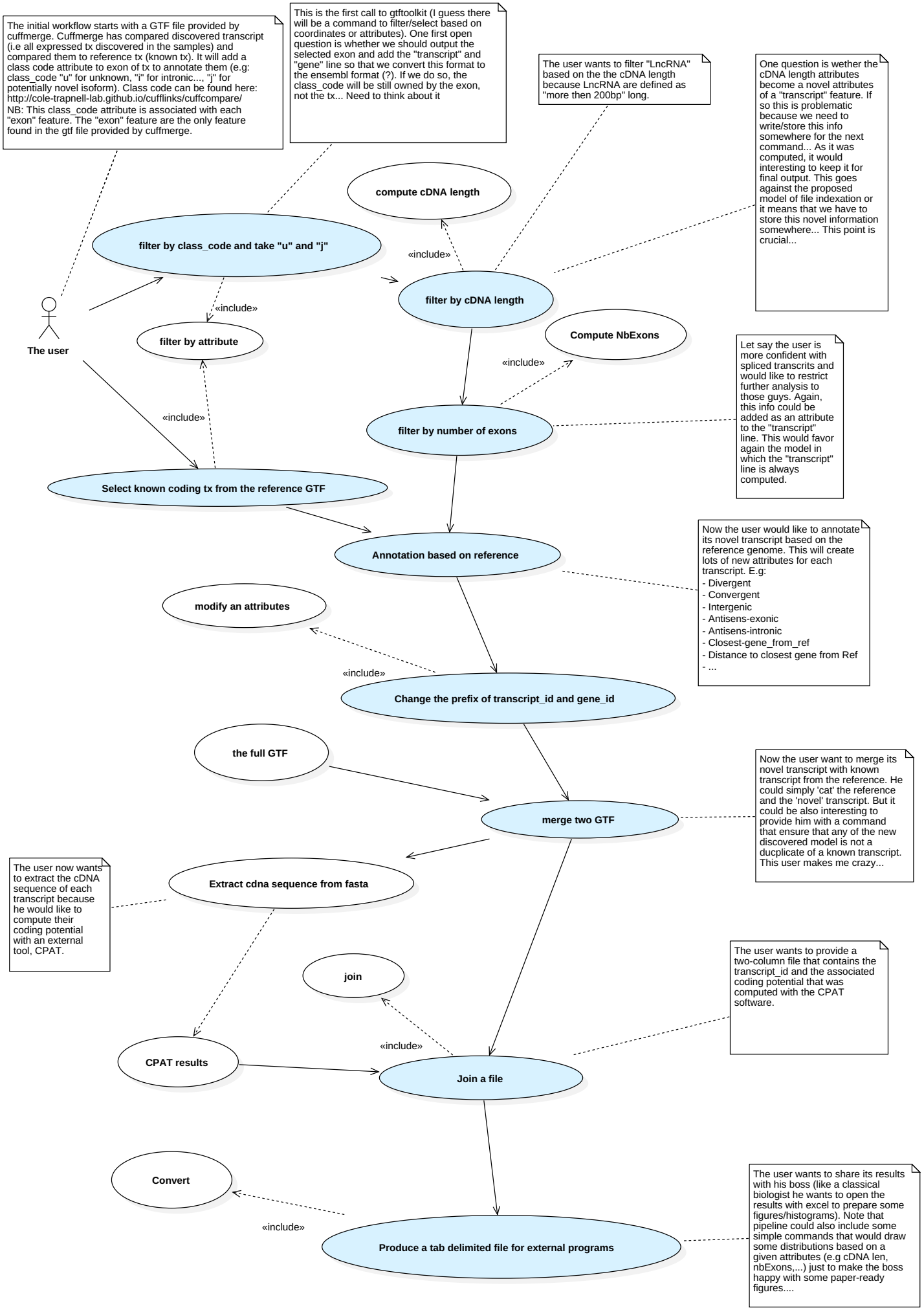
The initial workflow starts with a GTF file provided by cuffmerge. Cuffmerge has compared discovered transcript (i.e all expressed tx discovered in the samples) and compared them to reference tx (known tx). It will add a class code attribute to exon of tx to annotate them (e.g: class_code "u" for unknown, "i" for intronic..., "j" for potentially novel isoform). Class code can be found here: http://cole-trapnell-lab.github.io/cufflinks/cuffcompare/
NB: This class_code attribute is associated with each "exon" feature. The "exon" feature are the only feature found in the gtf file provided by cuffmerge.
This is the first call to gtftoolkit (I guess there will be a command to filter/select based on coordinates or attributes). One first open question is whether we should output the selected exon and add the "transcript" and "gene" line so that we convert this format to the ensembl format (?). If we do so, the class_code will be still owned by the exon, not the tx... Need to think about it
The user wants to filter "LncRNA" based on the the cDNA length because LncRNA are defined as "more then 200bp" long.
One question is wether the cDNA length attributes become a novel attributes of a "transcript" feature. If so this is problematic because we need to write/store this info somewhere for the next command... As it was computed, it would interesting to keep it for final output. This goes against the proposed model of file indexation or it means that we have to store this novel information somewhere... This point is crucial...
compute cDNA length
filter by class_code and take "u" and "j"
«include»
filter by cDNA length
The user
«include»
filter by attribute
Compute NbExons
«include»
«include»
filter by number of exons
Let say the user is more confident with spliced transcrits and would like to restrict further analysis to those guys. Again, this info could be added as an attribute to the "transcript" line. This would favor again the model in which the "transcript" line is always computed.
Select known coding tx from the reference GTF
Annotation based on reference
Now the user would like to annotate its novel transcript based on the reference genome. This will create lots of new attributes for each transcript. E.g:
- Divergent
- Convergent
- Intergenic
- Antisens-exonic
- Antisens-intronic
- Closest-gene_from_ref
- Distance to closest gene from Ref
- ...
modify an attributes
«include»
Change the prefix of transcript_id and gene_id
the full GTF
merge two GTF
Now the user want to merge its novel transcript with known transcript from the reference. He could simply 'cat' the reference and the 'novel' transcript. But it could be also interesting to provide him with a command that ensure that any of the new discovered model is not a ducplicate of a known transcript. This user makes me crazy...
The user now wants to extract the cDNA sequence of each transcript because he would like to compute their coding potential with an external tool, CPAT.
Extract cdna sequence from fasta
join
The user wants to provide a two-column file that contains the transcript_id and the associated coding potential that was computed with the CPAT software.
«include»
CPAT results
Join a file
Convert
«include»
Produce a tab delimited file for external programs
The user wants to share its results with his boss (like a classical biologist he wants to open the results with excel to prepare some figures/histograms). Note that pipeline could also include some simple commands that would draw some distributions based on a given attributes (e.g cDNA len, nbExons,...) just to make the boss happy with some paper-ready figures....