**ActiveMQ and Skye TaskManager thoughts**

-JDBC: Should be pretty easy to integrate, the one issue I can see at the moment is that I haven't found a lot of info on how to create a JDBC persistence adapter dynamically through code, everyone seems pretty determined to do all that configuration through xml. Not to say it isn't possible, I just haven't found it yet.

-Task timeout/failure: Matt, I know you mentioned this as a particular concern. It looks like if we use a failover Broker along with a TransportListener (which both seem to work together to ensure that a connection is open, and reconnecting or connecting to a different broker if needed), that should help. You can set a timeout option this way as well, although I'm not sure right now about how it handles long running jobs or if there's some kind of default timeout. More on those are here if anyone wants to look <http://activemq.apache.org/failover-transport-reference.html>

I think we could incorporate this into Skye by basically writing a general Producer and Receiver class, then (based on what was sent or received), calling the more specific Skye tasks (ArchiveTasks, IngestTask, etc). So the parts would be something like this:

-Manager: The closest equivalent to our current TaskManager, this is where all our current Task information (store, channel, etc) would be sent to. This is also the part that would send and receive info from the JDBC database

-Producer: Would get created whenever a user sets up a job, gathers the necessary information, creates a new Task, serializes it, and sends it to a receiver

-Receiver: Would de-serialize the information sent from the Producer and then create a new, more specific Task (think the getTask method in InMemoryTaskManager)

I don't think these parts would necessarily need to be separate (the manager and producer in particular can probably be one and the same from what I'm seeing in the various code examples I've been reading), but these are my initial thoughts.