

A New “Ruleset” Application Testing Standard to Support Performance-Path Compliance and Above-Code Programs

Performance Ruleset Modeling Standard

Last 5 digits of project number | 75471

Principal Investigator (PI) | Supriya Goel

Lead Organization | Pacific Northwest National Lab

PI Email | supriya.goel@pnnl.gov

Performance-Path Compliance: Critical for Achieving Net Zero Goals

Performance Rating Methods (PRMs): methodology for code and above-code performance calculations

- Use whole-building energy simulation (BEM) to compare proposed building to a reference baseline to assess compliance
- Baseline transformation defined by “ruleset”
- Many examples: ASHRAE 90.1 Appendix G, CA Title 24 ACM, RESNET ERI
- + Supports both code and above code applications
- + Provides additional design flexibility, credits high-performance designs

Only used <5% of projects in >50% of jurisdictions surveyed!!

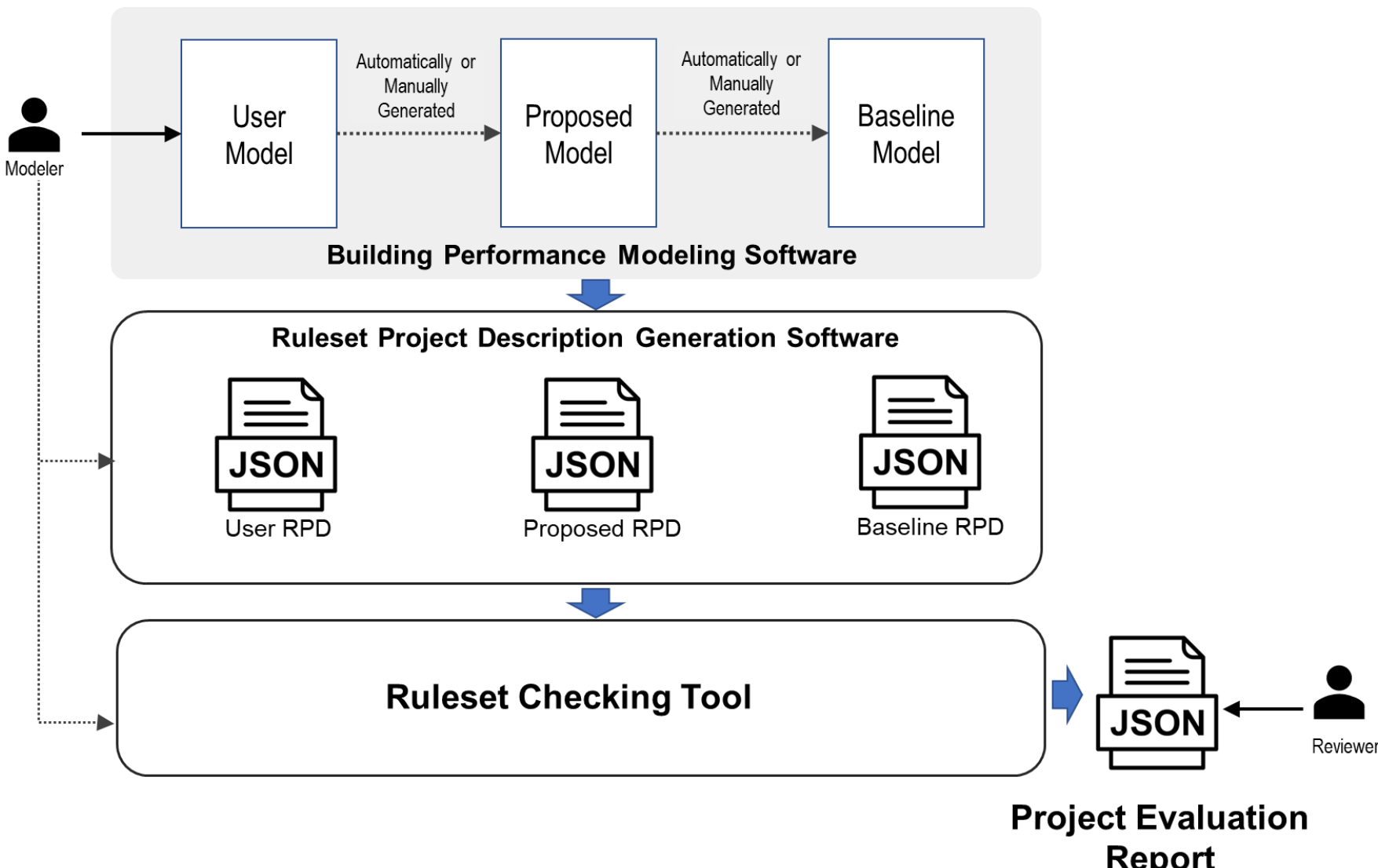
- Time and cost associated with generating energy model
- Ruleset automation inconsistent, users can manually “game” baselines
- Submission review is tedious, challenging, inconsistent, and unpredictable
- Review process requires multiple iterations, reducing project throughput

Code & program reviewers want confidence that models comply with rulesets!

Streamlining and Automating Performance-Path Submission Review

One Approach: Software Certification

- Test software using N representative cases
- Used by California, Florida, RESNET
- + Removes maximum burden from reviewers
- Requires interoperability that doesn’t exist
- Doesn’t handle manually generated baselines
- No way to flag ambiguous situations



New Approach: Project Checking

- Check ruleset application on a project basis
- + Doesn’t require large interoperable test suite, handles manually generated baselines and saves reviewing time
- + Flags ambiguous cases for manual reviews
- + Focuses manual review on “difficult” aspects
- + Can add coverage incrementally
- + Can help modeler create a clean submission
- Unfamiliarity with this concept, hasn’t been attempted before.
- Would still require some manual review

ASHRAE 229P “Protocols for Evaluating Ruleset Application in Building Performance Models”

Two new software components:

Ruleset Evaluation Schema (RES)

- Schema defined at ruleset level of detail
- Not a detailed BEM-to-BEM schema
- Not ruleset or simulation engine specific
- RPD exported by BPM vendor

Ruleset Checking Tool (RCT)

- Checks implementation of ruleset logic on Ruleset Project Description submittals
- Produces standard output report. Open-source

BEM software requirements

- Ruleset Project Description (RPD) export

Ruleset Checking software requirements

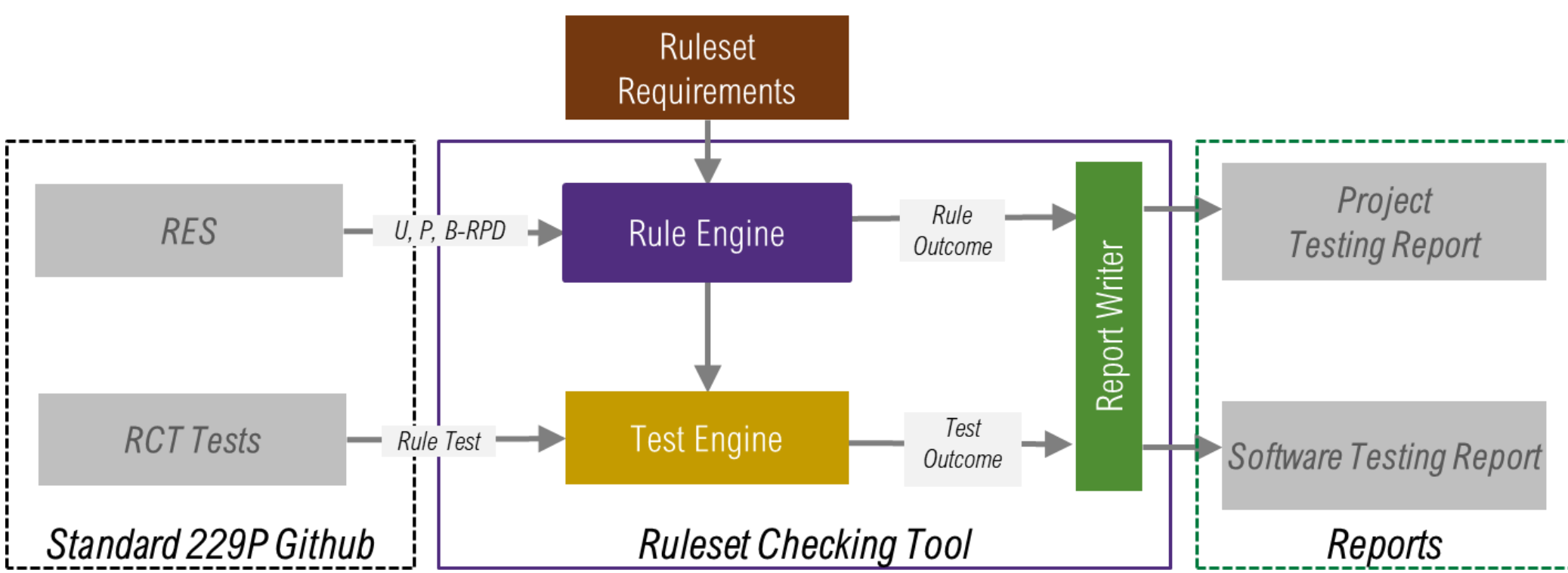
- Standard output report
- Compliance with rule tests defined in Standard



Completed 1st
Public Review
in 2024!

Open-Source Ruleset Checking Tool (RCT) for ASHRAE 90.1 -2019 Appendix G

Not a normative part of the standard ... others free to create checkers for other rulesets



Ruleset Checking Tool

- **Ruleset Requirements:** Description of rules and requirements to be verified by the RCT (including data tables)
- **Rule Engine:** Validates RPD files against schema, evaluates ruleset application in RPD files
- **Test Engine:** Contains all rule tests defined in ASHRAE Standard for RCT compliance
- **Report Writer:** Generates software testing report and project testing report

```
“zones”: [
  {
    “id”: “0”,
    “spaces”: [
      {
        “id”: “1”,
        “lighting_space_type”: “GUEST_ROOM”,
        “floor_area”: 100,
        “interior_lighting”: [
          {
            “id”: “1”,
            “power_per_area”: 0.41,
            “daylighting_control_type”: “NONE”,
            “lighting_multiplier_schedule”: “lm_sched_1”
          }
        ],
        “occupant_multiplier_schedule”: “om_sched_1”
      }
    ]
  },
  {
    “id”: “2”,
    “lighting_space_type”: “LOBBY_HOTEL”,
  }
]
```

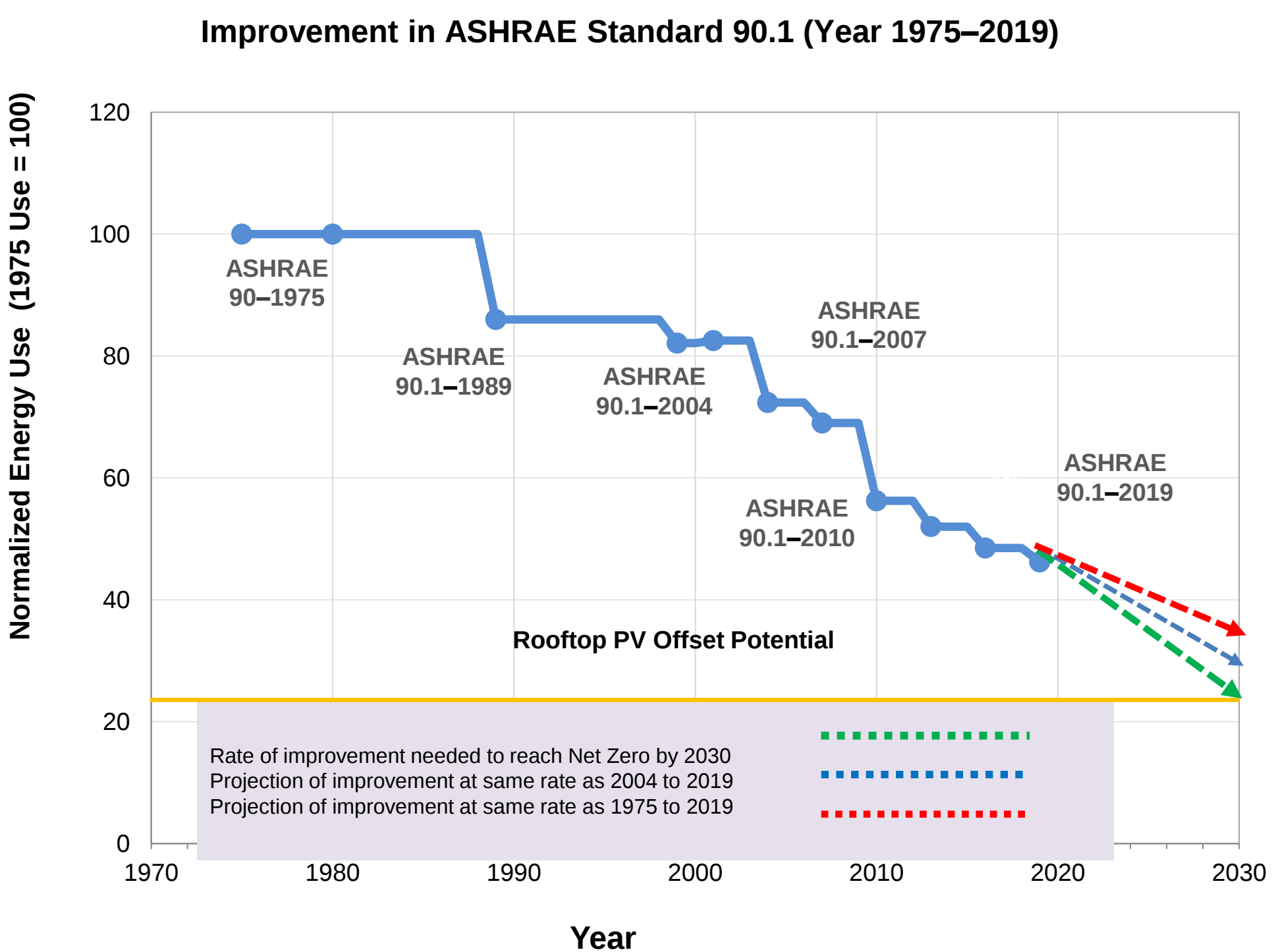
Baseline RPD Snippet

Rule Definition

```
def get_calc_vals(self, context, data=None):
    space_lighting_power_per_area = sum(
        find_all("interior_lighting[*].power_per_area", context.baseline)
    )
    lighting_space_type = context.user["lighting_space_type"]
    interior_lighting_power_allowance = table_G3_7(space_type=lighting_space_type)
    return {
        "space_lighting_power_per_area": space_lighting_power_per_area,
        "interior_lighting_power_allowance": interior_lighting_power_allowance,
    }
def rule_check(self, context, calc_vals, data=None):
    return (
        calc_vals["space_lighting_power_per_area"]
        == calc_vals["interior_lighting_power_allowance"]
    )
```

Standard 90.1 Rule:
Interior lighting power in the baseline building design shall be determined using the values in Table G3.7

Rule Test:
Baseline LPD for “Office-Enclosed” should be equal to 1.1 W/ft2



PROGRAM OFFICE LEAD | Amir Roth
Supported by Building Energy Modeling Program

PROJECT TEAM | Supriya Goel (PNNL), Michael Rosenberg (PNNL),
Weili Xu (PNNL), Jason Glazer (GARD Analytics), Maria Karpman (Karpman Consulting).