

March 11th, 1981
Danny Cohen
USC / ISI

Addressing and Routing

There are several approaches to addressing such as hierarchical and flat, but there is only one approach to routing: Take the best (usually the shortest) path.

The choice of addressing scheme which is both manageable and may support an efficient routing is the subject of this note.

Hierarchical addressing is probably the easiest to manage. The universe is divided into some small number of units, and each of which is assigned an address which is some kind of a string, typically numeric but not necessarily so.

If any of these units contains more than one member then the same process is repeated (iterated). The address of the subunit is appended somehow, typically to the end (but sometimes to the front) of the address of the parent unit.

This hierarchy is easy to manage such that addresses are uniquely assigned.

Typically in an hierarchy each unit is capable of communicating both with its ancestor and with all of its immediate descendants.

A default routing can always be easily derived from an hierarchical address. This route consists of going up to a common ancestor (parent) and then down to the destination.

Hence, an hierarchical address is always a route from the root (sorry about that).

This default routing requires only $N-1$ communication links which is the minimal connectivity for N nodes.

In this situation the default route is also the only existing route, up to trivial changes.

In reality, for efficiency reasons, the connectivity is richer than that. Typically, brothers are in direct communication, and a "top" (root) may not exist at all.

Because of the connectivity richness the routing process is non-trivial and requires that decisions be made.

Routing is the process of choosing a possible communication link, where there is more than one.

The other extreme of addressing is the flat space approach where locations have totally random dependency on their addresses, meaning that a fully instantiated table lookup must be used in order to locate an address.

The important feature of flat and random addressing is that it can conveniently support mobile users.

A simple, however expensive, way for implementing routing when flat addressing is used is by providing all the switches (where routing decisions have to be made) with tables containing all the required information about all the addresses.

In the case of flat addressing it is required that all addresses be globally unique. It is not required however that all addresses be known to all the switching nodes, provided that nodes can ask for, and get, information about addresses which were not known to them.

In the hierarchical case it is possible either that all addresses be specified in an absolute (complete) mode or in a relative (partial) mode. The former requires that all addresses be fully expanded, and have the same format regardless of where they are specified (e.g., 1-213-822-1511x104), and the latter supports short forms for local addresses, at the possible expense of additional control information for remote addresses. The same address shown before may be specified as 104 from inside ISI as 822-1511x104 from the STN at LA, as 9-822-1511x104 from UCLA, as 213-822-1511x104 from Palo Alto or as 010-1-213-822-1511x104 from London. Similarly, intra-office memos do not require long addresses, and intra-country mail does not require the country name to be specified in the address.

Any system which can handle perfectly flat addresses can take advantage of rich communication connectivity and can also handle multi-homing (destination with more than one address).

An hierarchical addressing scheme which uses only the default hierarchical routing may have difficulties in handling efficiently rich connectivity and multi-homing.

However, between these two extremes there is a wide spectrum of other possibilities.

The claim of the rest of this paper is that (i) hierarchical addresses are easier to manage, and (ii) if you can handle flat addresses - you can handle hierarchical addresses even better.

Hierarchical addressing is at least as good (in any respect) as flat one because flat addressing is a special case of hierarchical addressing, but not the other way round.

Suppose that there is a system which can handle very well flat addressing. Since the only requirement for the flat addressing is that addresses are unique, one can assign unique addresses in any way, for example in a very structured way, like hierarchical. The introduction of hierarchical addresses to the flat addressing scheme should not degrade its performance since the uniqueness was not disturbed.

Suppose further that all the addresses are examined at every switch. All addresses which result in the same routing decisions by both the flat addressing scheme and by the hierarchical addressing are marked, and the entries corresponding to them are removed from the flat addressing tables.

This causes no degradation of the flat addressing scheme (which was assumed to be a good one to start with) since the hierarchical addressing yields the same routing decision for these addresses. This process repeats at all levels, and only the "GCD" addresses are stored.

Hence, after removing these addresses the size of the tables decreases in all switches which probably may yield some performance improvement and increased capacities. In fact, the resulting scheme is now an hierarchical scheme with tables of exceptions which are handled in the best known way. The format of the exception table is discussed later.

Note that this results in uneven distribution of knowledge. At different switches different set of destinations are known. For example, all the traffic to the East coast may be treated in the same way when being far away from there, but as the message approaches its destination, it is likely that the granularity of the address examination changes such that more details are used. When leaving Tokyo the only part of the address which was used for the routing was the fact that the message is addressed to the States. After arriving at California, the Boston traffic may be treated separately from the Washington traffic, later the Boston traffic is examined even closer to distinguish the Cambridge from the Lexington traffic, and later the MIT traffic is isolated from the Harvard and the BBN traffic. Upon entering MIT the traffic is sorted further to the appropriate local MIT net, and later to its terminal destination.

This scheme allows the switches in Tokyo NOT to know about the internal structure of the addressing in the USA in general and at MIT in particular. It also helps the switches in California to take advantage of special HU (high utilization) trunks to the Boston area, if any.

Since this hybrid contains both the flat and the hierarchical addressing, multi homing can be handled as well as by flat addressing.

Another variant of this scheme is not to start with full tables of flat addresses (which are impossible to manage) but to treat the system as if it was a pure hierarchical one, but in addition to maintain a table of exception which is dynamically updated.

The entries in these exception tables should include both addresses, their granularity level and the associated routing decision. If, for example, addresses always consist of a sequence of bytes (of any size) the granularity level may be measured by the number of bytes which are used.

For example, a telephone exchange in Marina del Rey may know that the traffic for 1-213-828X and 1-213-821X is local, all the traffic for addresses 2X and 4X (Europe) should be routed to a certain cable which goes directly to Europe, the traffic to 1-212X be routed to a some cable which goes directly to NY, the traffic to either 1-617-49X or 1-617-253X should be routed to the HU trunk which goes to Cambridge, and similarly the traffic to 1-202-694-3049 should be given to a certain line which is connected to a certain destination, somewhere. All the rest of the traffic should be routed to downtown LA where smarter computers can solve the routing.

In the above example the various entries have different levels of granularity, varying from 1 (in the case of Europe) to 11 (in the last example).

Such a scheme yields tables of reasonable size, can benefit from any structure and logic/regularity (as opposed to randomness) of the communication subsystem, can handle multihoming, and has the capability to utilize special irregular HU lines and shortcuts.

Note that in this scheme the addressing may be highly structured (e.g., in a hierarchical way) but the routing is not necessarily so. The routing is hierarchical only when this is known to be the right thing, or when nothing else is known and the hierarchical routing is used as a default.

In summary, this is a way to benefit from the simplicity of hierarchical routing whenever possible, while being able to use the flat-addressing features when needed, on a case-by-case basis, without the need to maintain tables of unreasonable size.