Fedora 6 Layer Responsibilities and Interactions

Notes from a discussion between Jared Whiklo, Ben Pennell, Bethany Seeger and Danny Bernstein

In this design we have

**Http Layer**

1. Translate the incoming URI to an internal Fedora ID.
2. Get or create a transaction.
3. Call a **wrapper service** for the action (ie. GET|POST|PUT) with the following arguments:
   a. Fedora ID
   b. Transaction
   c. All HTTP headers
   d. The request body

**Service Layer**

Wrapper service will:

1. Load the FedoraResource from the persistence layer.
   a. Throws Exception if expecting to load something and can’t find it.
2. Determine appropriate internal service to call based on RDFSource, NonRdfSource, and NonRdfSourceDescription types
3. Call the internal service with:
   a. FedoraResource
   b. Transaction
   c. All HTTP headers
   d. Request body

**Internal service**

1. Validate needs headers and body for request type
2. Validate permissions
3. Create a OperationBuilder using the OperationFactories
4. Populate the Operation with the needed information using the OperationBuilder
5. Call the persistence layer with the Operation.

Persistence Layer will:

1. Take the operation and execute it (if possible) in the scratch/transaction space.

Additional thoughts:

For DELETEs:

1. The delete service holds the knowledge to delete associated resources (ie. ACLs, Mementos, Timemaps, NonRdfSource descriptions) and resources ldp:contain-ed
2. The delete service follows the chain of ldp:contain -ment to traverse down the tree until there are no more children, then begin deleting back up.
3. Each delete is a single operation to the persistence layer. (ie. Deleting an RDFSource is 3+ deletes (resource, ACL, Mementos, Timemap))