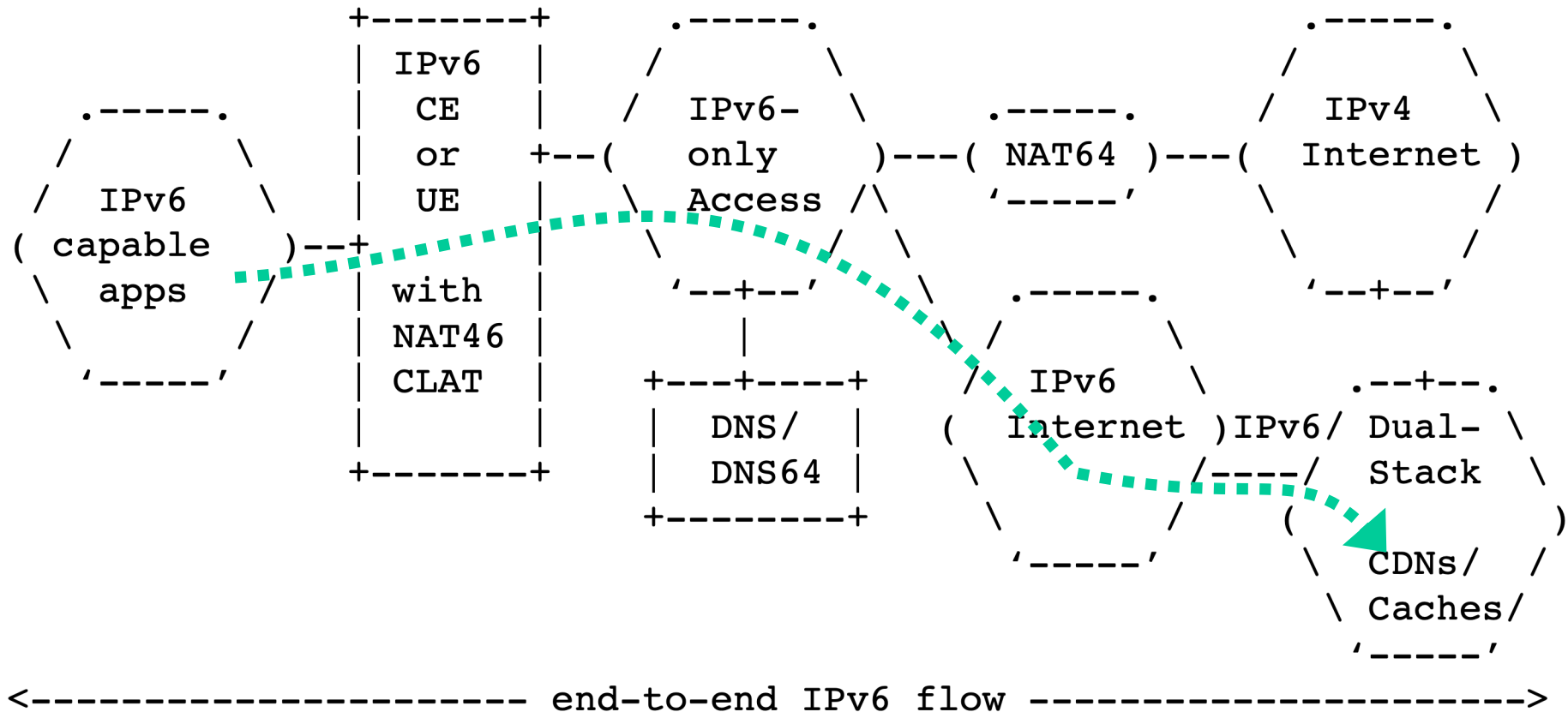


Figure 3: 464XLAT access to CDNs/Caches by IPv6-capable apps

IPv4-only device

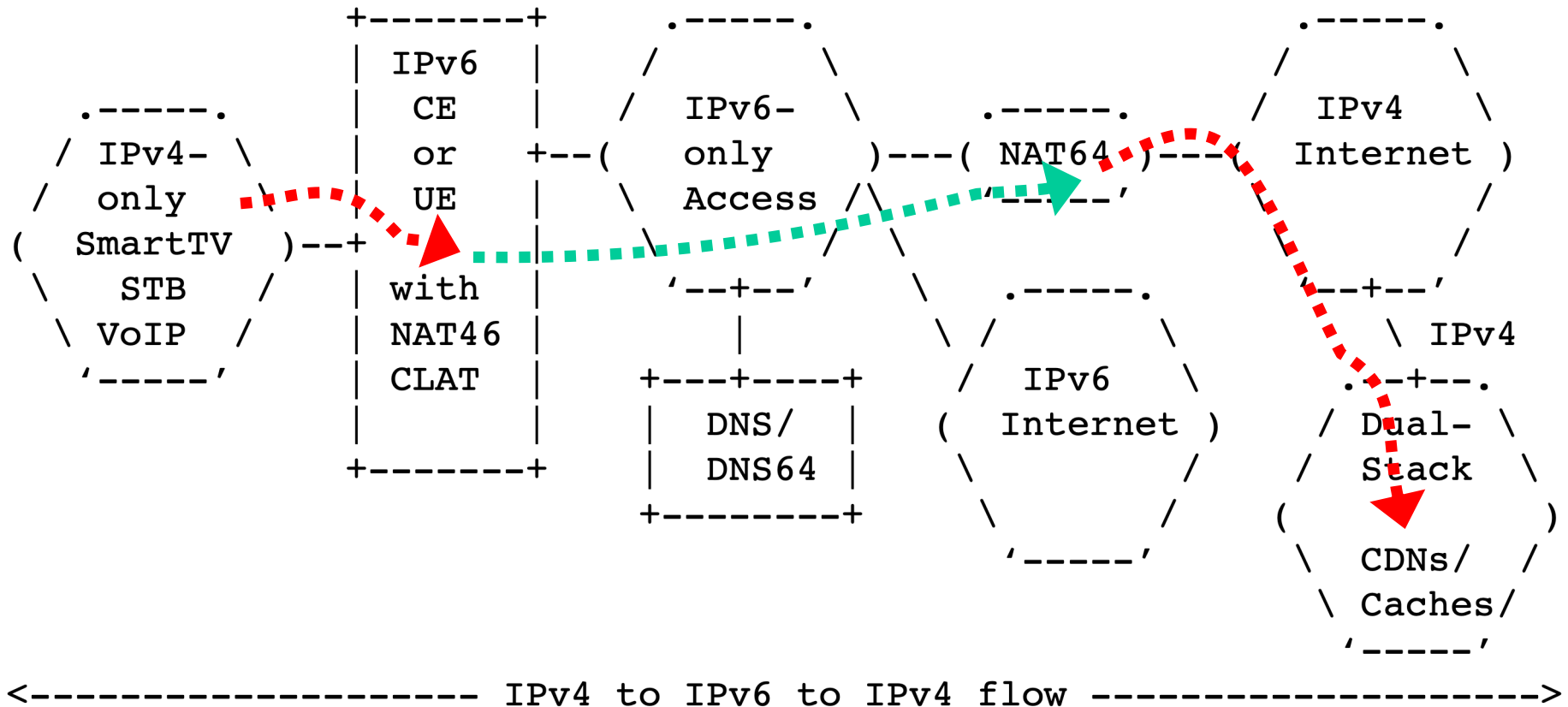


Figure 4: 464XLAT access to CDNs/Caches by IPv4-only apps

IPv4-only device (optimized)

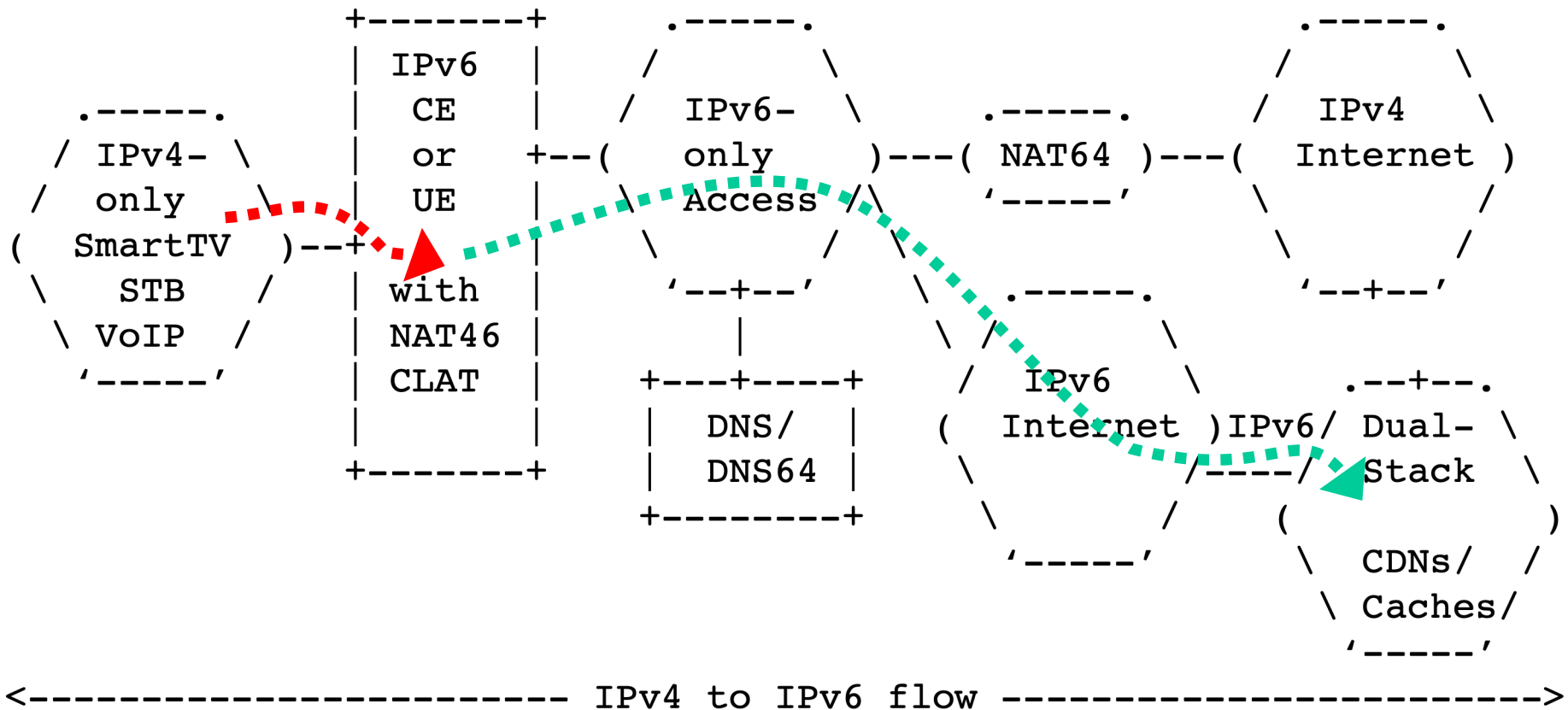


Figure 5: Optimized 464XLAT access to CDNs/Caches by IPv4-only apps

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
	AAAA	64:ff9b::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 more specific route to		64:ff9b::192.0.2.1

- draft-ietf-v6ops-nat64-wkp-1918 facilitate this, so CDNs can do the same they do for IPv4 CGN

IPv4-only device (optimized)

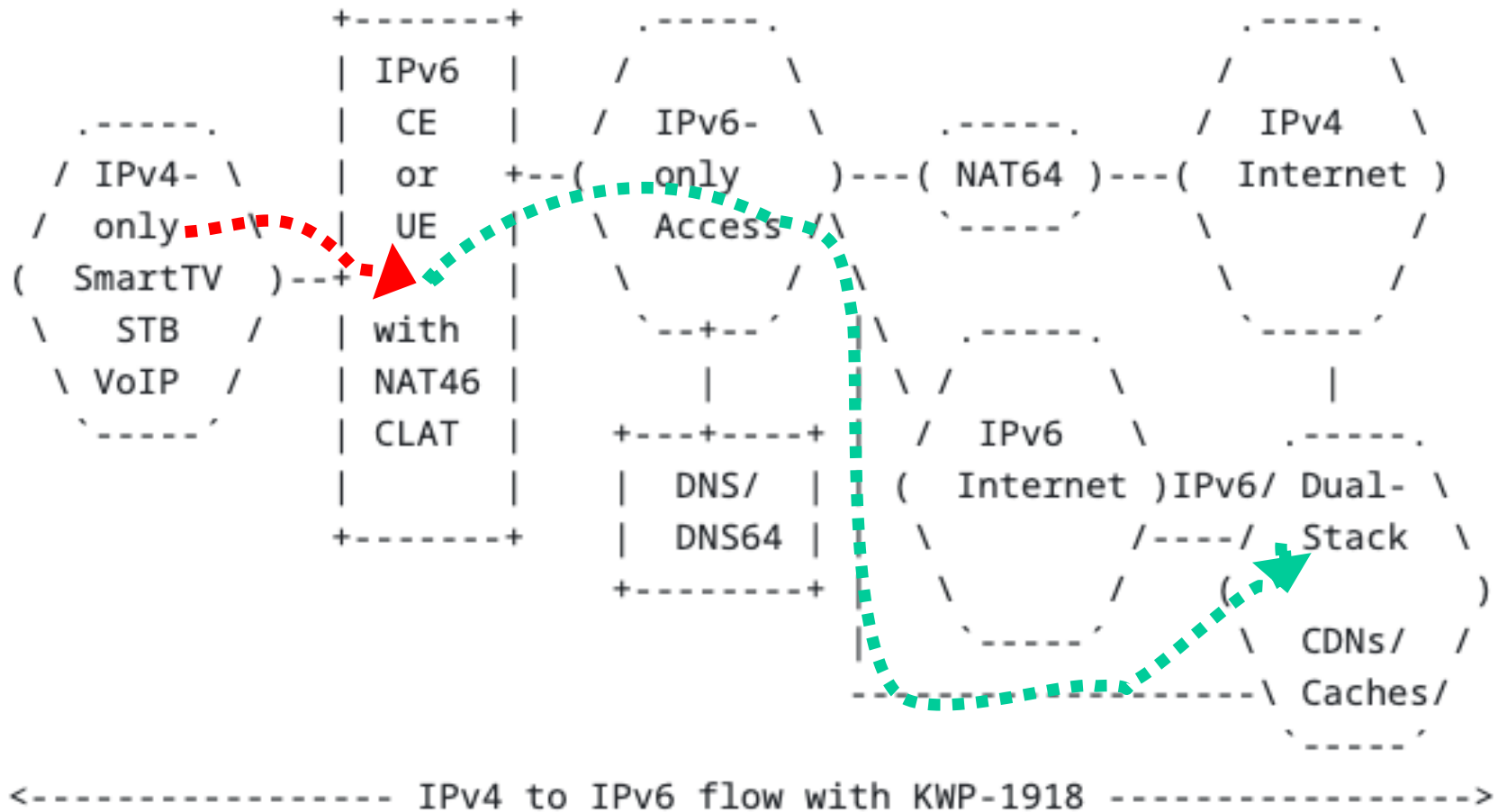


Figure 6: 464XLAT by IPv4-only apps with DNS/Routing-based Solution

Approach 2: CLAT/DNS-proxy-EAMT

- NAT46/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
 - Same MAC bound only to IPv4 address, not IPv6.
 - Detection of IPv6-enabled service
 - Creation/maintenance of extended EAMT (RFC7757)
 - Forwarding path for existing EAMT entries via stateful NAT46

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?
 - The CDN/cache provider needs to provide API to update the EAMT, or the operator use their own caches to build it

Conclusions

- Previously Approach 1 (DNS/Routing-based), was not recommended.
- However, if we accept the change proposed by draft-ietf-v6ops-nat64-wkp-1918, Approach 1 is a much simpler solution, and equivalent to what is being done by CDNs/caches.

Next Steps

- Can CDN/cache operator confirm if they agree with this view?
- Should we split the document in 2:
 - Actual one, informational with the 3 approaches
 - One short BCP with the recommended DNS/Routing-based approach