Main processing steps for a new load of SoundTrap data.

1. Process soundtrap data with PAMGuard to extract soundtrap clicks into binary data files. You may be running other detectors too, but this is only dealing with the soundtrap click detector output.
2. For each dataset (comprising a PAMGuard database and a folder of binary data) you need to run two separate tasks. The first is called clusterclicks, which goes through all binary data one day at a time and looks for clusters of clicks which have similar spectral properties. You then run classifyandwrite which reloads the extracted clusters, classifies them, then writes the results to the PAMGuard database as click events. These then become available next time you open that database in PAMGuard viewer mode.
3. If you’ve multiple similar datasets, then you can set up additional scripts to loop through the various sets. There are examples of this in clusterallclicks and classifyallclusters. Depending on how your data are organised, you may need to rewrite these functions.
   1. Run cluterallclicks, which loops over soundtraps and calls clusterclicks for each dataset. Arguments to clusterclicks are a folder of binary data, an output folder for click clusters, an interval for clustering (10s) and a minimum count per second, i.e. if count = 1, and interval is 10s, then 10 clicks will be required for a cluster.
   2. Run classifyalclusters. This takes the clusters from the previous step and classifies them. Clusters that are close in time and of the same species are then merged. Merged clusters are written to the database as events, populating both the ST\_Click\_Detector\_OfflineEvents and the ST\_Click\_Detector\_OfflineClicks tables
4. Run morlaislookup for each database (loops through databases In checkmorlaislookup) to correctly populate lookup tables in every database.
5. Run gatehrallnoiase / gathernoise to extract noise statistics from the start of every click in the binary data and write summaries for each day into additional data folders.
6. Run makeminutes to count porpoise positive minutes for varying noise thresholds and write to positive minutes database.