* Plumbing
  + Add retry logic, and timeouts
  + Plan out all the various states
  + Plan out return codes
  + Add “Names” for events for easy visualization?
  + Learn Cassandra partitioning
* 3 versions:
  + Desktop (for prototyping)
  + Synch (http calls)
  + A synch (Kafka?)
* Add "registered" services:
  + Docker
  + Kubernetes
* Add Admin GUI
  + Logs/Traces
  + Replay logic
  + Analyze replay to see how to keep idempotent
* Add analytics
  + Does the engine have to assign UIDs, or can the local library?
* Client side libraries:
  + Java
  + Python
  + Node
  + Etc.
* Create best practices paper
  + On idempotent services
  + Process domains (APIs, Adapters, Process, etc.)