

# Design notes 5-27

## Current status:

Multi-task training is working using interleaving dataset reader, off the shelf train command and off the shelf trainer. For metalearn-trainer, I'm copying large chunks of the functionality of the default gradient descent trainer. The `from_partial_objects` method for the meta trainer should be finished though there are some issues with regard to the commented out logging-related properties. The metatrainer needs to be modified to select data from the appropriate dataloader (currently it's using the old iterator interface)

## Design considerations:

I'm not using a mingler anymore for the multi-task trainer as the interleaving dataset reader handles that functionality. However, it cannot keep the datasets separate (e.g. if you have multiple dataset sources you want to sample from). These are the sampling strategies we would want to support:

- random homogeneous batches
- weighted task sampling
  - evenly weighted
- I think this should be done by writing a new batch sampler to either replace or augment the existing homogeneous batch sampler

As for the [discourse post](#), the first option is more akin to what I'm suggesting. However, the standard gradient descent trainer still can't be used as we need to do some metalearning. This would require changes to the inner loop logic in the metalearning trainer

the second option is how the older, iterator based metatrainer worked. This is possible and requires minimal changes to metatrainer code but to me seems less in line with existing allennlp implementations. In addition, this would mean multi-task training and meta training would require different specs in futl which seems messy.

---

Revision #1

Created Thu, May 28, 2020 1:24 PM by [kenneth](#)

Updated Thu, May 28, 2020 1:25 PM by [kenneth](#)