Internet Engineering Task Force
INTERNET-DRAFT
draft-draft-handley-malloc-arch-00.ps

MALLOC WG
M. Handley, D. Thaler, D. Estrin
ISI, U. of Michigan, ISI

December 15, 1997
Expires: June 1998

# The Internet Multicast Address Allocation Architecture

**Status of this Memo**

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress".

To learn the current status of any Internet-Draft, please check the "1id-abstracts.txt" listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Distribution of this document is unlimited.

**Abstract**

This document proposes a multicast address allocation architecture for the Internet. The architecture is three layered, comprising a client-¿server protocol, an intra-domain protocol and an inter-domain protocol.

## 1 Introduction

This document proposes a multicast address allocation architecture for the Internet. This architecture is designed to scale to allocating a very large proportion of the 270 million IP4 multicast addresses available. It will also perform well in an IP6 environment where addresses are not a scare resource, but it is not currently clear whether different mechanisms would be more appropriate if good address space packing were not a primary requirement.

This architecture assumes that the primary scoping mechanism in use is administrative scoping. While solutions that work for TTL scoping are possible[1], they introduce significant additional complication for address allocation. Moreover, TTL scoping is a poor solution for multicast scope control, and our assumption is that TTL scoping of sessions will cease to be used before this architecture is widely used.

## 2 Requirements

From a design point of view, the important properties of multicast allocation mechanisms are robustness, timeliness, low probability of clashing allocations, and good address space utilisation. Where this interacts with multicast routing, we would like multicast addresses to allocated in a manner that aims agregation of routing state.

The robustness requirement is that an application requiring the allocation of an address should always be able to obtain one, even in the presence of other network failures.

### Timeliness

From a timeliness point of view, a short delay of a few seconds is probably acceptable before the client can be given an address with reasonable confidence in its uniqueness. If the session is defined in advance, the address should be allocated as soon as possible, and should not wait until just before the session starts. It is acceptable to change the multicast addresses used by the session up until the time when the session actually starts, but this should only be done when it averts a significant problem such as an address clash that was discovered after initial session definition.

### Availability, Correctness, and Address Space Packing

A multicast address allocation scheme should always be available, and always able to allocate an address that can be guaranteed not to clash with that of another session. However, to *guarantee* no clashes would require a top-down partitioning of the address space, and to do this in a manner that provides sufficient spare space in a partition to give a reasonable degree of assurance that an addresses can still be allocated for a significant time in the event of a network partitioning would result in significant fragmentation of the address space. In addition, providing backup allocation servers in such a hierarchy, so that fail-over (including partitioning of a server and its backup from each other) does not cause collisions would add further to the address space fragmentation.

Given that we cannot achieve constant availability, guarantee no clashes, and achieve good address space usage, we must prioritise these properties. We believe that achieving good address space packing and constant availability are more important than guaranteeing that address clashes never occur. What we aim for is a high probability that an address clash does not occur, but we accept that there is a finite probability of this happening. Should a clash occur, either the clash can be detected and addresses changed, or hosts receiving additional traffic can prune that traffic using source-specific prunes available in IGMP version 3, and so we do not believe that this is a disastrous situation.

In summary, tolerating the possibility of clashes is likely to allow allocation of a very high proportion of the address space in the presence of network conditions such as those observed in [2]. We believe that we can get good packing and good availability with very good collision avoidance, while we would have to compromise packing and availability significantly to avoid all collisions.

## Address Dynamics

Multicast addresses may be statically allocated or dynamically allocated. Statically allocated addresses are allocated by IANA for specific protocols that *require* well known addresses to work. Examples of static addresses are 224.0.1.1 which is used for the Network Time Protocol and 224.2.127.255 which is used for global scope multicast session announcements. Protocols should not normally be given static multicast addresses unless they provide basic infrastructure that must self-organise and cannot therefore use dynamic addresses. Local protocols that use multicast for bootstrap purposes should not normally be given their own static multicast address, but should bootstrap themselves using a well known service location address which can be used to announce the binding between local services and multicast addresses.

For most purposes, the correct way to use multicast is to obtain a dynamic multicast address. These addresses are provided on demand and have a specific lifetime. An application should request an address only for as long as it expects to need the address. Under some circumstances, an address will be granted for a period of time that is less than the time that was requested. This will occur rarely if the request is for a reasonable amount of time. Applications should be prepared to cope with this when it occurs. At any time during the lifetime of an existing address, applications may also request an extension of the lifetime, and such extensions will be granted when possible. When the address extension is not granted, the application is expected to request a new address to take over from the old address when it expires, and to be able to cope with this situation gracefully.

These restrictions on address lifetime are necessary to permit the address allocation architecture to self-organise around current address usage patterns in a manner that ensures addresses are agregatable and multicast routing is reasonably close to optimal. In contrast, statically allocated addresses may be given sub-optimal routing.

# 3 Overview of the Architecture

There are three parts to this architecture:

- A protocol (MDHCP) that a multicast client uses to request a multicast address from a *local* multicast address allocation server (MAAS).
- A protocol (AAP) that MAAS servers use to claim multicast addresses and inform their peer MAAS servers which addresses are in use.
- A protocol (MASC) that allocates multicast address sets to domains. Individual addresses are allocated out of these sets by MAAS servers.

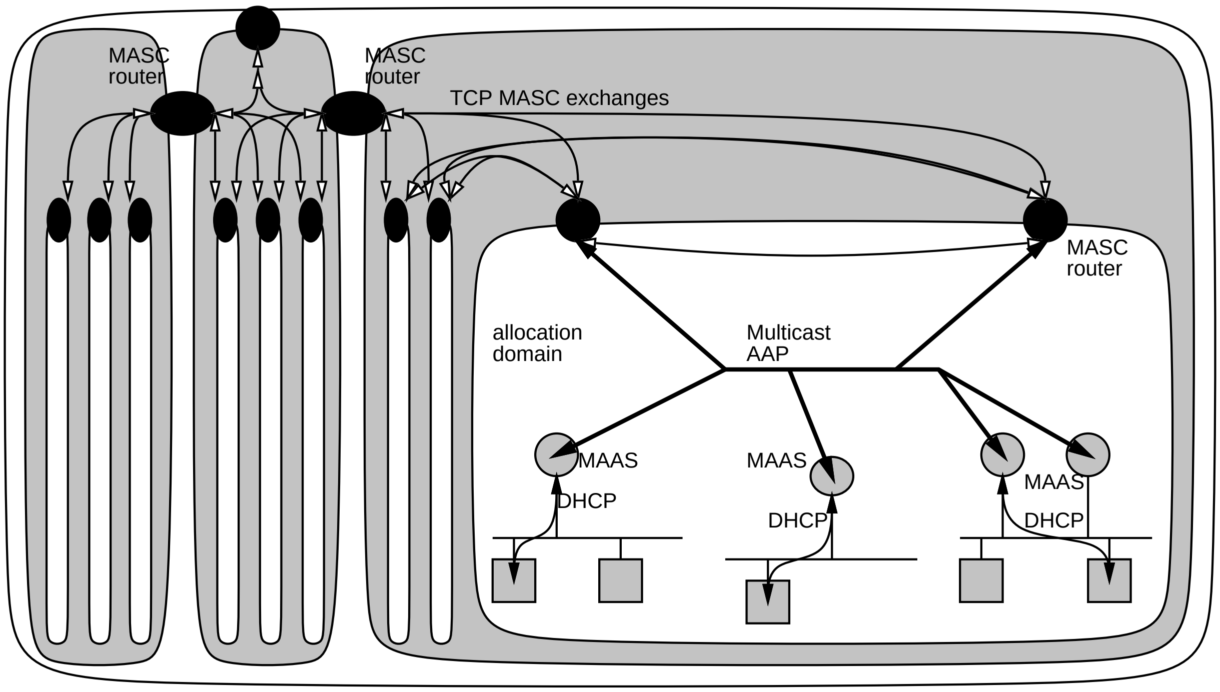


Figure 1: An Overview of the Multicast Address Allocation Architecture

We have three protocols because they serve slightly different purposes and require different design trade-offs. An overview of how these protocols fit together is shown in figure 1.

### Multicast Dynamic Host Configuration Protocol (MDHCP)

MDHCP is used by a client to request an address from a MAAS server. When the server grants an address, it has become the server's responsibility to ensure that this address is not then reused elsewhere within the same scope.

### Address Allocation Protocol (AAP)

AAP is used by a MAAS server to claim multicast addresses that it has allocated, and if necessary to defend these addresses if another MAAS server attempts to allocate the same address. A MAAS server keeps track of all the other multicast addresses in use within the same allocation domain, and when it allocates an address it ensures that the address is not already in use. AAP is also used by nodes performing MASC to inform the MAAS servers of the address set (consisting of a list of address/mask/lifetime tuples) that is available. Under normal conditions a MAAS server should only allocate an address from the unused addresses in this advertised set.

AAP uses multicast, and operates on a timescale of milliseconds to seconds.

### Multicast Address Set Claim (MASC) Protocol

MASC is used by nodes (typically these nodes are routers) to claim address sets that satisfy the needs the MAAS servers within their allocation domain. Thus when a MASC node discovers that there are close to insufficient multicast addresses available for AAP to perform well, the MASC node claims a larger address set. MASC is hierarchical, so MASC nodes below the top level see address set advertisements by higher level MASC nodes, and must choose new address sets from those being advertised. Address sets are also claimed with a lifetime, and that lifetime cannot be longer than the lifetime of the parent address set. When the lifetime of an address set expires, that set will normally be given up. At this point AAP should no longer be advertising addresses from the set. However, if there is still sufficient demand, and the parent set is renewed, then the address set may be renewed. Typically each allocation domain will be advertising several address sets with different lifetimes at any time, allowing the MAAS servers to choose appropriate addresses for their clients.

MASC uses unicast TCP. MASC cannot use multicast as inter-domain multicast routing using BGMP relies on the address sets allocated by MASC to build trees of domains. Typically MASC is performed in routers that are running BGP, and the TCP connections parallel those used by BGP.

## 4 Overview of the Allocation Process

Assuming that allocation has been performed for some time (the startup conditions for MASC are slightly more complex), then one or more MASC nodes bordering an allocation domain will be advertising address sets into the domain using multicast AAP.

MAAS servers within the domain receive these address sets and cache them as the currently allowable addresses for that domain. These address sets are unconditionally valid for their advertised lifetime and cannot be revoked before their lifetime has expired.

A MAAS server also receives individual domain-wide multicast address claims via AAP from other MAAS servers within the domain. It also caches these addresses as being in use for their reported lifetime.

When a client needs a multicast address, it locally multicasts a request for scope information using MDHCP. Any local MAAS server can respond, although usually such servers will be configured to have

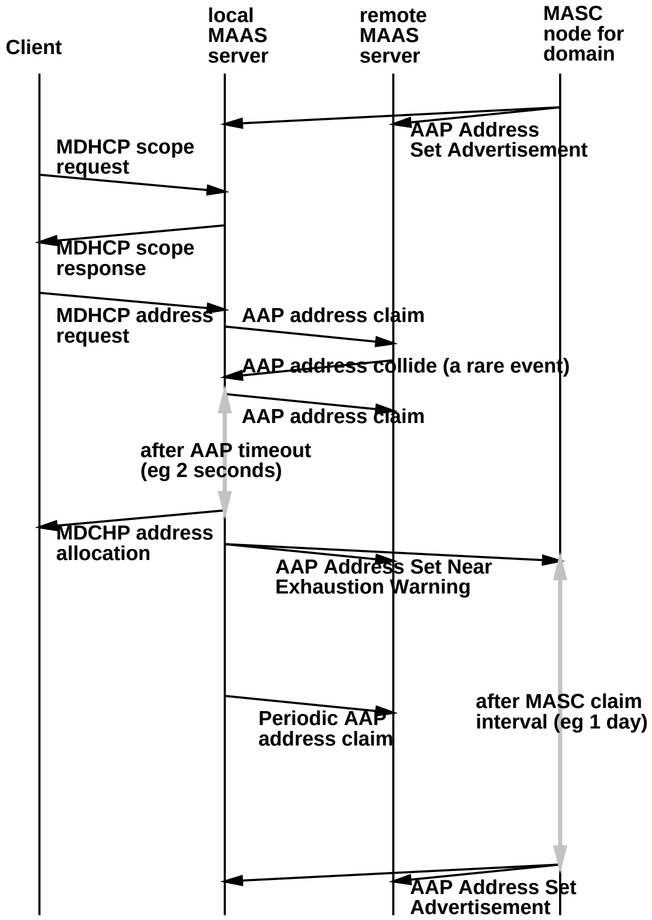


Figure 2: Some Message Exhanges in the Address Allocation Process

primaries and backups. The MAAS server that responds provides a list of valid scopes to the client. The client then chooses a scope, and requests an address from the MAAS server for a certain time interval. The MAAS server then chooses an address from those not currently used in the MASC address set that satisfies the requested time interval (if possible), and advertises this domain-wide using AAP. If no clashing AAP claim is received within a short time interval, then the address is returned to the client by MDHCP. If a clashing claim is received by the MAAS server, then it chooses a different address and tries again. If no address set is long enough to match the requested time interval, then the MAAS server truncates the time interval to that of the longest address set available before advertising the address using AAP.

Some of the exhanges in this process are illustrated in figure 2.

## 4.1 Allocation Domains

In this document and the related document we use the term allocation domain. An allocation domain is an administratively scoped multicast-capable region of the network. Typically it will be bordered by routers that perform BGMP-interdomain multicast routing. We expect that allocation domains will normally coincide with unicast Autonomous Systems (AS's). This is based on the assumption that BGMP and BGP are closely tied together and we want the “best” root domain for the BGMP tree.

If an AS is too large, or the network administrator wishes to run different intra-domain multicast routing in different parts of an AS, that AS can be split by manual setup of a BGMP boundary that is not a BGP unicast boundary. This is done by setting up a multicast AS boundary dividing the unicast AS into two or more multicast AS's, with border routers having the M-RIB (including the GRIB).

If an AS is too small, we'll get address space fragmentation if the AS is its own MASC domain. Here, there's no real reason why the border router to the site need run BGMP, even though it's running BGP. The domain can use AAP directly to talk to the MASC routers of its provider, and not cause any additional fragmentation. An AS should probably take this course of action if:

- it's connected to a single provider.
- it has no children (it's not transit to another AS)
- it has fewer than N multicast addresses of larger than AS scope allocated on average.

The strawman value for N is 256.

## References

[1] Mark Handley, “Multicast Session Directories and Address Allocation”, Chapter 6 of PhD Thesis entitled “On Scalable Multimedia Conferencing Systems”, University of London, 1997. http://north.east.isi.edu/ mjh/thesis.ps.gz

[2] Mark Handley, “An Analysis of Mbone Performance”, Chapter 4 of PhD Thesis entitled “On Scalable Multimedia Conferencing Systems”, University of London, 1997. http://north.east.isi.edu/ mjh/thesis.ps.gz