Scenarios:

1. Consider a complete family tree. Just choose a random one with 5 generations or more, filled out. Include siblings among multiple generations and especially the person who’s generation will be worked with. Add another sibling to such person and install matching links.

   With master person class the above scenario becomes too complicated.

2. Same complete family tree. Find father of person in question, be able to update all relevant links/data on input.

   Family concept with "unit" concept fits much better than massive person class.

3. Same (or different) family tree. Find all (or even handle multiple) marriages of person p. What is required and is there a change in the data structures that would allow us to do this better or faster?

4. Search through family tree, looking for death dates and misc information. Since this will be a comprehensive search, is there changes in the data storage or class layout that would reduce overhead?

5. Check for missing vital information in the entire database, or in p's know family tree. Again, an intensive operation, are we sure that nothing can be done to speed up the process. Do the current necessary collaborators make sense?