



MOTOROLA

Scoped Address Discovery Protocol (SADP)

Roger G. Kermode

Motorola

Chicago Corporate Research Laboratories

1301 East Algonquin Road, MS 2712,

Schaumburg, IL, 60196-1078, USA

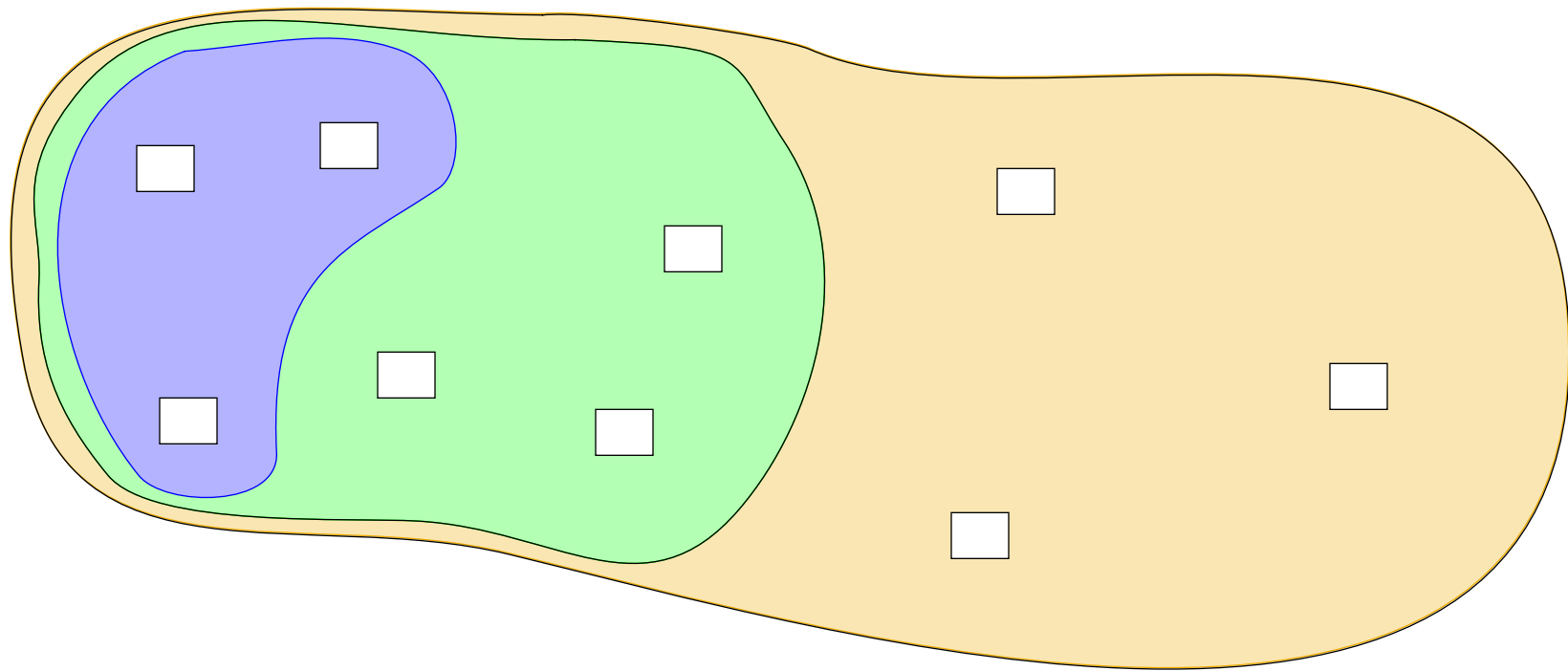
+1 (847) 538 4587

kermode@ccrl.mot.com

The Problem



How do we find the appropriate already-allocated scoped address(es) for a particular session at one or more scopes?



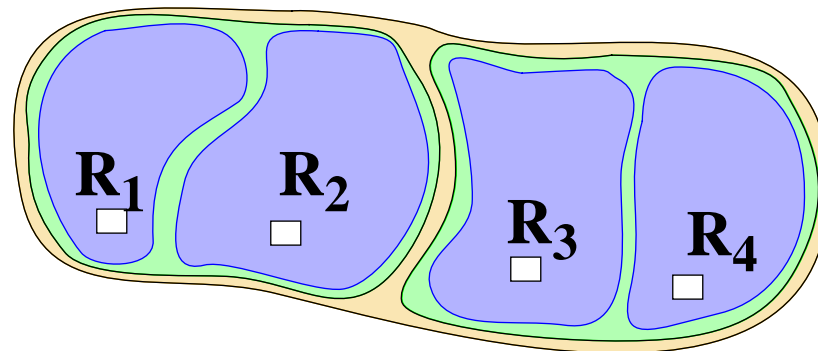
Interesting problem, why do we need this?

Helps solve the localization problem inherent within multicast.

TTL scoping has been shown to have problems.

Admin scoping affords better localization.

Admin scoping affords hierarchy through nesting



We need a dynamic means for discovering the various scoped addresses at each of the scopes.

Could use static allocation.....

Could use out-of-band messages, ala SDR.....

SADP is a strawman protocol that allows new members to a session to ask existing ones about which addresses to use at the various scopes. Aim is to devise a solution that

- 1) allows multiple addresses at different scopes.**
- 2) scales.**
- 3) is secure.**

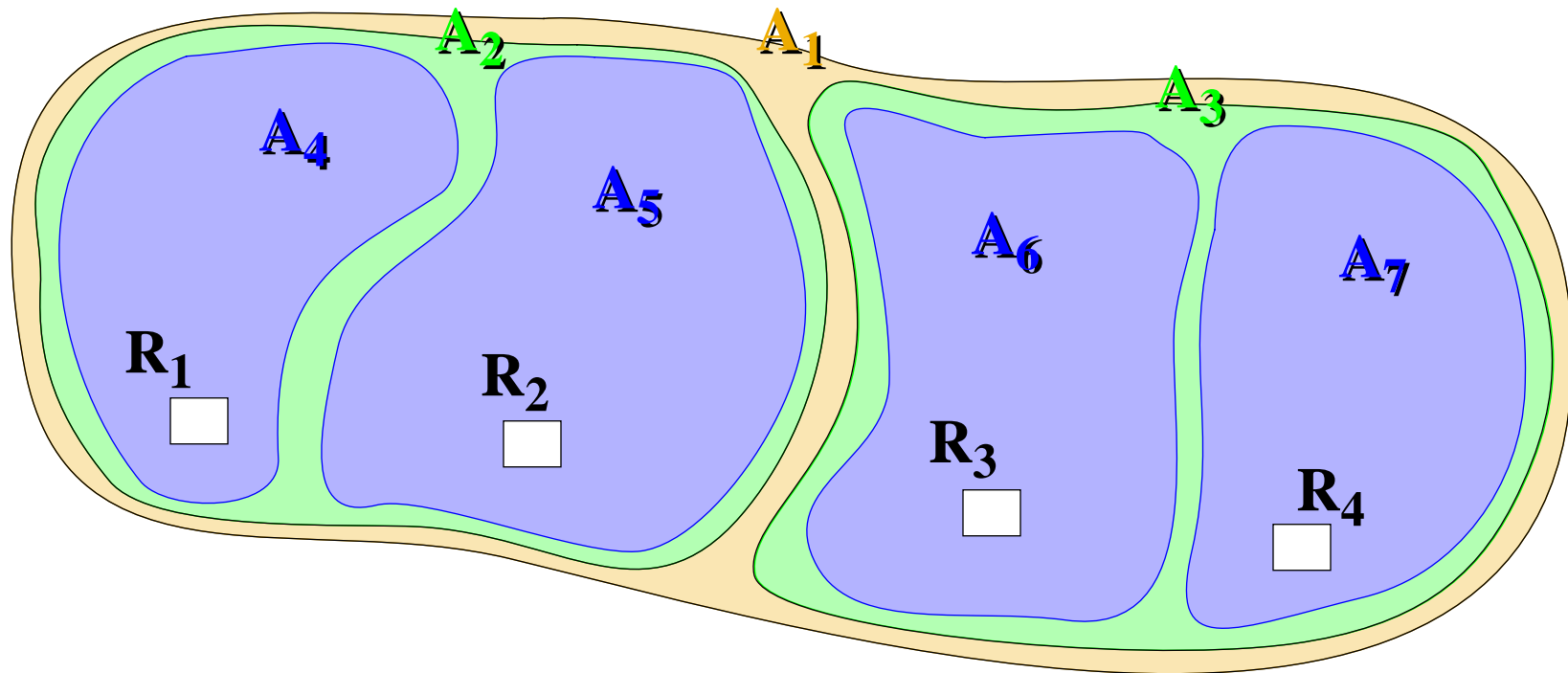
SADP does not aim to solve the multicast address allocation problem being tackled within malloc, it augments the proposed solution:

MZAP, MASC, AAP, MDHCP will be used to allocate new addresses.

SADP affords the means for their discovery by new members.

Complicating factors

- 1) Different zones with the same scope may use different addresses: R_1 uses A_1 , A_2 , & A_4 , R_3 uses A_1 , A_3 , & A_7



- 2) Must minimize traffic volume but at the same time find all relevant addresses associated with a session.



Solution

Use known information about the size and nesting properties of the various zones to do a truncated expanding zone search:

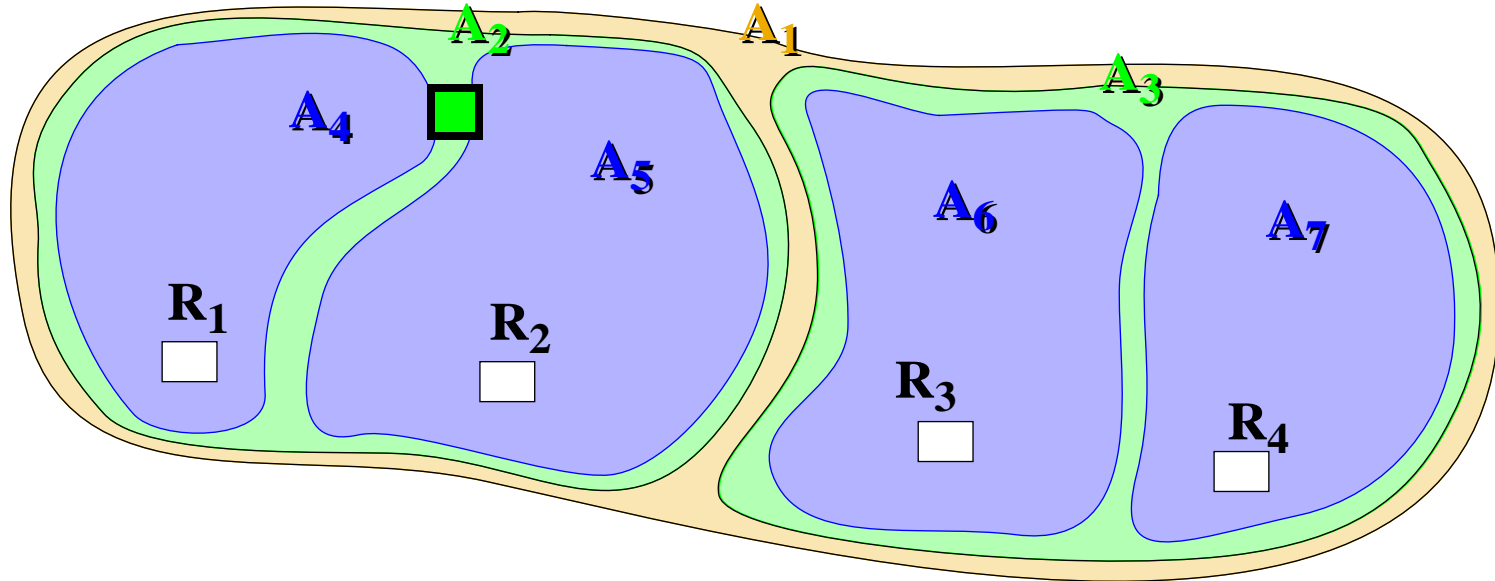
- 1) Start with the smallest scope (usually local).
- 2) Multicast request for information at that scope on a well known relative address (similar to MZAP).
- 3) If reply is heard stop, else increase admin scope, unless at second to largest scope.

Existing members use random delays for suppression.

One problem with this solution: Sparsely populated sessions will always increase scope to the largest possible value, thereby causing flooding.

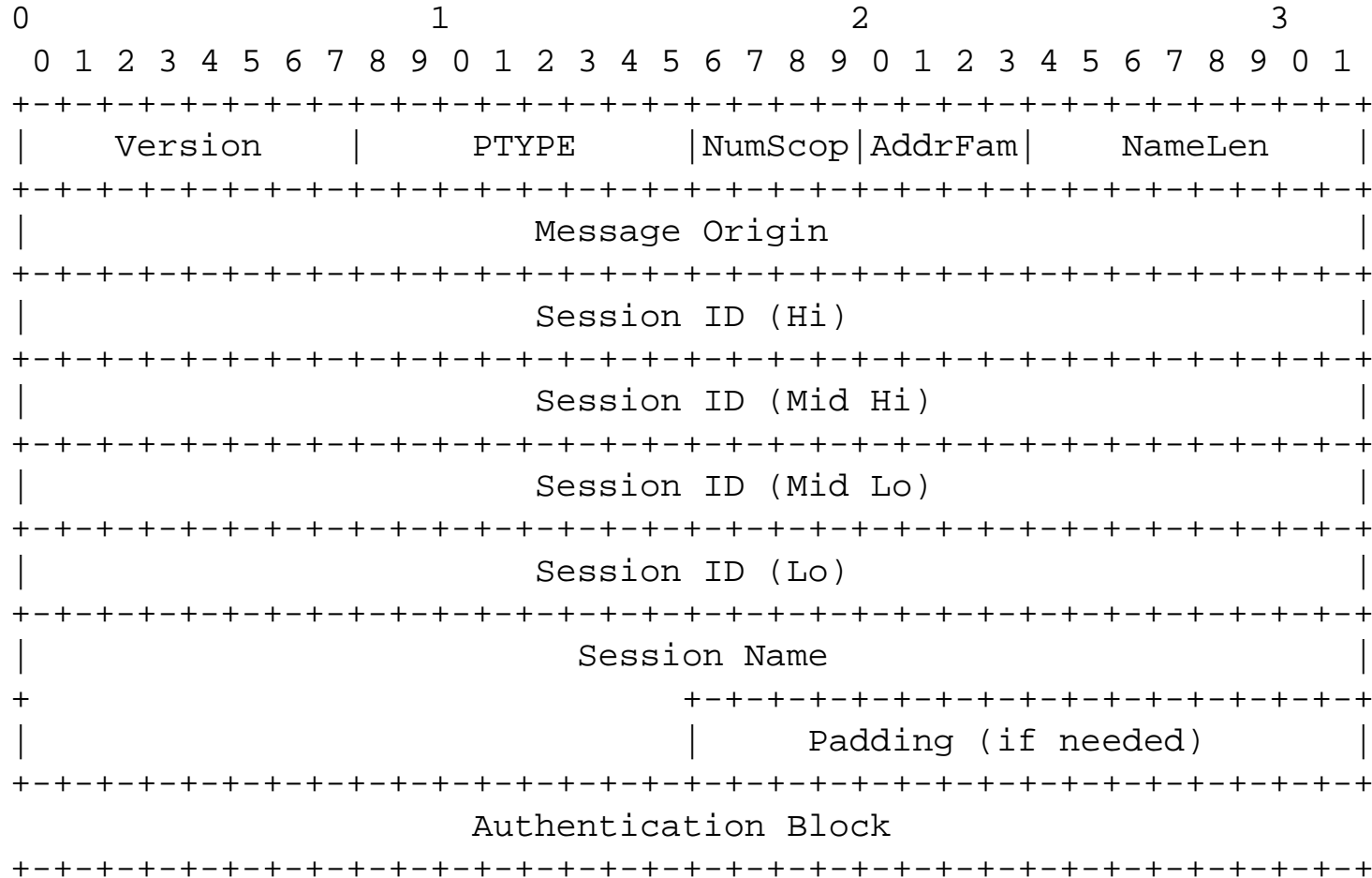
This can be fixed with one simple refinement: the addition of SADP servers that cache the replies heard at larger scopes.

SADP server now acts as proxy and returns information for scopes greater than or equal to that of request.



Packet Formats

SADP Header



SADP Request

Same as header

Response

