Mobile phone tracking system description

Solving the fundamental problem of cellular networks: When a phone is called, find where it is located so that a connection may be established.

As we know each mobile phone that is switched on is connected to the base station which is nearest. These base stations are popularly called cell phone towers. Although sometimes we may be within range of more than one base station, each phone is registered to exactly one base station at any point of time. When the phone moves from the area of one base station to another, it will be de-registered at its current base station and re-registered at new base station.

Making a phone call. When a phone call is made from phone p 1 registered with base station b 1 to a phone p 2 , the first thing that the base station b 1 has to do is to find the base station with which p 2 is registered. For this purpose there is an elaborate technology framework that has been developed over time. You can read more about it on the Web. But, for now, we will assume that b 1 sends a query to a central server C which maintains a data structure that can answer the query and return the name of the base station, let's call it b 2 , with which p 2 is registered. C will also send some routing information to b 1 so that b 1 can initiate a call with b 2 and, through the base stations p 1 and p 2 can talk. It is the data structure at C that we will be implementing in this assignment.

A hierarchical call routing structure

We will assume that geography is partitioned in a hierarchical way. At the lowest level is the individual base station which defines an area around it such that all phones in that area are registered with it, e.g., all phones that are currently located in Bharti building, School of IT and IIT Hospital are registered with the base station in Jia Sarai. This base station also serves phones in Jia Sarai and maybe some phones on the Outer Ring Road in front of Jia Sarai. Further we assume that base stations are grouped into geographical locations served by a level 1 area exchange. So, for example, the Jia Sarai base station may be served by the Hauz Khas level 1 area exchange. Each level i area exchange is served by a level i+1 area exchange which serves a number of level i area exchanges, e.g The Hauz Khas level 1 area exchange and the Malviya level 1 area exchange may be both served by a South-Central Delhi level 2 area exchange. A base station can be considered to be a level 0 area exchange in this hierarchical structure. Given a level i exchange f , we say that the level i + 1 exchange that serves it is the parent of and denotes this parent(f). We will call this hierarchical call routing structure the routing map of the mobile phone network.

Maintaining the location of mobile phones

Every level i area ex-change, e, maintains a set of mobile phones, Se , as follows. The set Se is called the resident set of e. The level 0 area exchanges (base stations) maintain the set of mobile phones registered directly with them. A level i + 1 area exchange e, maintains the set Se defined as follows:

Se = Union (Sf) for all Sf s.t. parent(f) = e

i.e., the union of the sets of mobile phones maintained by all the level i area exchanges it serves. Clearly, the root of the routing map maintains the set of all currently registered mobile phones.

Tracking a mobile phone

The routing map along with the resident sets of each area exchange makes up the mobile phone tracking data structure we will be using. This data structure will be stored at the central server C. The process of tracking goes as follows.

• When a base station b receives a call for a mobile phone with number m it sends this query to C.

• If the root of the routing map is r, we first check if m ∈ S r . If not then we tell b that the number m is “not reachable.”

• If m ∈ S r we find that e such that parent(e) = r and m ∈ S e , i.e. we find the child of r which contains m in its resident set.

• Continue like this till we reach all the way down to a leaf of the routing map. This leaf is a base station b’ . The central server sends b’ to b along with the path in the routing map from b to b’

Queries to be supported:

• **addExchange a b** This should create a new Exchange b, and add it to the child list of Exchange a. If node a has n children, b should be its (n + 1) th child. If there is no Exchange with identifier a, then throw an Exception.

• **switchOnMobile a b** This should switch ON the mobile phone a at Exchange b. If the mobile did not exist earlier, create a new mobile phone with identifier a. If there is no Exchange with an identifier b, throw an exception.

• **switchOffMobile a** This should switch OFF the mobile phone a. If there is no mobile phone with identifier a, then throw an Exception.

• **queryNthChild a b** This should print the identifier of the Exchange which is the (b)th child of Exchange a.

• **queryMobilePhoneSet a** This should print the identifier of all the mobile phones which are part of the resident set of the Exchange with identifier a.

• **queryFindPhone a** This should print the identifier of the exchange re-turned by the findPhone(MobilePhone m) method. Here, m represents the mobile phone whose identifier is a.

• **queryLowestRouter a b** This should print the identifier of the exchange returned by the lowestRouter(Exchange e1, Exchange e2) method. Here, e1 and e2 represent exchanges with identifier a and b respectively.

• **queryFindCallPath a b** This should print the list returned by the routeCall(MobilePhone m1, MobilePhone m2) method. Here, m1 and m2 represent mobile phones with identifier a and b respectively. Successive entries in the list should be separated by a comma.

• movePhone a b This action should set the level 0 area exchange of the mobile phone with identifier a to exchange with identifier b. Throw exception if mobile a is not available, or exchange b does not exist.