

# INTEGRATION & USER INTERFACE(S)

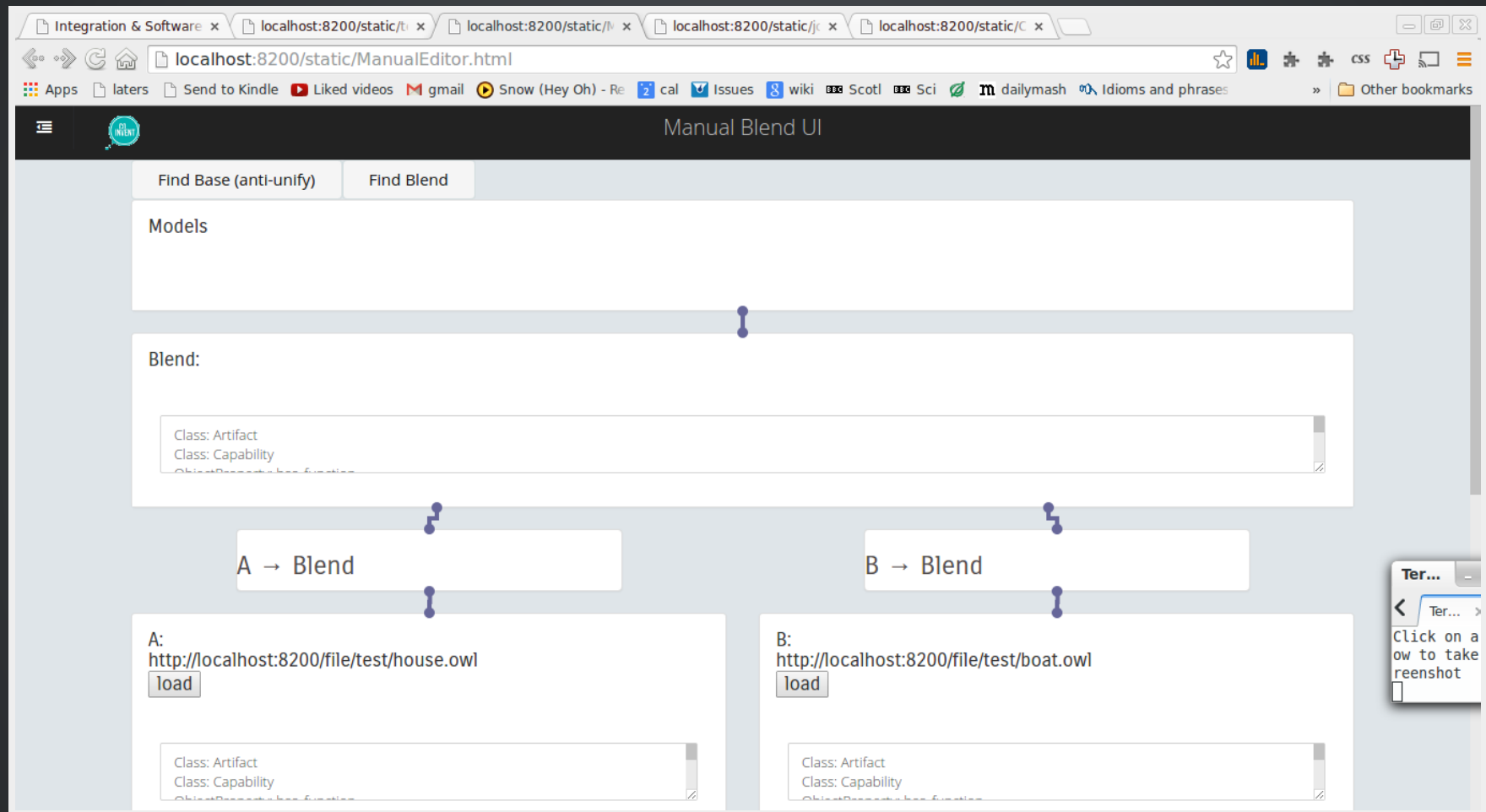
# WHAT DO WE NEED?

- File format (hopefully DOL)
- Base/analogy finder which handles ontology formats (hopefully HDTP)
- Push-out calculator: HETS ✓
- Consistency checker (hopefully HETS)
- Proof system supporting interactive/scripted proofs (hopefully HETS)
- Amalgams finder (TBD)
- Model synthesis (TBD)
- Quality metrics (TBD)

# INTEGRATION: WHAT WILL WE BUILD?

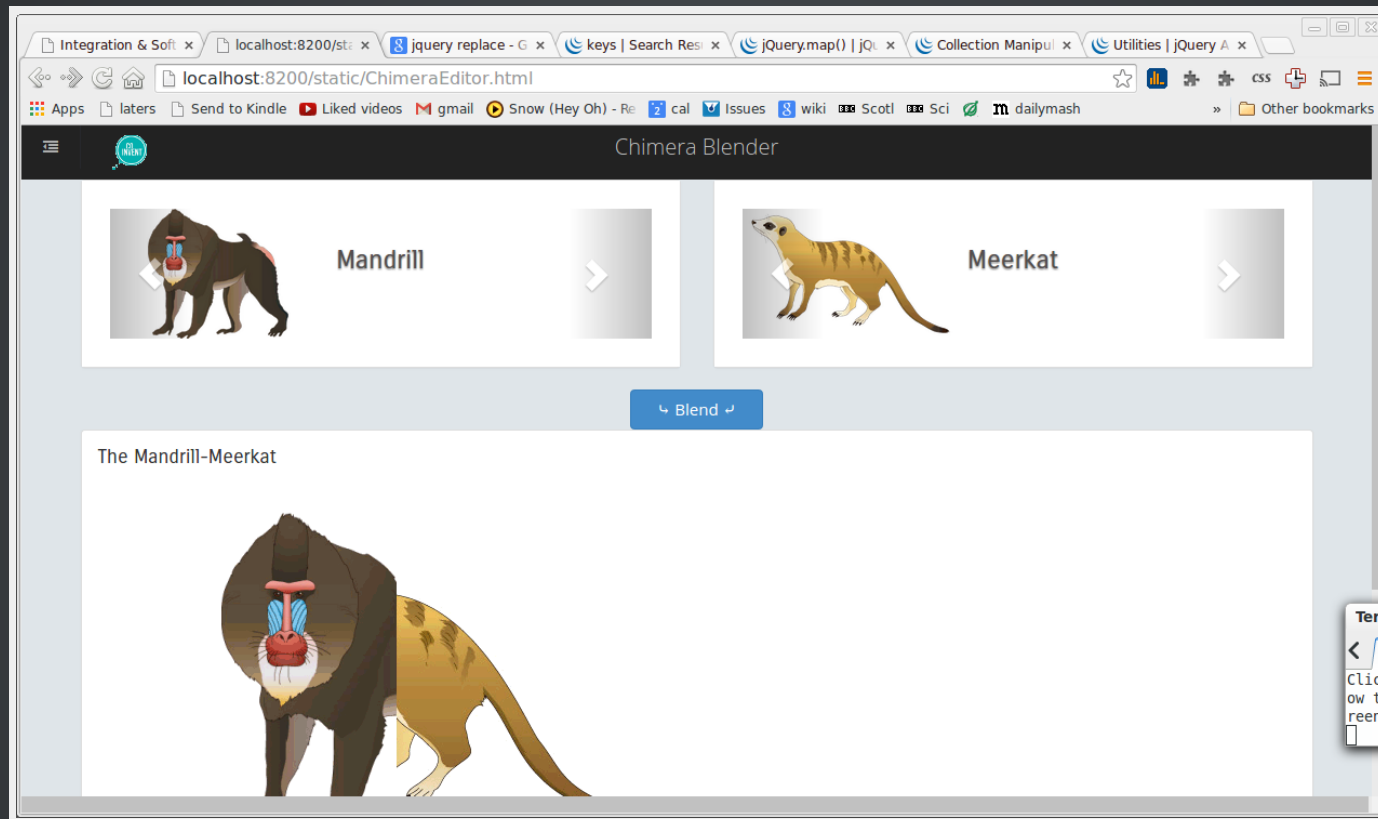
- Web-server with an http API for access to components
- Web-server for managing tasks -- human or computer
- Javascript client library for this API
- Interface and plugin architecture: easy to add new reasoning engines
- De-centralised: concepts referenced by URL -- can be stored anywhere
- ...reasoning engines too (no ties to one server)

# GENERIC USER INTERFACE



Too cumbersome!

# MULTIPLE USER INTERFACES



- Domain-specific is best
- The Javascript client library makes this easy (for a certain value of "easy")
- Anyone can build a Coinvent UI