**Social Network:** How would you design the data structures for a very large social network like Facebook or Linked In? Describe how you would design an algorithm to show the shortest path between two people (e.g., Me -> Bob -> Susan -> Jason -> You).

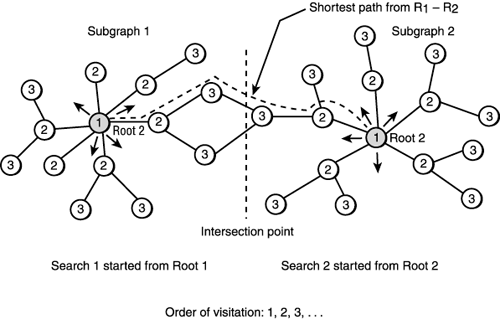
**Solution:**

The graph would be the ideal structure to handle the relations between various data.

**INFO:** Graph DB is a NoSQL based database where the data are stored in the form of relations with each other. It is ideal for the frequently alterable schema and the relation based once. It also ensures the ACID property which the other NoSQL database does not.

Now in the graph we need to find a way to do the search (Breadth wise or Depth wise). Breadth wise will be proper for this approach as we first find out the first degree of friends and then move on to the next. In order to increase the speed and decrease the steps we search the shortest path in the bi-directional way.

The step is greatly reduced in the bidirectional search



[img source](https://goo.gl/6zne3)

If the friend list is pretty huge then we need to process id of the user and the user will be split across many database (machines - based on their hashing). So we first determine the friend’s machine id and then hop to that machine and then find the friends list of friends. Here we can optimize by grouping: first process the list of 1 degree friends and calculate all the machine and group them by machine name and then process each machine accordingly