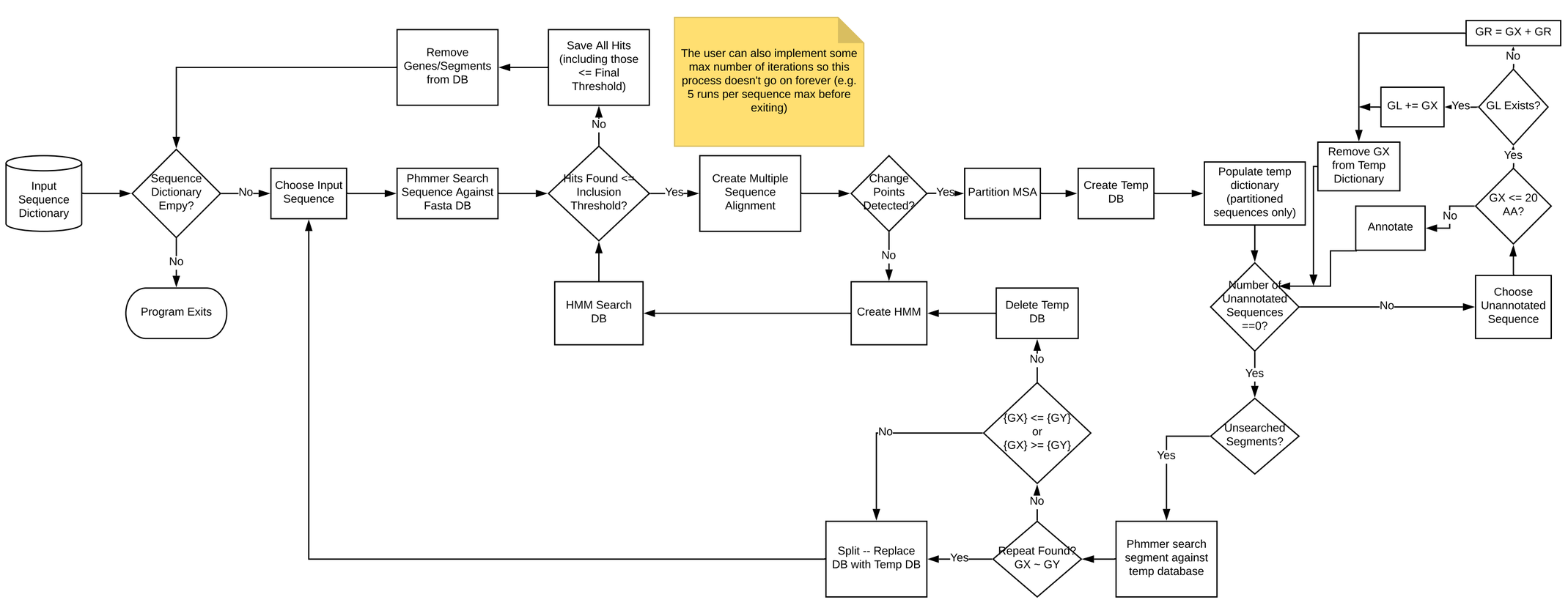
This is the general idea behind the HMMER Pipeline. There may still be some kinks to iron out. One of the more challenging things about this pipeline is the data storage. For future use, it might be worth revisiting how databases are updated and how sequences collect hits. Specifically, should all hit sequences collected by the first input sequence go through the pipeline and be collected into their own homology group and once the last sequence passes through as part of this little collective, all are removed from the input dictionary before moving on to the next collective.

One idea that was tossed around was arranging the input sequences by length.

A note on annotation. An input sequence's name is listed as G. When a gene is partitioned, it is partitioned into either GC,GL,and GR or GR and GL depending on the number of change points. The set of hits associated with any of these segments is {GX}, where X=L,R, or C.

To run the pipeline, an inclusion and a final threshold should be set. The inclusion threshold should generally be stricter than the final threshold. The inclusion threshold determines which sequences are included in creating the HMM. The final threshold is when the pipeline terminates and the remainder of the hits (satisfying the final threshold) are returned to the user.



The conditions for splitting are either if a repeat is found (a gene hits itself, determined by one segment of a gene hitting another segment of itself), or if something new is found by partitioning the genes. This is equivalent to searching the sets of hits. This should be done on the set of genes without annotations (removing L's, R's, and C's) and seeing if one set is a subset of the other.

For example, if G1L hits G2 and G1R hits G2 and G3, then a split is not performed. However, i f G1L hits G2 and G1R hits G3, then a split is performed.