Multicast Address Set Claim (MASC) Update

Deborah Estrin, Ramesh Govindan,
Mark Handley, Satish Kumar,
Pavlin Radoslavov (all USC/ISI)
and Dave Thaler (Microsoft)

Overview

1. MASC Overview
2. MASC Topology
3. Messages Types and Formats
4. MASC UPDATEs Processing and Propagation
5. Claims Comparison Function
6. Clash Resolving Mechanism
7. Misc. issues
8. Open issues
9. Conclusions

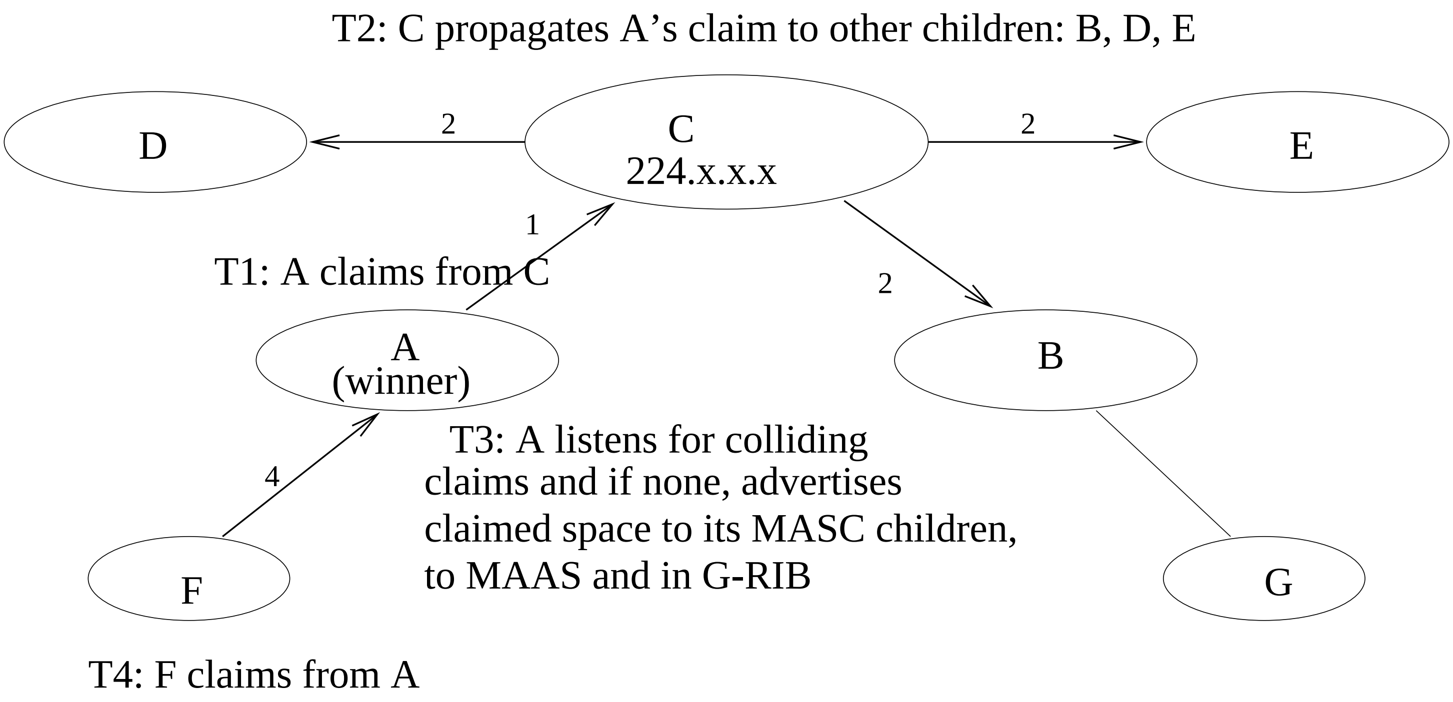
MASC Overview

- "Multicast-Address-Set Claim" (MASC):
  - Associates group ranges (prefixes) with AS's
  - Injects local associations into G-RIB (to be used by BGMP)
- Goals:
  - state aggregation (few prefixes per domain)
  - efficient use of address space
  - minimize flux in G-RIB
  - avoid third-party dependency (e.g. no registry)

Claim-Collide Mechanism

1. Choose a prefix
2. Claim and wait for collision
3. If no one opposes (i.e. no collisions), the claimed prefix could be used
4. If collision (lost), back-off and claim a new prefix

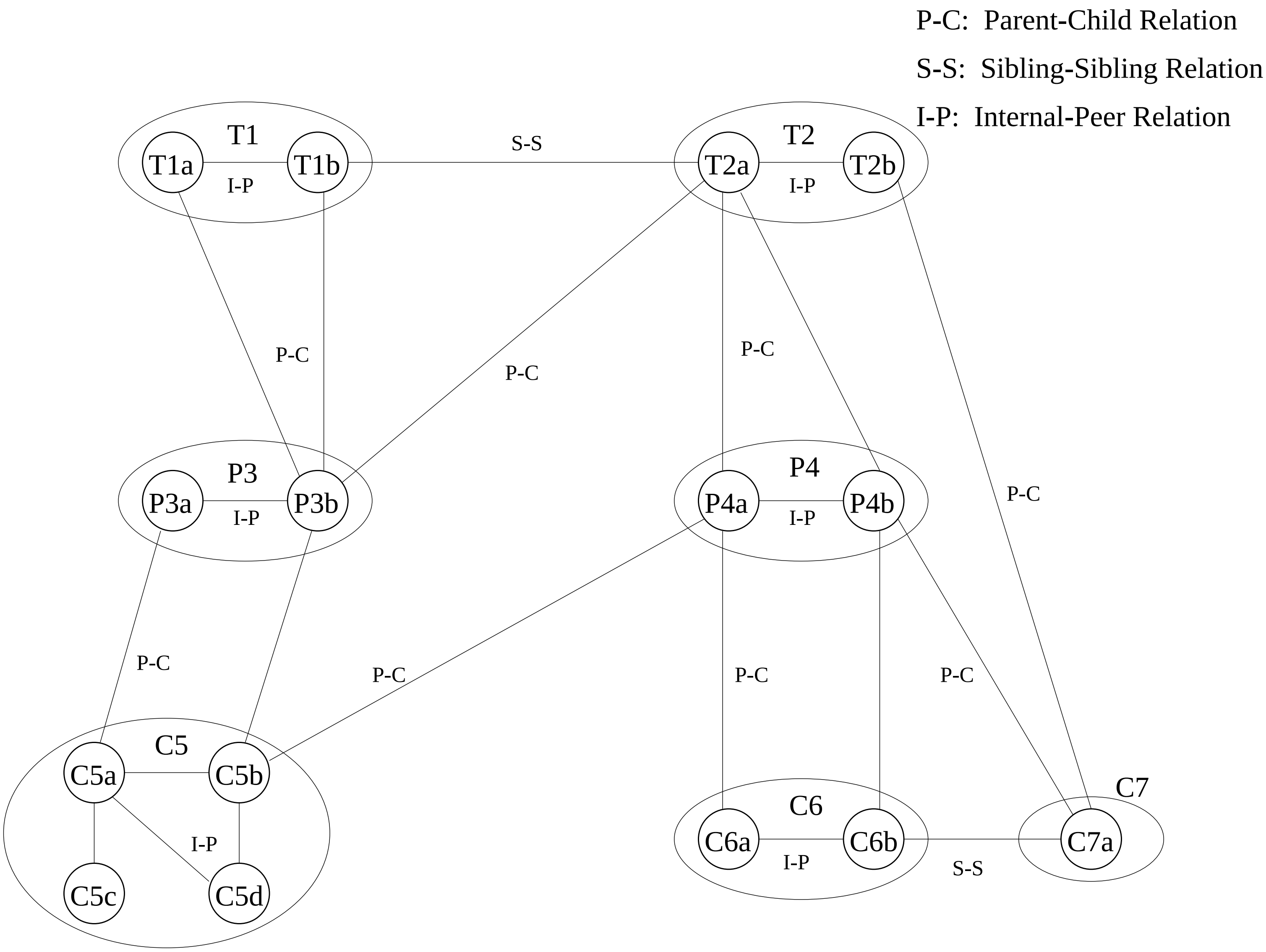
MASC Overview (cont.)



MASC Topology

- MASC will be usually run by BG(M)P routers
- A hierarchy of MASC domains; will follow the inter-domain topology
- A number of Top-Level Domains (TLDs), i.e. no single root
- Within a MASC domain, a number of MASC nodes connected by a TCP mesh (i.e. no need for any-to-any connectivity)
- A number of TCP connections between a parent and a child domain
- Could have TCP connections between siblings

MASC Topology Example



MASC Allocated Prefix

Each allocated (i.e. associated with a MASC domain) prefix has lifetime and can be used for:

- injection into G-RIB
- advertisement to the MASC children domain
- allocation of subranges by the local Multicast Address Allocation Servers

MASC UPDATEs Type

- NEW-CLAIM: “Claim new space”
- CLAIM-TO-EXPAND: “I already have allocated a subset of this claimed range”
- PREFIX-IN-USE: “I have allocated that space”
- CLAIM-DENIED: “The claim is denied”
- PREFIX-MANAGED: “Children, you can claim from that range”
- WITHDRAW: “The specified range is **temporary** obsoleted”

MASC UPDATEs Format

<table>
<tr><th>0</th><th>1 2</th><th>10</th><th>11 15</th><th>16 31</th></tr>
<tr><td>A</td><td>Rol</td><td>Reserved</td><td>AddrFam</td><td>Origin Domain Identifier</td></tr>
<tr><td colspan="5">Origin Node Identifier</td></tr>
<tr><td colspan="5">Claim Timestamp</td></tr>
<tr><td colspan="5">Claim Lifetime</td></tr>
<tr><td colspan="5">Claim Holdtime</td></tr>
<tr><td colspan="5">Address (variable length)</td></tr>
<tr><td colspan="5">Mask (variable length)</td></tr>
</table>

Rol = {INTERNAL, CHILD, SIBLING, PARENT}, updated hop-by-hop

MASC UPDATEs Processing

- Accept/reject
  - Timestamp + Holdtime > CurrTime?
  - Do I have same message in local cache?
- Compare with local domain's initiated messages/states and take appropriate actions: cancel a claim, withdraw an entry from G-RIB, etc.
- Save in the local cache (up to Timestamp + Holdtime)
- Propagate, based on UPDATE Type and relation to UPDATE Origin

UPDATEs Propagation and Failure Recovery

MASC-level (i.e. application level) “selected broadcast” to the MASC parent domain and/or all children and/or siblings:

- Simple
- “One-time” propagation
- Robust:
  - after connection (re)establishment, both sides exchange their UPDATEs
  - After a reboot, a child can obtain its own “claim-in-progress” state from its parent and/or siblings

Claims Comparison Function

1. PREFIX-IN-USE > CLAIM-DENIED > CLAIM-TO-EXPAND > NEW-CLAIM
2. Earliest claim (smallest timestamp) wins
3. Smallest Origin Node ID wins

Expanding a Prefix

Reclaim in advance the desired larger prefix

Renewing a Prefix

Reclaim in advance with the desired longer lifetime

Releasing a Prefix

- Implicit releasing (default): prefix left to timeout everywhere
- Explicit releasing (optimization): could send an explicit withdraw message; used for end-to-end interpretation

MASC Domain "Internal" Synchronization

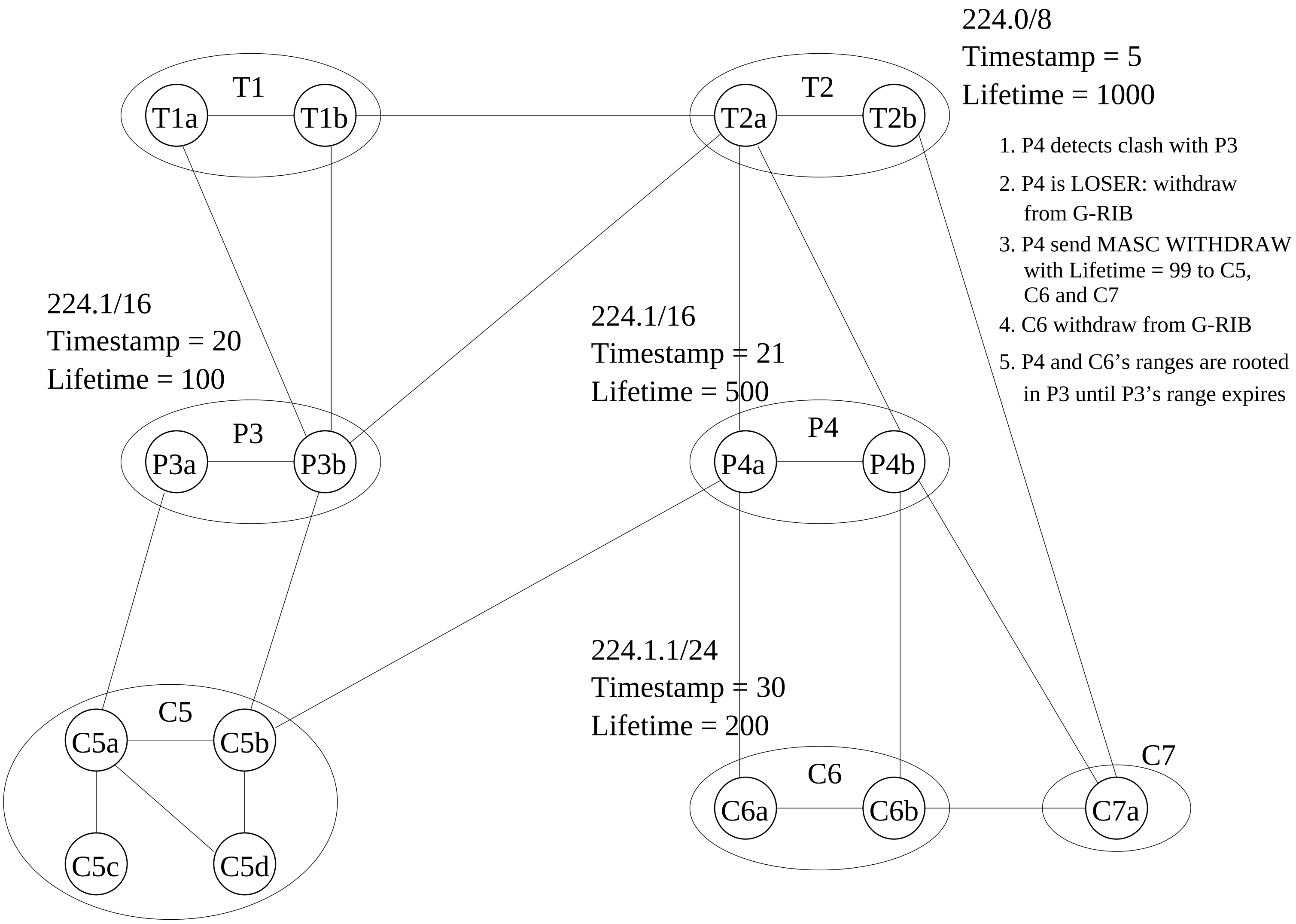
- Wait randomly between 0 and 10 minutes before initiating a new claim
- Use the same claim-collide mechanism
- Do not claim a new range, if collision winner is a node from the same domain

Clash Problems

Collision of PREFIX-IN-USE is a "clash". Could happen if network partitioning lasts longer than 48 hours, and if happens:

- Different sessions/applications may have obtained the same multicast address
- Different domains are injecting ambiguous entries into G-RIB, and will result in partitioning (lack of connectivity) in the inter-domain multicast routing

Clash Example



Clash Resolving Mechanism

- MUST solve the routing problem:
  - Only one domain (the "winner" of the clash) must remain Root Domain
  - The loser(s) of the clash have to withdraw from G-RIB
  - The loser(s) must inform their children to withdraw from G-RIB
- OPTIONAL: (if possible) use API to inform asyncronously applications to change the address

If need a better clash protection, increase the Claim Holdtime.

Changing network provider

- Keep temporarily a TCP connection with the old provider to “defend” my allocations
- If the new provider is willing to carry the traffic for the “migrated” prefix, inject into G-RIB

Administratively-Scoped Address Allocation

- Learn zones by listening to MZAP messages
- A domain without a parent within the scope zone will become a Zone TLD
- MASC domains within the same zone assume the same hierarchy relation as for the globally allocated range

MASC Debugging

Local MASC servers have the **aggregated prefix-to-domain** and **domain-to-prefix** lookup information. If more detailed information needed:

- Prefix-to-domain lookup: use mtrace
- Domain-to-prefix(es) lookup: open issue
  - Query a MASC/BGMP mode in that domain
  - How to find the address of a BGMP router in a particular domain?

MASC Storage

- Keep only local domain allocations in local storage (if such), otherwise use AAP to save into/query from the MAASs
- Do NOT need to save claims in progress

Default/Suggested Values

- CLAIM-WAITING-PERIOD: 48 hours
- INITIATE-CLAIM-DELAY: (0-10 minutes)
- Default initial claim size: 256 addresses
- Default initial claim lifetime: 30 days

Open Issues

- MASC TCP port
- Domain-to-prefix(ex) lookup mechanism
- Need to use relative time instead of absolute time?
- Change the 16 bits MASC Domain ID to 32 bits?
  - 16 bits AS Number
  - 16 bits locally assigned (to divide a large AS into several MASC domains)

Open Issues (cont)

- Allow indirect parent-child relationship (through child's sibling)?
- Get from IANA the Globally Allocatable Multicast Address Range:
  - (225.0.0.0 - 231.255.255.255) AND (233.0.0.0 - 238.255.255.255)?

Conclusions

- Simple and robust claim-collide mechanism
- MASC topology congruent to the unicast topology
- Low bandwidth and low memory usage
- Robust information recovery after reboot
- Simple messages propagation and processing
- Automatic clash resolving mechanism
- MASC protocol/mechanism is separated from the address allocation algorithm