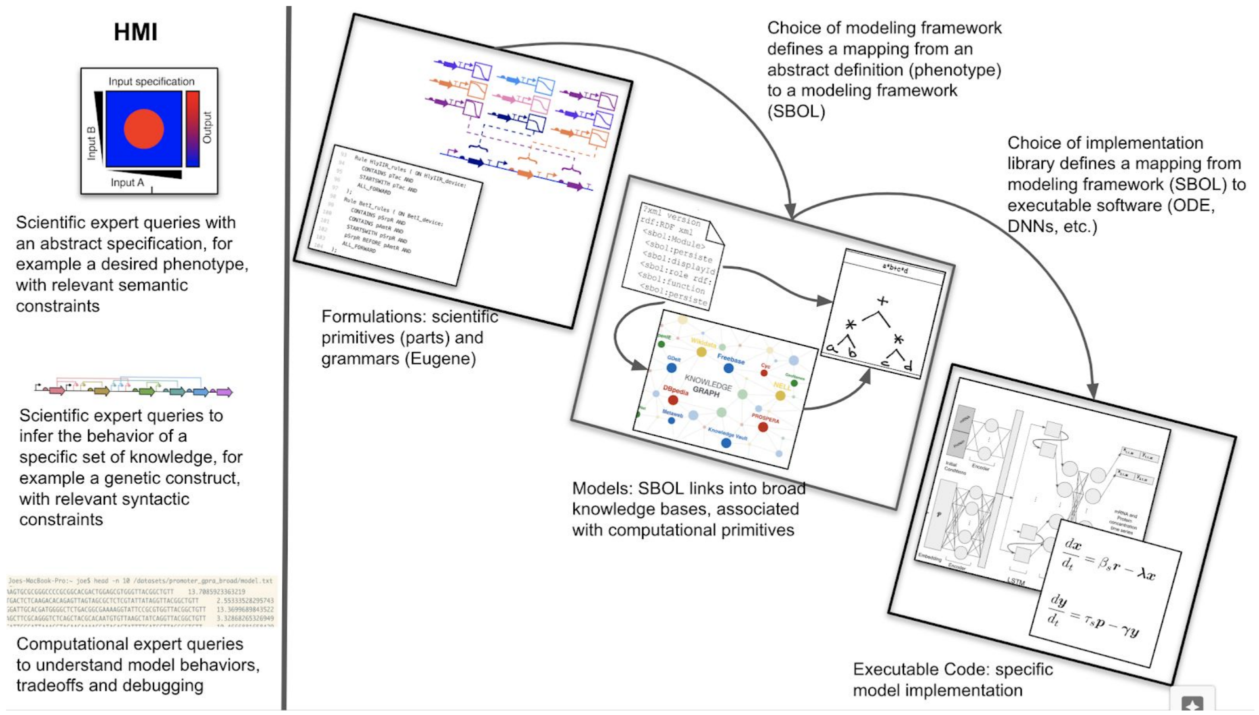
Overview

At the conclusion of the Phase II kickoff, Joshua asked teams to deliver a summary description of our project in the agreed revised representation, a practical narrative use-case example for the ASKE stack and other program questions around working groups, collaboration, and future demos.

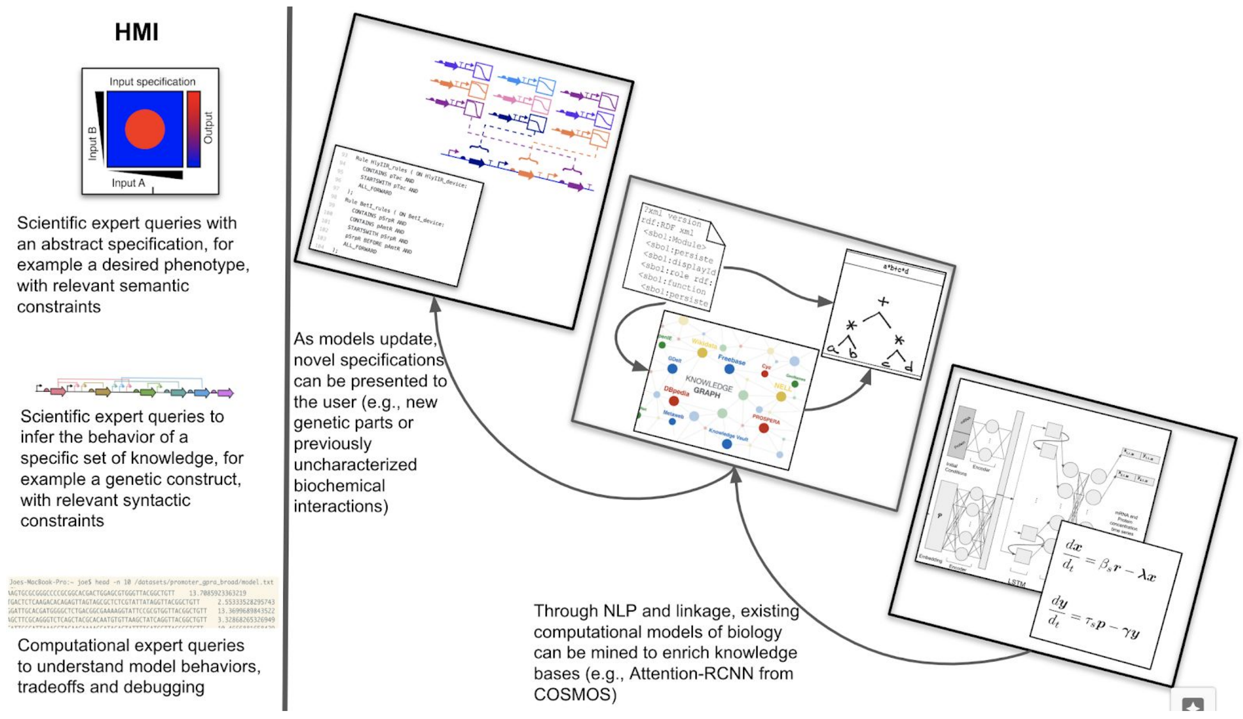
Asimov representation layers

Summary Figures

We present below two descriptions of Asimov's representation layers. The first details "top down" approaches to knowledge inference, conditional forecasting and counterfactual analysis:


HMI
Input specification
Input B
Input A
Output
Scientific expert queries with an abstract specification, for example a desired phenotype, with relevant semantic constraints
Scientific expert queries to infer the behavior of a specific set of knowledge, for example a genetic construct, with relevant syntactic constraints
Computational expert queries to understand model behaviors, tradeoffs and debugging
Formulations: scientific primitives (parts) and grammars (Eugene)
Choice of modeling framework defines a mapping from an abstract definition (phenotype) to a modeling framework (SBOL)
Choice of implementation library defines a mapping from modeling framework (SBOL) to executable software (ODE, DNNs, etc.)
Models: SBOL links into broad knowledge bases, associated with computational primitives
KNOWLEDGE GRAPH
Executable Code: specific model implementation
$\frac{dx}{d_t} = \beta_s r - \lambda x$
$\frac{dy}{d_t} = \tau_s p - \gamma y$

The second representation augments this view with a hypothesis for how "bottom up" knowledge extraction and further inform models and query patterns:



Scientific expert queries with an abstract specification, for example a desired phenotype, with relevant semantic constraints

Scientific expert queries to infer the behavior of a specific set of knowledge, for example a genetic construct, with relevant syntactic constraints

```
Joes-MacBook-Pro:~ joe$ head -n 10 /datasets/promoter_gpra_broad/model.txt
AGTGCGCGGGCCCCGCGGCACGACTGGAGCGTGGGTTACGGCTGTT    13.7085923363219
TGACTCTCAAGACACAGAGTTAGTAGCGCTCTCGTATTATAGGTTACGGCTGTT    2.55333528295743
GATTGCACGATGGGGCTCTGACGGCGAAAAGGTATTCCGCGTGGTTACGGCTGTT   13.3699689843522
AGCTTCGCAGGGTCTCAGCTACGCACAATGTGTTAAGCTATCAGGTTACGGCTGTT   3.32868265326949
```

Computational expert queries to understand model behaviors, tradeoffs and debugging

```
Rule HlyIIR_rules ( ON HlyIIR_device:
    CONTAINS pTac AND
    STARTSWITH pTac AND
    ALL_FORWARD
);
Rule BetI_rules ( ON BetI_device:
    CONTAINS pSrpR AND
    CONTAINS pAmtR AND
    STARTSWITH pSrpR AND
    pSrpR BEFORE pAmtR AND
    ALL_FORWARD
);
```

```
<?xml version
rdf:RDF xml
 <sbol:Module>
  <sbol:persiste
  <sbol:displayId
  <sbol:role rdf:
  <sbol:function
  <sbol:persiste
```

a*b+c*d

As models update, novel specifications can be presented to the user (e.g., new genetic parts or previously uncharacterized biochemical interactions)

$$\frac{dx}{d_t} = \beta_s r - \lambda x$$

$$\frac{dy}{d_t} = \tau_s p - \gamma y$$

Through NLP and linkage, existing computational models of biology can be mined to enrich knowledge bases (e.g., Attention-RCNN from COSMOS)

Layer Descriptions and Narrative

Layer 1, Executable Code

This layer is defined as a representation that can be directly exercised for quantitative inference and prediction. In our project, this empasses both DeepMOBB -- a deep learning executable for quantitative inference of (potentially unknown) genetic programs and our Simulator -- ODE executables for precise evaluation of known genetic interactions. Our goal for this layer is machine-interpreted code (e.g., PyTorch DNN) that can directly answer questions asked by users for quantitative analysis of biological systems:

- What is the steady state mRNA concentration of a part in a construct ?

Layer 2, Models

This layer is defined as a representation (or combination of representations) that includes the information necessary to generate an executable model. This is the SBOL layer. Our ontological model, SBOL, expresses known biological entities and their associated biochemical interactions in a RDF-triple ontology. These models can be directly mapped to an underlying executable layer (DeepMOBB or ODE Simulations), linked into other knowledge bases (e.g., dbpedia, freebase) or compiled into an AST representing computation over a generalized model. This opens modes of collaboration and model sharing with teams using different model representations. Human queries on this layer can be used to understand, validate and expand the underlying executable model.Example queries include:

- Given a set of designs, which optimize production of a target?
- How does adding or removing a part affect expression behavior?

Layer 3, Formulations

This layer is defined as a representation of scientific model primitives and compositional grammars. Specific to genetic circuit construction, model primitives would encompass known genetic parts and compositions of these parts enforced through a genetic grammar (e.g., Eugene) and a target specification such as:

- A steady state, digital logic specification (user defines a target scalar for expression)
- A steady state, analog logic specification (user defines a manifold for expression)
- A dynamic, analogic specification (user defines a manifold plus additional time dimension)

Such an abstract layer empowers users to query to discover new knowledge and methods that a user may not think to specify directly in the knowledge layer. The goal of our last ASKE technical phase is to create a user interface ("visual modeling language") such that it is natural to specify points in a high-dimensional manifold which queries the knowledge and executable layers. It is through this layer that we will be able to perform novel knowledge inference. Example queries include:

- What is the best circuit to produce a desired phenotypic outcome?

Narrative

A Top-Down view of circuit design:

A biologist seeks to produce a therapeutic (e.g, an antibody) that neutralizes an emerging biological threat, such as Ebola. Global production of such an antibody requires bioreactor scale manufacturing, using a mammalian cell culture that has been specifically engineered to produce the therapeutic. The biologist interacts with an ASKE abstract knowledge interface, querying for a genetic circuit that codes for high antibody production. The abstract knowledge layer applies reasonable biochemical contracts to this query, passing along the information to a structured knowledge layer. The structured knowledge layer in turn applies a specific model of genetic circuit construction to this abstraction, resulting in a genetic circuit design. An iterative process begins wherein a circuit design is passed to the executable knowledge layer, simulated via one or more computational models (DeepMOBB and/or an ODE system), and passed back up for verification. Once a feasible design is found, the resultant circuit is passed back to the user to physically construct.

Other Questions

Ideas for working groups

It would be interesting to spend some time defining a standards group. This group would be responsible for defining standards across

(1) communication protocols between layers: should a Protobuf encoded data payload, with specific mandatory and optional fields be used to move up-and-down layers?
(2) Representation of knowledge: should knowledge be defined exclusively as a RDF-triple knowledge graph or as an AST defined in JSON
(3) Transforms between knowledge representations: should teams use different knowledge standards so long as all are Turing-complete languages

Summary of potential linkages

In future programs, our structured knowledge layer could be cross linked into other programs for more complex analysis. As one example, HMS has built a knowledge layer of biochemical reaction networks. With sufficient time, circuit design models could be mapped into a comparable knowledge space. Such effort would yield an inference engine that can generalize beyond circuit function prediction, into the realm of endogenous network interactions.

Demo Day

Our initial aim is for a demo of "moving down the stack": a biologist specifies a phenotype, that can be optimized into a genetic circuit which is predicted to have a phenotype matching the specification.