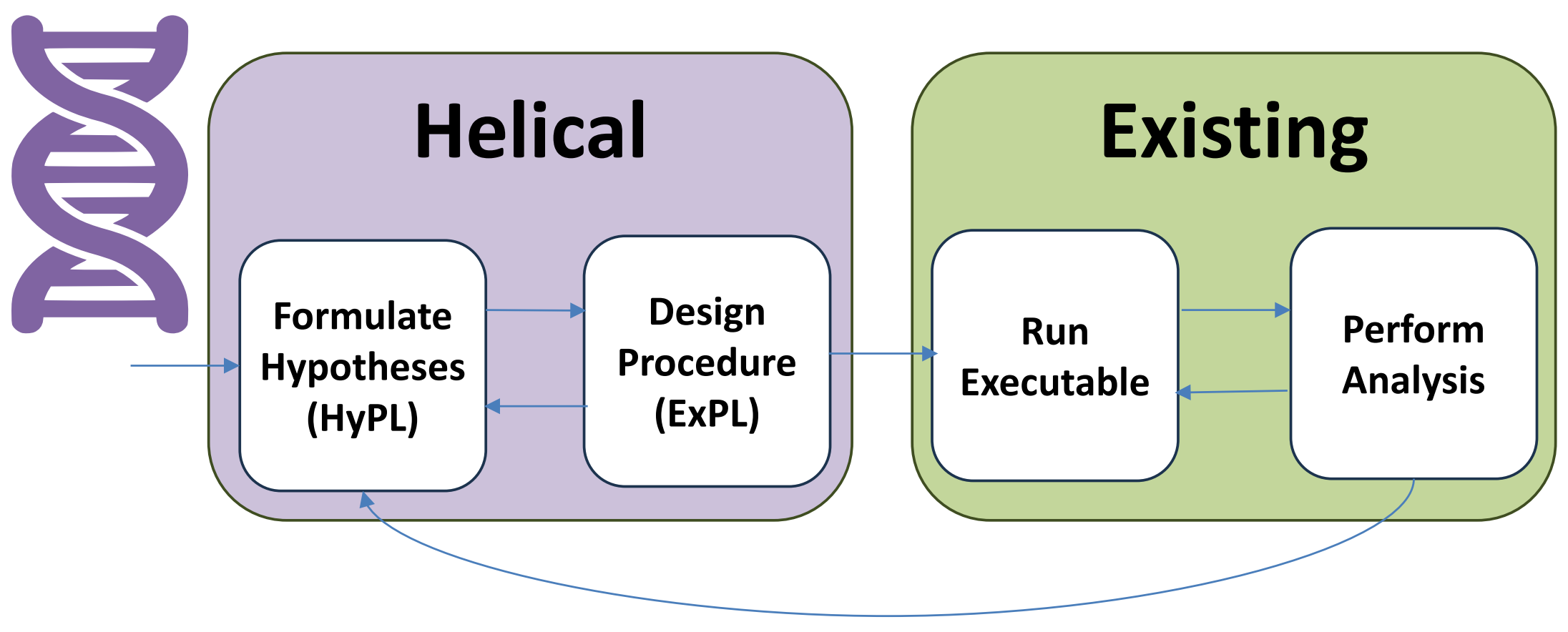
# Formal Methods in Software Support for Sound Experimentation



PIs: Emma Tosch and Chris Martens, Northeastern University

Am I asking the *correct* questions?

*Motivating Challenge*:

No automated enforcement nor validation of consistency between hypotheses, experiments, and analyses; undetected violations of internal validity can lead to issues with replication and reproducibility.

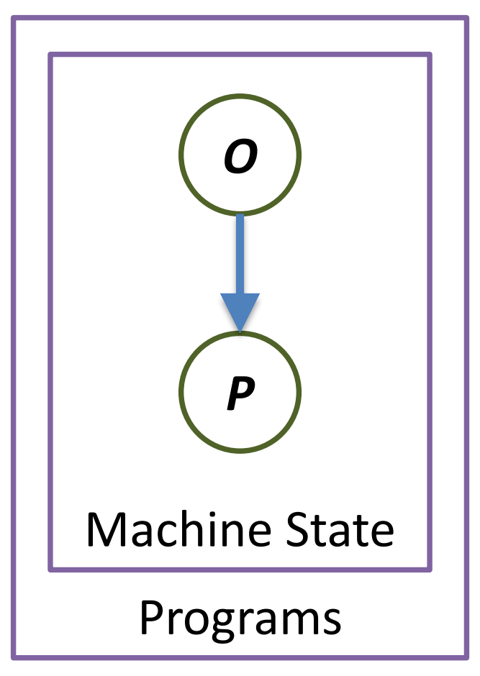
*Approach*:

- **Domain-specific languages** for encoding hypotheses and experiments.
- Enforce **consistency** via program analysis.
- **Integrate** with legacy tooling for both data collection and statistical analyses.

1. You have a hypothesis; ***HyPL*** helps: *Optimization level -O3 is better than -O2*

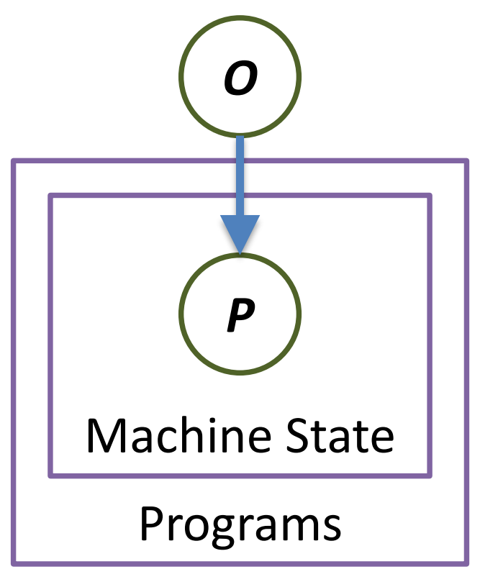
```
# HyPL encoding of prior work as described in
# Mytkowicz et al., ASPLOS 2009
O : { ‘-O2, '-O3 }
P : nat

(programs) P <- O
(programs) assert (P | O = '-O3) > (P | O = '-O2)
```



2. You create your experiment in ***ExPL***: Helical detects you're testing a specific ***refined*** hypothesis.

```
# Corresponding ExPL encoding
for prog in config.benchmarks @(samples prog      rams) do
 for optlevel in O do
  run prog.compile optlevel > @(intervenes O)
  for trialid in [1..config.ntrials] @(samples mstate) do
   run config.timing prog.exec > @(measures P)
  done
 done
done
```



I better rework my experiment; it can be better!

3. You can rework your hypothesis or collect data. This ex: Helical ensures the effect of ***Optimization*** on ***Performance*** is identifiable.

Now I have confidence in what I'm testing.

**Solution:**

- Specification language and tool support to tightly couple hypotheses and experiments.
- Static and dynamic analysis tools to automate checking that statistical analyses are consistent with hypotheses and data collection.

## Broader Scientific Impacts

- **Empirical Sciences:** Encoding past studies yields novel insights into sources of (in)validity.
- **Formal Methods:** Hypotheses as types for experiments.
- Aid in **replication**, **reproducibility**, and **auditing**, reducing overhead to validate findings.

## Broader Impacts

- FOSS with target user population beyond computing
- Jupyter notebook extension for interactive experimental design to support adoption
- Potential to identify scientific misinformation or invalid studies generated by malicious AI

## Grant Outputs

- Workshop keynote on artifact evaluation
- PostDoc mentorship at NEU
- Robotics & Software Engineering Seminar Talks
- NEU Coop Student funding and mentorship (Kevin Yang)
- Three UVM Graduate Students involved in early work