Core Use-Case Requirements for MSR6
from operational/developer perspective
aka: extracted/summarized from use-cases

IETF114 Philadelphia
v1.1 - 07/21/2022

how to fit 30 years of IP Multicast protocol design, deployment and multicast app-development experience into too little time and slides

Toerless Eckert (Futurewei USA), tte@cs.fau.de
Core MSR6 use-case requirements

There is really only one requirement:

“Simple”\(^1\), IPv6 integrated\(^2\), “End-to-End”\(^3\), stateless \(^4\), IPv6 multicast\(^5\) for IPv6-only networks \(^6\)

(6) What IPv6 only networks ?
   A: All IPv6 networks that require IPv6 multicast!

   Explained by other drafts/presentations: SP/WAN/Metro native IPv6 (with/without SRv6) (e.g.:IPTV, MVPN), DCN, OTT/Overlays

   IMHO also: Any enterprise, transportation, IoT network (small..large)
   Not considered by current MSR6 drafts – but should be
Refresh: BIER and IPv6-only networks

BIER RFC8279 (arch) + RFC8296 (header) is new layer ~ L2 / L2.5

- BIER router (BFR) forwarding is not IPv6 forwarding (RFC8200)
- BIER packets are not IPv6 packets (RFC8200)

Now: “One additional multicast forwarding plane for all unicast networks”

- Initially optimized for MPLS: header: label field (BIFT-ID), TC, OAM, signaling, ...
- Also BIER over L2 only, but little/no operator interest / stalled drafts

BIER-WG solution for IPv6 only networks / IPv6 Multicast?

- Do not build IPv6-only networks!

  draft-ietf-bier-bierin6 draft:
  Run separate BIER hop-by-hop forwarding plane - parallel + overlay

1. End-to-end tunnel for IPv6 Multicast over BIER: 2 headers
   BIER header + IPv6 (multicast) Header (so-called BIER flow overlay)

2. Transit over IPv6-unicast only routers (loose hops): 3 headers
   IPv6 unicast header (lower) + BIER header + IPv6 (multicast) header
Simplicity\(^{(1)}\), End-to-End\(^{(2)}\)
Operational / Architectural alignment/integration\(^{(2)}\) with IPv6 (unicast)

30 year experience: IP Multicast solutions are most successful when they minimize the additional ecosystem differences / work over the networks unicast solution

\textit{Do not introduce additional unnecessary multicast technology – reuse everything you can}

1989: IP/IPv6 Multicast is re-using / extending IP (RFC1112)\(^{(2)}\)

- Allowed to re-use or logically amend IP/IPv6 ecosystem components: SDKs/ sockets-API\(^(*)\), QoS DiffServ/IntServ(RSVP)\(^(*)\), ACLs\(^(*)\), any IP L2 encaps\(^(*)\), IPFIX\(^(*)\), IPsec\(^(*)\), ..

\(^(*)\) Most of this would all have to be reinvented / duplicated for BIER but not for MSR6.

Would have put bitstrings into IPv6 addresses if they where long enough (e.g. Cisco drafts in BIER)!

1990th: We tried novel multicast routing (MOSPF, DMVPN, …)

- Replaced by PIM + unicast routing (OSPF, ISIS, RIP,…) – because: do not re-invent routing for multicast!

200x: IPv4 multicast MVPN solution for MPLS/VPN SP networks – additional forward/control!

- IPv4 Multicast replaced by native MPLS multicast (mLDP/RSVP-TE/P2P)
- PIM/MVPN signaling replaced by BGP/MVPN signaling – because: we want a single protocol!

IMHO: BIER driven by MPLS SP use-cases (MPLS/MVPN). Well aligned/integrated there!

But not for the wide range of IPv6 networks – end-to-end – into IPv6 applications

Non-MPLS networks do not want or need an additional BIER ecosystem
Stateless$^4$, (1) .. and End-to-End$^3$

**BIER or native IPv6/MSR6: Scale and Convergence**

Finance / Telemetry / Content distribution / adaptive streaming would require hundreds of thousands of multicast states. Can not create, re-converge, operate!

**Native IPv6/MSR6: Operational simplicity (troubleshooting), safety, reliability**

All stateful multicast (IP or MPLS): Applications create (tree) state on routers in the network. No IETF standard multicast circuit beaker / state congestion/control solutions for multicast state. Any bad or attack multicast application can bring down stateful multicast routers

Unicast state: routing tables - do not grow with traffic, only with topology

Global MPLS SPs where deploying ingress-replication to avoid Multicast state on P nodes (RFC7988)

*after we invested 10 years in IETF to specify MPLS multicast – Core reason for BIER*

**Native IPv6/MSR6: Additional new multicast paradigm for applications**

IP Multicast (and SSM): application signaling: flow based – receiver join/leave group/channels

Only with stateless multicast:

Sender can **DIRECT EVERY PACKET SEPARATELY** – Destinations (and Path – with TE model)

**Only way to enable e.g.: adaptive streaming at scale via multicast**

BIER always wanted/wants to explore this. But IMHO NO WAY to get a ubiquitous BIER socket API

*Prior multicast socket extensions took almost 20 years (e.g.: SSM)*

*IPv6 extension header API already defined since 2003 (RFC3542) (no BIER API work)*
Result 1: Network centric core MSR6 arch reqs./goals
Expressed as Diffs over BIER

For native IPv6 networks

With or without SRv6: Reuse existing native IPv6, where ever possible
Common: Core network centric requirements from SP services design (e.g.: MVPN, IPTV).

Native IPv6 source routing hp-by-hop, end-to-end
Native IPv6 multicast, hop-by-hop, end-to-end

Apply RFC8200 rules hop-by-hop – applicable to IPv6 multicast and/or IPv6 source-routing
Replace only replication/state rules - not part of RFC8200, but PIM

Day 1 support for path steering and Strict + Loose hops

Best support for incremental adoption

More scalable BE and TE (path steering) modes

BIER “flat bitstring” scale limitation for large networks – BIER-TE but also BIER!

See draft-eckert-msr6-rbs for explanations

BIER-TE had to re-use BIER forwarding plane as much as possible
Forwarding planes now can do better!

Integrated support for service guarantees beyond best-effort

Latency, throughput, loss-protection
E.g.: support for DetNet natively with MSR6
not on top of MPLS/BIER + UDP end-to-end tunnels

Terminology

BIER: BFR, BFIR, BFER (BIER) -> MSR6: MSR, MSIR, MSER

Sample topology showing need for non-equal cost path steering and loose-hop support
**Result 2: End-to-End MSR6 architecture reqs./goals**

"Host based"

Terminology: MH: IPv6/MSR6 Host (or router) with MSR6 Application.

- No MSIR/MSER required
- But easily an add-on in existing network centric architecture deployments

For (use-case examples)

- IPTV Server -> Caches/Streamers in metro SP
- Data-Center IPv6 Multicast (stateless on DCN switches!)

Support for Host is Router

- Eg.: DC Server running all necessary routing (BGP, IGP)
  - E.g.: trusted Hypervisor
  - Much easier and important for native IPv6 than BIER (IMHO)

Host API for native source-driven multicast

- Eliminates IP Multicast (RFC1112), SSM (RFC3678) API where not ideal for apps
- E.g.: adaptive streaming cache to subscriber (per-packet functionality)

Support for Host is not router

- More novel network<->host signaling to explore (e.g.: PCEP ?)
- Use-cases: IoT (MANET/ROLL ?), Industrial, Enterprise
Summary / Conclusion

Stateless bitstring replication is the best new multicast direction in 40 years

BIER-WG - great stateless intra-SP solution for MPLS networks
But inferior fit for IPv6 networks and end-to-end applications

IETF/BIER can-not / should not change any more than necessary of the IPv6 ecosystem
Want to improve on what we learned from BIER (and its limitations) to get best IPv6 solution.

**MSR6: Keep it simple & Make IPv6 Multicast great again!**

*Native stateless multicast - for all IPv6 networks*
*Stateless IPv6 multicast into applications (DC, industrial, IoT, SP-edge, ...)*
*Re-use / share all of BIER that fits!*

Stuffed Agenda, No questions now ?!
But welcoming questions any time after the meeting!