Interface Irreversibility Under Rule Externalization

YANLIN LI

December 18, 2025

Abstract

We formalize a decision interface and a verification interface as distinct computational contracts. We define a rule-externalization operation that moves rule-structure from the decision procedure into auxiliary information. We prove an interface-irreversibility statement: once rules are externalized, recovering a decision procedure without re-internalizing the same rule-structure is not well-defined as an interface transformation.

1 Motivation

Decision procedures presuppose that the criteria governing admissibility are fixed and explicitly available within the computational process itself. Given an input x, a decision procedure is expected to determine membership by evaluating x against an internally realized rule structure, producing an unconditional accept or reject outcome.

When such rule structure is externalized or left implicit, the computational task changes in character. Correctness can no longer be established by direct evaluation on x alone, but only relative to auxiliary information specifying how admissibility should be assessed. The task thus shifts from decision to verification: from determining membership outright to checking consistency with externally supplied conditions.

2 Two Interfaces

Definition 1 (Rule system)**.** *A rule system is a predicate $R(x)$ over inputs $x \in \Sigma^*$ determining admissible membership for a target language $L_R = \{x : R(x) = 1\}$.*

Definition 2 (Decision interface)**.** *A decision interface for R is an algorithm (or machine) D_R such that, for all x, $D_R(x) = R(x)$ and D_R is evaluated under a fixed internal rule realization of R.*

Definition 3 (Verification interface)**.** *A verification interface is a polynomial-time predicate $V(x, y)$ such that*

$$x \in L \iff \exists y\, V(x, y) = 1.$$

Here y is auxiliary information (a witness, certificate, or rule-encoding).

3 Externalization as an Interface Transformation

Definition 4 (Rule-externalization operator)**.** *Let E be an operator that transforms a rule system R into a relation $V_R(x, y)$ where y encodes missing rule structure needed to establish correctness.*

We say E is sound if for all x:

$$R(x) = 1 \iff \exists y\, V_R(x, y) = 1.$$

Remark 1. *E preserves the underlying language family in an extensional sense (membership), but changes the contract by which correctness is established: from direct decision to relative verification.*

4 Irreversibility

Definition 5 (Interface-preserving recovery map)**.** *A recovery map F is interface-preserving if it takes a verifier $V_R(x, y)$ and returns a decision procedure $\widehat{D}$ such that $\widehat{D}(x) = R(x)$ for all x,* without *access to any particular witness y at runtime.*

Remark 2 (Observation: Interface irreversibility under rule externalization)**.** *Decision and verification constitute distinct computational interfaces. A decision interface evaluates membership as a total function $x \mapsto R(x)$ under an internally fixed rule structure. A verification interface evaluates a relation $(x, y) \mapsto V(x, y)$ relative to auxiliary information y.*
When rule structure is externalized into y, the verification interface no longer determines a unique decision function on inputs x alone. Any attempt to recover a decision procedure from such a verifier must either (i) re-internalize rule-structure equivalent to that encoded in y, or (ii) leave the resulting decision behavior underdetermined.
This reflects an interface-level irreversibility: the transition from decision to verification preserves extensional membership but discards the functional contract required for decision. The loss cannot be reversed by interface manipulation alone.

Proof sketch. Decision is a functional contract $x \mapsto R(x)$; verification is a relational contract $(x, y) \mapsto V_R(x, y)$ with existential quantification over y. If y encodes rule-structure not determined by x, then multiple distinct rule-realizations can induce the same verification behavior on some pairs, while differing on the induced function $R(x)$.
Any F that outputs $\widehat{D}(x)$ from the verifier alone must, for each x, resolve which rule-realization the existential quantifier is referencing. But resolving this requires selecting (implicitly or explicitly) the missing rule-structure—precisely the information externalized into y.
Therefore, either (i) F is not well-defined as an interface transformation (it cannot uniquely determine $\widehat{D}$ from the verification contract), or (ii) F reconstructs and embeds an equivalent rule system internally, collapsing back to a decision interface by re-internalization. □

5 Implications and Scope

This note does not address the resolution of the classical P versus NP question. No claims are made regarding the relative power of deterministic and nondeterministic computation under standard formulations.
The purpose of the discussion is limited to well-formedness considerations. Specifically, we examine how changes in the computational interface—such as the externalization of rule structure—alter the meaning of feasibility predicates. The observations concern interface shifts rather than complexity separations.