Wyrm v0 Structure

Branching Protocol

1. Branch current repository to experimental
2. Remove trash from develop branch
3. Refactor repository in `develop` to the scheme below.

Directory/Code Structure

* Readme.md
* License.md
* Environment
  + wyrm.yaml
  + wyrm\_silicon.yaml
* example\_data
  + Port\_Townsend\_M4.3
    - bulk.mseed
    - station.xml
* wyrm
  + core
    - base.py
      * class Wyrm(object)
    - mltrace.py
      * class MLTraceStats(obspy.core.trace.Stats)
      * class MLTrace(obspy.Trace)
    - mlstream.py
      * class MLStreamStats(obspy.AttribDict)
      * class MLStream(obspy.Stream)
  + streaming
    - mltracebuffer.py
      * class MLTraceBuffer(MLTrace)
    - mlwindowstream.py
      * class MLWindowStreamStats(MLStreamStats)
      * class MLWindowStream(MLStream)
  + processing
    - window.py
      * class WindowWyrm(Wyrm)
    - mlpick.py
      * class PredictWyrm(Wyrm)
      * class PickWyrm(Wyrm)
    - inplace.py
      * class MethodWyrm
      * class OutputWyrm(MethodWyrm)
  + coordinating
    - sequence.py
      * class TubeWyrm
      * class CanWyrm
    - module\_ops.py
      * class HeartWyrm
    - io
    - ew\_ring.py
      * RingWyrm
      * EarWyrm
    - client.py
    - disk.py
  + module\_examples
    - ring2ring
      * prediction\_example.py
      * prediction\_ensemble\_example.py
    - disk2disk
      * prediction\_example.py
      * prediction\_ensemble\_example.py
  + util
    - logging\_util.py
    - pyew\_util.py
    - time\_util.py
    - input\_compat.py
    - seisbench\_util.py
    - feature\_extraction.py