

Environment Scaling Engine

1 Repo Artifacts

 Repositories

 Issues

 Pull Requests

 Commits

2 Task Mining

Code Patch

Test Patch

 Problem Statement

 Requirements

 Interface Constraints

3 Clarity Check



 *ambiguous, incomplete, underspecified, inconsistent*

✓ Clarity Verified

4 AutoBuilder

 Build & Verify Agent

 Config Script

 Dependencies

 Clean Checkout

 Sandbox Execution

 *git exploitable traces*

5 Verification

 Parsed outputs

 >90% tests collected

 Reproducible P2P/F2P

6 Environments

 100K+ Environments

 12 Languages

 16.5% → 57.2% SUCCESS

Data Scaling Flywheel

1 Rollout Trajectories

 **Failed Attempts**

 **Near-Miss Attempts**

 **Passing Attempts**

2 Hint-Boosted

Process hints help near-misses become verified patches



Process Hints



success 0% → 20%

3 Hint-Free Replay

Regenerate clean trajectories from the original task context

 Original Context



✓ No hint leakage

✓ Verified patch

4 Process Filtering

Score and filter trajectories

 Exploration ☒

 Localization ☒

 Fidelity ☒

 Minimality ☒

 Verification ☒

 Honesty ☒

 Remove or down-weight low-quality process behaviors

5 Harness Robustness

Multiple Interfaces

 >>_  \$_  PS>  py ...

Randomize across

 func  arg  format  template

Perturbations

 deps  failures  truncated  noisy



 *Precise Tasks*

 *Executable Envs*

 *Validation Tests*

 *Verified Patches*

 *Robust Trajectories*

 *Reward Signals*

 *Harness-Invariant*