Formal Methods in Software Support for Sound Experimentation
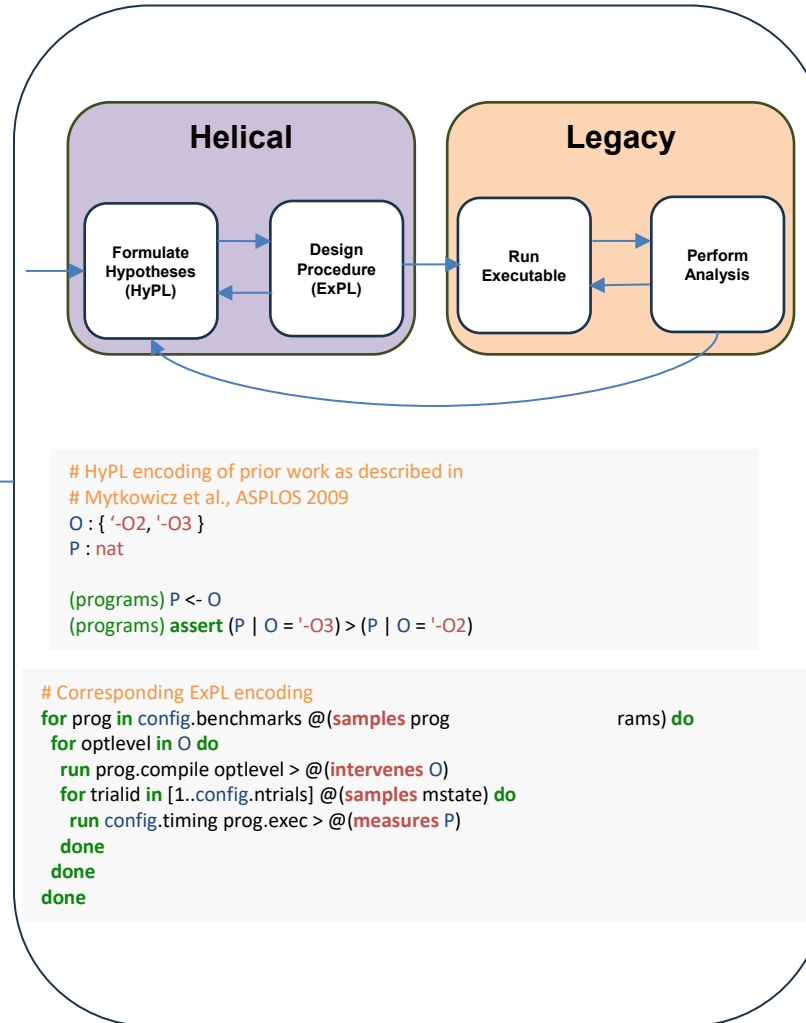
Challenge:

- Lack of automated enforcement of consistency between hypotheses, experiments, and analyses can violate internal validity and lead to issues with replication and reproducibility.

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.

Helical | **Legacy**

Formulate Hypotheses (HyPL) → Design Procedure (ExPL) → Run Executable ⇄ Perform Analysis

```
# 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)
```

```
# 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
```

Scientific Impact:

- Encoding past studies yields novel insights into sources of (in)validity.
- Treatment of data collection in empirical studies as a process that can be made "correct by construction"

Broader Impact and Broader Participation:

- Provide a common language and tooling for studying software-mediated phenomena.
- Aid in replication, reproducibility, and auditing, reducing overhead to validate findings.
- Active collaboration with and mentorship of fulltime undergraduate co-op student
- Workshop keynote on artifact evaluation