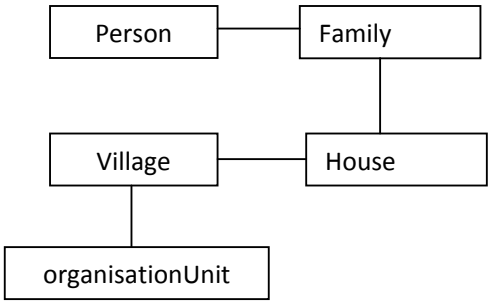
The requirement is to have a system capable of registering and coordinating visits for specific health programs (FP, ANC and Immunization at least to start with). And below is a little break-up of this.

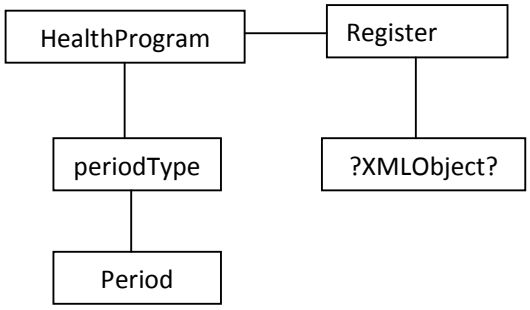
1. **Individual/Patient Management Module** –the requirement here is to systematically manage (as well as track) individuals. Very much similar to orgunit hierarchy operation and module in DHIS2.

    - Person
        - First_name – string
        - Last_name – string
        - Age – (integer) or shall we make it dateOfBirth? Because at somepoint we will register birth!
    - Family
        - Husband – person
        - Wife – person
        - Daughters – Set<Person>
        - Sons – Set<Person>
        - Address – house
    - House
        - HouseNo – string
        - GpsLocation – string
    - Village
        - Name – string
        - Children – Set<House>
        - Parent – organisationUnit



2. **Health Program Management –** This is something (I would say) similar to advanced dataset management. This module is to enable users manage programs (for e.g. Family Planning, ANC, Immunization, Line-List of Birth, Line-List of Death, …)

    - HealthProgram
        - Name – string
        - Freqency – periodType
        - ProgramPhase – Set<Register>
    - Register
        - Name – string
        - Header – ?XMLObject?
        - Footer – ?XMLObject?
        - Columns – ?XMLObject?



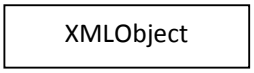
3. **Register Management –** Probably this is the most complicated part of the system. And I would be happy if Saptarshi can pull something from OpenMRS especially from the XML_Reporting module. Here things need to be saved in plain XML raw data – which I think is quite ok – because there will be a wide range of objects as part of a register (for example - person, dataelement, period, house, date, village,….)

    - Functionalities expected from this module

        - Add/Edit/Remove register

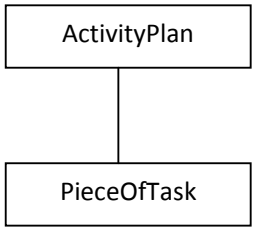
- Assign objects to register
    - objectes could be – dataelement, individual, house, village, period, date,…
- Arrange objects or create a register form
    - Depending on the type of objects assigned to the register a form need to be designed using different dataentry boxes (boolean, calendar, text/integer box, dropDown, hierarchy browsing, …)

- XMLObject
    - Set<dataElement>
    - Set<houseHoldVisit>
    - date
    - village
    - …



4. **Activity Plan Generator –** this module is to cross-check services/programs provided and forecast the next programPhase – activity plan. The forecast is expected to have a list of tasks detailing whatServiceToProvide(which phase, registerRelatedInformation, forWhom(individual), whereToGo (village, houseNo)

- ActivityPlan
    - Owner – person (Health Extension Worker)
    - Supervisor – person (Medical Officer)
    - Date – date
    - Activities – set<PieceOfTask>

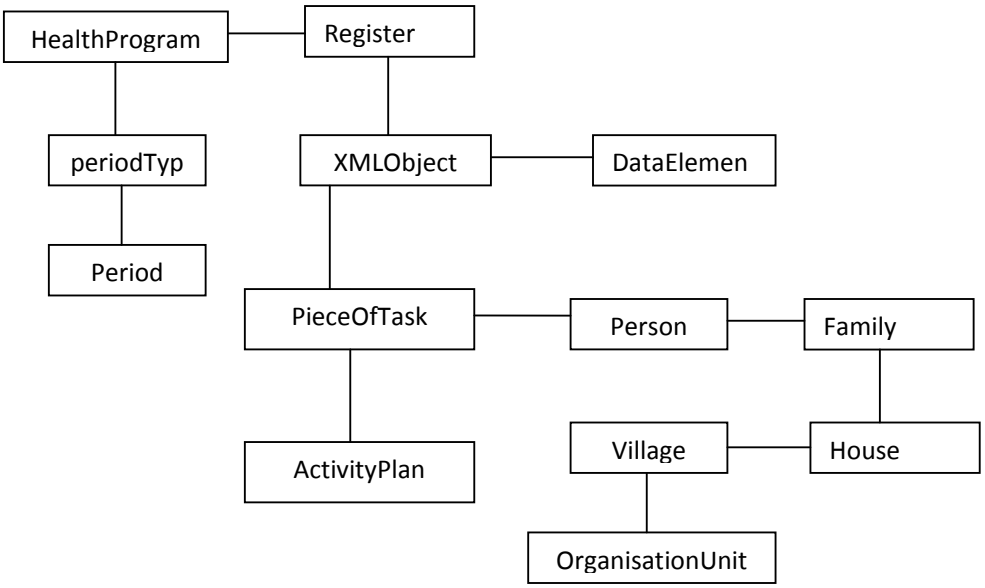


- PieceOfTask
    - Where – house
    - When – date
    - ForWhom – person
    - What – houseHoldVisit

5. **Aggregation Module**

- As much as possible a ?freeText? (combinations of objects/dropDowns/,…) query engine. Again there is a lot to take from OpenMRS cohort manager.

- Queries should be mapped to existing objects (or their combinations) and saved for later use.


HealthProgram
Register
periodTyp
XMLObject
DataElemen
Period
PieceOfTask
Person
Family
ActivityPlan
Village
House
OrganisationUnit