Merging metadata

This document goes over some of the issues concerning the harmonization and merging of resource metadata, particularly when we plan to use that metadata as filters or inputs in the context of an analysis workspace.

Background/context

Initially, we thought that each file/`Resource` should have metadata associated with it. This metadata includes:

- `ObservationSet`: The set of samples or `Observations` that a file contains. For instance, given an expression matrix with N columns, this would be the N column names. For files where we have a 1:1 association with an `Observation` (e.g. a FastQ file), the `ObservationSet` contains only a single item. The `Observation` instances nested within the `ObservationSet` are used to hold additional data about each sample. That is, the expression matrix can have the *name* of the sample, but cannot tell us its phenotype, treatment group, etc. The `Observation` data structure allows us to associate a set of attributes.
- `FeatureSet`: Similar to above. Here, we imagined (again, using the example of an expression matrix) that each `Observation`/sample would have a number of `Feature`s associated with it. For example, an `Observation`/sample has m genes. Similar to `Observation`s, our `Feature`s can carry additional information about the gene (such as its status as a putative oncogene, its mutation status, etc.)

The whole idea of the metadata concept was to provide the user with the ability to filter their data during the analysis. For instance, one can say, "give me all `Observation`s/samples from the treated group" and we can iterate through the `ObservationSet` associated with their `Workspace` and return that subset.

Similarly, if we perform, say, a differential expression analysis, we can then ask for all the genes that passed significance at some threshold (e.g. 0.05). In our nomenclature, this amounts to saying, "give me all `Feature`s/genes where its `pvalue` attribute is less than 0.05"

Where we have some problems

Regardless of the analysis, the `ObservationSet` within a `Workspace` is fairly static. We can imagine adding attributes to each `Observation`, but an analysis operation does not fundamentally change any thing about each `Observation`.

However, the situation is different for `Feature`s. Imagine running two different statistical tests for differential expression. The analyses produce similar tables.

From analysis A:

gene	logFoldChange	pval
geneA	2.2	0.02
geneB	-0.2	0.05

From analysis B:

gene	lfc	pval
geneB	-0.03	0.1
geneA	2.03	0.00005

Above, both would have p-values representing the significance of differential expression.

Thus, depending on the context, the "attributes" of geneA or geneB would be different. There is no immediately sensible way to handle this in a general way.

Possible fix

Recall that the primary reason for defining `FeatureSet`s and `ObservationSet`s is to provide a way to filter down to samples or genes of interest. We can use `ObservationSet`s to set up a contrast and we could use a `FeatureSet` to hold a set of genes that are important (e.g. because they are differentially expressed, or maybe because they are related to some biological function).

If we restrict the selection of `FeatureSet`s to the context of a single `Resource`, then we can avoid this issue of "conflicting" `Feature` attributes. One drawback is that the user is required to remember exactly how the `FeatureSet` was created. For instance, the set of selected genes may be different in analysis A and analysis B and the user should know the file from which they originated. For this reason, we should have a way for the user to name/label the `FeatureSet`s.