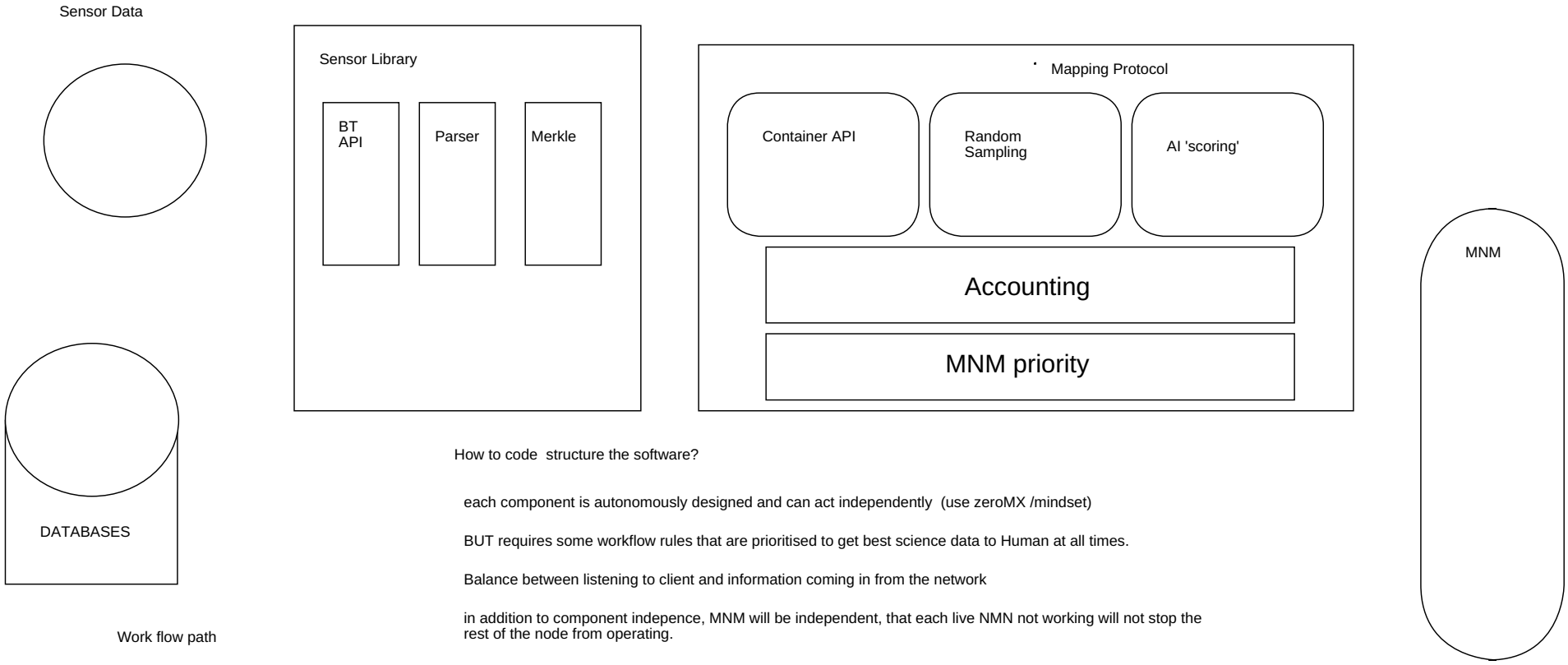
Model::Main

1. On awakening perform checks, network, sensors (sensor library), databases, AI (mapping protocol).

2 Assume one sensor and one NMN (compute theory) two person ptop network

3. UI listening for Human attention, while that going on, check to see if any MNM are live in the client, if so check if MNM requires updating, if yes, update, while that was giong on sensor data was being updated if required, then run compute and update feedback required to Human.



How to code structure the software?

each component is autonomously designed and can act independently (use zeroMX /mindset)

BUT requires some workflow rules that are prioritised to get best science data to Human at all times.

Balance between listening to client and information coming in from the network

in addition to component indepence, MNM will be independent, that each live NMN not working will not stop the rest of the node from operating.

Work flow path

1. Client and node downloaded and auto installed.

2. Public / Private Key setup or defaults if existing use

3. Register and connect to a sensor, data inflow, parsed, merkled and accounting entries on a blockchain.

1. Seed peer to peer connections made with network

2. Instantly listening for random sampler requests (checking if suitable for this client/node). At same time reivewing existing 'consensus' science and providing list for this node

3. On going listening and mangement of consensus building local to node and participate in network.

NOTE ON ACCOUNTING: need to decide on minimum level of entries to blockchain to provide reproducibility, what data from what sensor at what time, with what code under what compute environment