

Constructing Sufficient State Descriptions for Coupled Deterministic Workflows

Wanhong HUANG
huangwanhong@serendip.ngo

Abstract

A state description can support a component's local task while failing after that component is connected to another workflow. This paper examines the resulting sufficiency problem in deterministic discrete procedures. It reconstructs continuation equivalence, derives a sufficient condition for synchronous composition, and then characterizes the effective interface required by a fixed receiving description. Exported values can be grouped when they induce the same retained receiver update under every declared control. On a full product carrier, preserving this effective interface is necessary and sufficient for an otherwise adequate sender description to support the coupled update. A fictional intake-and-routing dossier establishes exact description sizes and shows how a new operation makes an auxiliary field relevant through a delayed continuation. Further analysis separates global sufficiency from the allocation of information among components, and separates construction of a finer state space from recovery of its actual occupancy from old records. Verified mathematical antecedents locate the approach without a claim of conceptual priority. The philosophical discussion distinguishes behavioral equivalence from identity, conditional information need from authority to obtain information, and formal consistency from empirical and institutional adequacy. Open questions concern interface revision, distributed retention, and evidence for actual workflow dependence.

Keywords: sufficient state descriptions; deterministic workflows; continuation equivalence; component coupling; effective interfaces; record recovery

Research Purpose and Scope

This note states the purpose of circulation, the status of the argument, and the role of research assistance. The paper presents current insights for criticism and wider discussion in public and scholarly settings. It makes no claim to conceptual priority. Similar concepts and arguments may already exist in other disciplines, languages, and intellectual traditions; further comparison may alter the terminology, scope, or contribution.

Definitions organize a bounded inquiry. Conditional formal results carry their stated assumptions, and illustrative cases support conceptual analysis. Empirical validation, wider applicability, and the adequacy of proposed normative premises remain open where the main text identifies those obligations. Relational analysis supplies a method of investigation while stronger ontological questions remain suspended.

AI assistance contributed to source discovery and pre-use verification, argumentative development, formal checking, organization, and drafting. The accompanying project records distinguish retained research insights, newly proposed analysis, and verified literature. The manuscript is prepared for the author's review; circulation and submission decisions remain with the author. Critical responses concerning antecedents, assumptions, counterexamples, and alternative formulations are welcome.

DRAFT

1 Introduction

This paper examines state-description sufficiency in coupled deterministic workflows. Its objective is to identify the assumptions under which a description that supports one component's task continues to support a larger procedure after information exchange. The argument develops a continuation-based baseline, a composition theorem, a sharper result for a fixed receiving task, and a fictional intake-and-routing dossier. It then examines the epistemological and institutional limits of those constructions.

A workflow can retain a compact state description because many differences among its underlying configurations make no difference to its declared outputs. A receipt service, for example, might report whether an acknowledgment has been issued while retaining additional internal fields. If future acknowledgment behavior is independent of those fields, the reporting task can use a smaller description. Connecting that service to a routing procedure changes the inquiry. A field that the receipt task could safely omit may determine the route selected by the receiving procedure. The earlier description can remain adequate for its original task while becoming inadequate for the connected task.

This change creates a problem of compositional sufficiency. Descriptions are often assessed one component at a time, whereas their eventual use involves dependencies among components. A local assessment can establish that an observation supports recursive updates under a declared input repertoire. That assessment leaves open whether a later connection reads information outside the retained description, introduces a new operation, or makes a previously irrelevant distinction consequential. The paper supplies a way to locate these possibilities precisely.

The central position is conditional. Local descriptions compose when the information used by the coupling and the effects of the permitted local inputs are determined by the retained descriptions. A convenient sufficient condition requires each component to retain its exported interface. A more economical condition becomes available when the receiving task is fixed: exported values may be grouped whenever they induce the same update on the receiver's retained state for every declared external control. The resulting effective interface can be smaller than the raw exported field. Its adequacy still depends on the receiver, the allowed controls, and the sender's future dynamics.

The distinction between a raw field and an effective interface matters conceptually. A mathematical failure of an old description can identify information that a proposed connection would require. Several responses remain available. A designer can refine the retained description, change the receiving procedure, restrict the connection, or reconsider the larger task. The formal problem alone supplies insufficient grounds for choosing among those responses. In an institutional setting, the legitimacy of the task and the authority to obtain or disclose information require separate arguments.

The paper uses a relational method without resolving stronger ontological questions. A represented state is an equivalence class under a declared comparison. That construction specifies a role within a model. It leaves open the identity of a person, the individuality of an event, the historical formation of a relation, and the practical significance of differences excluded from the comparison. Two configurations can have the same workflow behavior while carrying different histories. A person can remain identifiable while the person's behavior changes. These distinctions constrain the interpretation of the mathematics throughout the paper.

The contribution builds on familiar mathematical ideas. Distinguishing continuations, behavioral descriptions, component interfaces, and information states already have developed antecedents. The paper's purpose is to connect selected results to a bounded representational problem and to make the associated assumptions inspectable. It makes no claim that state minimization, quotient construction, or compositional reasoning originates here. The proposed combination and its philosophical use remain open to comparison with more developed treatments in other disciplines.

The scope is deliberately explicit. A workflow is represented by discrete configurations and deterministic transitions under declared inputs. This representation can describe an idealized procedure or a fully specified formal system. Its application to actual institutional activity would require evidence that the relevant transitions, interruptions, interpretations, and discretionary choices have been represented adequately. The deterministic assumption concerns the model used in the argument. Its extension to

social life remains a distinct empirical and philosophical question.

The formal development proceeds through several layers. First, the paper distinguishes an observed snapshot from a recursively sufficient description. Second, it reconstructs the continuation construction that yields a coarsest sufficient description for fixed observations and inputs. Third, it shows how coupling can invalidate a locally sufficient description. Fourth, it proves conditions for composition and derives the effective-interface refinement. The worked dossier then establishes exact state counts and a delayed-relevance counterexample. The final sections examine record revision, uncertainty, institutional authority, objections, and open research obligations.

This sequence keeps the mathematical and philosophical claims connected. The proofs establish consequences of specified representations. The examples explain what those consequences permit an analyst to infer. The objections identify changes of context that require a fresh argument. The normative discussion addresses the legitimacy of the representational task itself. A useful state description should therefore be accompanied by an account of its purpose, its operating conditions, and the questions that its construction leaves open.

2 Antecedents in State Abstraction and Component Interaction

This section locates the argument within verified mathematical and methodological antecedents. Its objective is to identify the inherited ideas that support the analysis while preserving the difference between those ideas and their present application. The discussion considers distinguishing continuations, observation-based descriptions, interface composition, and information states, then states the epistemological position adopted for the workflow examples.

The distinguishing-continuation argument provides a basic reason for retaining a difference between states. Stanford’s CS103 treatment of nonregular languages explains that two input histories requiring different responses to a common continuation must lead a deterministic automaton to different states. The slides use this observation in a Myhill–Nerode argument about finite-state recognition [4]. The present paper uses the same elementary separation idea for workflow outputs. A future operation that produces different declared observations witnesses a difference that a sufficient description must preserve. The workflow construction allows a general output set and begins from an explicit configuration carrier; the cited teaching material concerns language recognition.

Jacobs and Rutten’s tutorial supplies a broader behavioral setting. Its opening account considers systems with hidden state that users access through specified observations and operations, and places such systems within coalgebraic investigation. The tutorial identifies coinduction and bisimulation among the associated methods [2]. This antecedent supports the use of observations and transitions in defining a state description. The present argument remains at an elementary set-theoretic level. It derives its quotient claims directly and leaves a more general coalgebraic treatment for later comparison.

Component interaction has an equally important antecedent. De Alfaro and Henzinger’s interface automata represent assumptions about inputs and guarantees concerning outputs, including their temporal order. Their optimistic composition treats components as compatible when some environment can make them work together [1]. The quantification differs from the theorem developed here. The present sufficient condition covers every external control in a declared class under a fixed synchronous coupling. The comparison matters because an interface claim always depends on the environments and interactions it permits. Compatibility in one such class leaves other classes open.

LaValle distinguishes a system’s actual state from an information state available to a planner. In the nondeterministic setting discussed in Chapter 11, an information state can retain possible current states and update them through transitions and observations [3]. This distinction becomes useful when an institution holds an old coarse record and later adopts a finer description. The model may identify which refined states are compatible with the old record while leaving their actual occupancy uncertain. A set of possibilities represents that uncertainty; selecting one member requires additional grounds.

These antecedents perform different jobs. The continuation argument concerns distinctions required by future observations. Behavioral abstraction concerns the relation between hidden configurations, ob-

servations, and operations. Interface reasoning concerns the conditions of interaction. Information-state reasoning concerns what a retained record warrants about actual occupancy. The present paper connects these jobs within a single workflow problem. Their connection requires explicit assumptions because a result about one job need not settle the others.

The epistemological position is consequently task-relative and fallible. Task relativity means that sufficiency is assessed for specified outputs and operations. Fallibility means that an empirical model may omit an operation, misdescribe an observation, or assume a relation that actual practice fails to sustain. These features are compatible with exact mathematical results. A theorem can hold exactly within a representation while the adequacy of that representation remains subject to evidence and criticism.

Task relativity also permits disagreement about the task. A reporting body may seek a compact description of completed transactions, while participants care about waiting, interruption, or the reasons for a refusal. The first description can be sufficient for its declared output and still leave the participants' questions unanswered. The disagreement then concerns the observation regime and its institutional justification. Describing the first regime precisely helps identify the disagreement without deciding it in advance.

The use of a carrier set introduces a further boundary. The carrier represents configurations available to the model. Its elements need not be interpreted as ultimate constituents of social reality. Likewise, an equivalence relation supplies a rule for grouping represented configurations, and a quotient supplies the resulting descriptive classes. Neither construction alone settles metaphysical identity. The analysis suspends that issue while examining the mathematical consequences of a declared grouping.

The verified antecedents establish a foundation for a modest contribution. The paper reconstructs standard state-description reasoning, develops its consequences for a fixed class of workflow connections, and examines the institutional choices concealed by an unqualified appeal to sufficiency. Wider literature comparison may reveal closer precedents or better terminology. Such comparison could refine the contribution while leaving the worked distinctions useful for ongoing discussion.

3 Description Regimes and Workflow Boundaries

This section defines the representational objects used in the derivations. Its objective is to distinguish the configuration being modeled, the observation selected for a task, and the retained description used for recursive updates. The discussion first states the domains and assumptions, then explains the interpretive boundaries that determine the later composition problem.

Let X be a nonempty configuration set, U a nonempty input set, and Y an observation set. For each input $u \in U$, let $T_u : X \rightarrow X$ be a total deterministic transition. Let $o : X \rightarrow Y$ be the declared observation. A finite input word is an element of U^* , and the empty word ϵ performs zero transitions. Words are read in execution order: applying u and then v produces $T_v(T_u(x))$. These conventions fix the order used by every continuation argument below.

Definition 3.1 (Description regime). A description regime is the tuple $(X, U, (T_u)_{u \in U}, o)$ together with an interpretation of its configurations, inputs, and observations. A change of the carrier, transition repertoire, observation, or interpretation requires a stated comparison with the earlier regime.

The tuple separates several decisions that an informal description can collapse. The carrier determines which configurations enter the comparison. The input set determines which operations may test or change those configurations. The transitions determine their modeled consequences. The observation identifies what the current task retains. An inquiry into admission counts and an inquiry into the reasons for admission can therefore use different regimes even when they begin from the same underlying records.

Let $q : X \rightarrow Q$ be a surjective description map. The value $q(x)$ is the retained description of configuration x . Surjectivity removes unused labels from Q and permits proofs to speak directly about the descriptions that actually have representatives. Equality of descriptions induces an equivalence relation on X : two configurations are grouped when their q values agree.

Definition 3.2 (Recursive sufficiency). A description q is recursively sufficient for a description regime when there is an observation map $\bar{o} : Q \rightarrow Y$ and, for every $u \in U$, a transition $\bar{T}_u : Q \rightarrow Q$ such that $o = \bar{o} \circ q$ and $q \circ T_u = \bar{T}_u \circ q$.

The first condition permits recovery of the current observation from the retained description. The second permits the retained description to update after an input without reopening omitted configuration details. Together they permit recovery of every declared observation along a finite input sequence. The definition concerns exact determination. Approximate prediction, average accuracy, and probability distributions over possible transitions require additional structures and different adequacy criteria.

Claim 3.3 (Pairwise sufficiency test). A surjective description q is recursively sufficient exactly when equality $q(x) = q(x')$ implies both $o(x) = o(x')$ and $q(T_u(x)) = q(T_u(x'))$ for every declared input u .

Proof. If the factor maps exist, applying $\bar{o}$ or $\bar{T}_u$ to the common description gives the required equalities. Conversely, define $\bar{o}(q(x))$ to be $o(x)$ and define $\bar{T}_u(q(x))$ to be $q(T_u(x))$. Surjectivity supplies a representative for each description. The pairwise conditions ensure that choosing a different representative leaves the assigned values unchanged. The resulting functions therefore satisfy the required factorization identities. $\square$

The pairwise test identifies a common failure of a snapshot description. Two configurations may share their current output while producing different retained outputs after a common input. The current observation then groups a distinction needed for recursive use. Adding a timestamp to the snapshot repairs this failure only if the timestamp actually determines the missing distinction under a justified model. The mere presence of more metadata supplies no general sufficiency guarantee.

The definition also separates a sufficient description from evidence of the current description. A model can establish that Q contains the states needed to run the procedure while an observer remains uncertain which element of Q currently applies. Conversely, an observer may know a detailed fact about a configuration that the declared recursive task never uses. Representational adequacy and epistemic access therefore vary independently.

The deterministic assumption makes the distinction particularly transparent. Once a configuration and input are fixed, the model assigns one successor. An analyst can nevertheless be uncertain about the configuration, the input that occurred, or the transition rule itself. Those forms of uncertainty should be represented separately. Replacing uncertainty about the rule with a single convenient transition would promote a modeling conjecture to a fact.

The workflow boundary is equally consequential. A local model can treat an incoming message as an input whose value is supplied externally. A coupled model can generate that same input from another component's state. The local transition table may remain unchanged while the dependence structure changes. This is the setting in which local sufficiency needs a composition argument: the connected procedure must be able to determine the relevant input from information retained somewhere in the product description.

Finally, input availability can be part of the task. The present core uses total transitions to keep the composition argument readable. A later section considers partial transitions and blocked operations. In an application, an operation's formal availability, practical availability, and institutional authorization should receive separate interpretations. A transition table can represent any of these under a declared convention, but its notation alone cannot decide which interpretation is warranted.

4 Continuation-Based Construction of Sufficient Descriptions

This section reconstructs the local state-description result needed for the composition analysis. Its objective is to derive a coarsest recursively sufficient grouping under fixed observations and inputs, then explain the finite refinement procedure. The proof makes observation retention, transition preservation, and greatestness separate steps so that later changes of regime can be assessed against the exact assumptions used here.

For each configuration x , consider its observation after every finite input word. Equality of those continuation observations defines the relation in Equation 4.1.

$$x \equiv_o x' \iff \forall w \in U^*, \quad o(T_w(x)) = o(T_w(x')). \quad (4.1)$$

Equation 4.1 includes the empty word, so equal continuation profiles entail equal current observations. It also includes arbitrarily long finite words. A difference that becomes visible only after several operations can therefore separate two configurations even when every shorter comparison agrees.

Proposition 4.1 (Coarsest sufficient grouping). *Continuation equivalence is the greatest equivalence relation, by inclusion of pairs, that preserves the current observation and is preserved by every declared transition. Its quotient map $q_o : X \rightarrow X/\equiv_o$ is recursively sufficient. Every recursively sufficient description separates any pair separated by q_o .*

Proof. First, equality of complete observation profiles is reflexive, symmetric, and transitive. Thus $\equiv_o$ is an equivalence relation. The empty-word case gives observation retention.

Second, suppose $x \equiv_o x'$ and fix an input u . For any continuation w , the observation after applying u and then w is the observation associated with the concatenated word uw . Equation 4.1 gives equality of those observations from x and x' . Consequently, $T_u(x) \equiv_o T_u(x')$. Every declared transition preserves the relation.

Third, let R be any equivalence relation that retains the current observation and is preserved by every T_u . If xRx' , induction on word length shows $T_w(x)RT_w(x')$ for every finite w . The base case is the original related pair. For the inductive step, apply preservation by the next transition to the related successors. Observation retention then yields equality of the final observations. Hence $x \equiv_o x'$, proving $R \subseteq \equiv_o$.

Fourth, use the pairwise sufficiency test to define the quotient observation and transitions. The preceding steps ensure that both are independent of the representative chosen. This establishes recursive sufficiency.

Finally, the kernel of any recursively sufficient description is an equivalence relation retaining observations and preserved by transitions. The greatestness result places that kernel inside $\equiv_o$. Therefore, a pair separated by continuation equivalence cannot be merged by a recursively sufficient description. Additional retained distinctions remain possible. $\square$

The direction of inclusion deserves attention. A greater equivalence relation contains more related pairs and produces a coarser partition. The proposition identifies the largest grouping compatible with the declared task. It supplies a minimal description in that precise sense. It does not identify a privileged description for every inquiry or establish that configurations within one class are identical in every relevant respect.

On a finite carrier, the relation can be constructed by repeated refinement. Let R_0 group configurations with equal current observations. Equation 4.2 defines the next relation from the previous one.

$$xR_{h+1}x' \iff o(x) = o(x') \wedge \forall u \in U, \quad T_u(x)R_hT_u(x'). \quad (4.2)$$

Equation 4.2 tests the current observation and all one-step successors using the distinctions already available at horizon h . Induction shows that R_h means agreement after every word of length at most h . The base case concerns the empty word. At the next stage, every nonempty word has a first input followed by a word of the shorter horizon, giving the recursive clause.

Proposition 4.2 (Finite stabilization). *If X has n elements and the initial observation partition has r classes, the refinement sequence reaches a fixed relation after at most $n - r$ strict refinements. That fixed relation is continuation equivalence. Finite enumeration of the input tests additionally requires a finite input set or a justified finite reduction of those tests.*

Proof. Each stage refines the preceding partition because agreement for a longer horizon entails agreement for every shorter horizon. A strict refinement increases the number of classes by at least one. The number of classes begins at r and cannot exceed n , so at most $n - r$ strict refinements can occur.

If two consecutive relations are equal, the fixed update rule in Equation 4.2 produces the same relation again at every later stage. The fixed relation therefore represents agreement for every finite horizon, which is precisely Equation 4.1. An implementation may perform an additional comparison to certify that equality has been reached. $\square$

The construction can incorporate several observation tasks. If an inquiry retains both o_1 and o_2 , use the paired observation (o_1, o_2) . Equality of all paired continuation observations is equivalent to agreement for each task separately, so the resulting continuation relation is the intersection of the two individual relations. This consequence supplies the local repair later used for an exported interface. The new task is first stated, and the dynamics then determine which additional distinctions its continued use requires.

A change of input repertoire has a parallel effect. Adding an input introduces additional continuation tests and can refine the sufficient description. The old proof remains valid for the old repertoire. Extending its conclusion to the new input requires checking transition preservation again. A field may become consequential because a new operation reads or transforms it, even when its current value was previously irrelevant to every output.

The finite procedure is a construction within a specified model. Observing a finite collection of actual workflow traces does not by itself reveal the complete transition table used by the construction. Different unobserved transitions may remain compatible with the available traces. Model identification therefore remains an evidence problem, and the finite stabilization bound supplies no automatic certificate that an empirical workflow has been fully observed.

5 Local Sufficiency and Hidden Dependence under Coupling

This section establishes the specific failure that motivates a composition theorem. Its objective is to show how two individually adequate descriptions can fail to support a connected update. The discussion constructs a minimal hidden-field example, identifies the missing dependency, and distinguishes a change of coupling from an error in the original local proof.

Consider a component A with configurations $a = (r, m)$, where both coordinates are binary. Its local observation is $o_A(a) = r$. Its standalone inputs can change r in declared ways, while the update of r remains independent of m . The component may also change m without changing r . Under these assumptions, the description $q_A(a) = r$ is recursively sufficient for the local observation: every declared input determines the next r from the current r and the input.

Now consider a component B with a binary state b . Its observation and retained description are both b . The component accepts a binary input e and updates to that value. Its local description is sufficient because the current state and supplied input determine the next retained state directly. The local specification leaves the origin of e to the environment.

Connect the components by supplying $e = m$ from A whenever a routing command occurs. At that command, A keeps its configuration and B sets its state to the exported bit. The proposed product description is (r, b) . Two global configurations with the same r and b , but different m , therefore share the proposed description. Their successors after the same routing command have different values of b .

Example 5.1 (Hidden exported distinction). Take global configurations $x = ((1, 0), 0)$ and $x' = ((1, 1), 0)$. Their product descriptions are both $(1, 0)$. After the coupled routing command, their product descriptions are $(1, 0)$ and $(1, 1)$ respectively. The proposed product description consequently fails the transition-preservation condition.

The example identifies a dependency that the local descriptions leave open. Component B can update from its description when the input value is supplied. The proposed product description must also

determine that input value, or at least determine its effect on the retained successor. Because the exported bit was omitted from A 's description, the connected procedure cannot obtain a unique retained successor from the product description alone.

The example preserves the correctness of both original local assessments. Component A was assessed for its observation of r , and m did not affect that observation under any allowed standalone continuation. Component B was assessed with its incoming value supplied externally. Coupling introduced a relation between the components that the two separate tasks did not require the product description to represent. The failure concerns the proposed transfer of local results to a new task.

This distinction helps interpret changes in institutional use. A record designed to report the acknowledgment of an application can later be used to select a route through another procedure. The later use may require distinctions that the acknowledgment report legitimately omitted. An analyst should then state the added task and its dependence on the record. Describing the old record as simply inadequate would conceal the narrower task for which it was constructed and assessed.

The hidden-field example also limits a familiar repair. Retaining the whole raw configuration would restore a deterministic update in the stipulated model, but that repair may retain more information than the receiving task uses. Several exported values may have the same effect on the receiver. A later theorem therefore asks for the smallest interface distinction required by a fixed retained receiving state, rather than assuming that every raw field should become available to the connected procedure.

The word "smallest" needs a declared comparison. The relevant interface can be minimal for preserving a given receiver description while remaining larger than necessary for a different receiver description or a different global output. A receiver that keeps a detailed audit state can distinguish inputs that a receiver recording only completion would merge. The analysis must therefore specify the retained receiving task before claiming that an interface distinction is necessary.

Coupling can also alter the available operation repertoire. A field omitted by A may remain harmless under the first connection but become relevant after a new command copies it into the exported field. In that case, the immediate interface value is insufficient for the sender's recursive use. The sender must retain enough information to update the interface under future commands. The next two sections address these two obligations in order: determining the present effect of a connection and preserving that determination through continued operation.

6 A Sufficient Condition for Synchronous Composition

This section proves a componentwise condition for preserving recursive sufficiency under a declared coupling. Its objective is to expose the information and timing assumptions needed to transfer local descriptions to a global procedure. The construction defines the coupled update from pre-step exports, proves quotient descent in separate steps, and then examines the result's limits concerning minimality and operating context.

Consider N components indexed by i . Component i has a nonempty configuration set X_i , a local input set U_i , and a deterministic update $T_i : X_i \times U_i \rightarrow X_i$. It has a local observation $o_i : X_i \rightarrow Y_i$ and an exported value $e_i : X_i \rightarrow E_i$. The export can represent a message, a record field, or a signal used by the coupling. Its mathematical role is specified independently of whether any particular participant can inspect it.

Let V be a nonempty set of external controls. A selector $\kappa_i : V \times \prod_j E_j \rightarrow U_i$ determines component i 's input from the external control and the tuple of current exports. The global carrier is initially the full product $X = \prod_i X_i$. Equation 6.1 defines one synchronous step.

$$F_v(x)_i = T_i(x_i, \kappa_i(v, e_1(x_1), \dots, e_N(x_N))) . \quad (6.1)$$

Equation 6.1 evaluates every selector from the configuration before the step. The right-hand side therefore defines an ordinary function. Using exports produced later in the same step would require an order of evaluation or a separate existence and uniqueness argument for simultaneous constraints. The synchronous convention prevents that additional assumption from entering implicitly.

For each component, let $q_i : X_i \rightarrow Q_i$ be surjective. Form the product description $q(x) = (q_1(x_1), \dots, q_N(x_N))$. The proposed global observation may be the tuple of local observations, or any fixed function of the local observations and exports retained by the model. A task that directly reads an additional hidden configuration field lies outside this observation assumption and must be added explicitly.

Assumption 6.1 (Local information and update retention). For every component i , equality $q_i(x_i) = q_i(x'_i)$ entails equality of o_i and e_i , and entails $q_i(T_i(x_i, u)) = q_i(T_i(x'_i, u))$ for every $u \in U_i$. Every coupled input is selected only from the declared exports and external control.

Proposition 6.2 (Product-description sufficiency). *Under the stated local retention assumption and the synchronous coupling in Equation 6.1, the product description is recursively sufficient for the declared global observation and every external control in V .*

Proof. Take any two global configurations x and x' with $q(x) = q(x')$. Equality of the product descriptions gives $q_i(x_i) = q_i(x'_i)$ for every component.

First, local export retention gives $e_i(x_i) = e_i(x'_i)$ for every i . The two global configurations therefore supply the same tuple of exports to every selector.

Second, fix an external control v . Because each selector is a function only of v and that tuple, component i receives the same local input from x and x' . Denote the common input by u_i .

Third, local transition preservation applies to the equal descriptions and the common input. It gives $q_i(T_i(x_i, u_i)) = q_i(T_i(x'_i, u_i))$. Using Equation 6.1, this is equality of the i th retained coordinates of $F_v(x)$ and $F_v(x')$.

Fourth, the equality holds for every component, so $q(F_v(x)) = q(F_v(x'))$. The global update therefore preserves the kernel of the product description.

Fifth, local observation retention gives equal local observations. The exports also agree by the first step. Any declared global observation factoring through these values consequently agrees. The pairwise sufficiency test now supplies a well-defined global observation and update on the product description. $\square$

The proof also establishes a finite-trace consequence. Begin two modeled procedures from configurations with equal product descriptions and apply the same finite external-control word. Transition preservation gives equal descriptions after the first control, then after each subsequent control by induction. Observation retention gives equal declared outputs at every step. The conclusion concerns the same external-control sequence; different actual controls require an account of the process that selects them.

If a controller selects its next control from retained observations and its own retained memory, that controller can be included as another component. Its memory then becomes part of the global configuration and its update must satisfy the same kind of sufficiency condition. Leaving a history-sensitive controller outside the model while treating its choices as determined by the current product description would discard a dependency needed for recursive use.

The theorem is sufficient without claiming universal necessity. Preserving every raw export is a convenient way to ensure equal local inputs, but different input values can induce the same retained successor. The next section exploits that possibility. Likewise, preserving local behavior under every input in U_i can be stronger than the requirement imposed by a particular coupling, which may reach only part of that input repertoire. The larger input condition offers a reusable contract across a broader declared environment class; its cost is a potentially finer description.

The global observation assumption is indispensable. Suppose a reporting service preserves only whether a transaction is complete, while the global task asks for the exact time spent waiting. The product can update completion correctly and still fail to answer the added timing question. The observation should then include the required timing information, and the component descriptions should be reconstructed or reassessed for that enlarged task.

The theorem also leaves the question of global minimality open. A product of sufficient local descriptions can retain distinctions that a particular connected task never reveals. A restricted carrier can

introduce correlations that make one component's retained value recoverable from another. A fixed coupling can prevent inputs that were needed to distinguish local states in isolation. These possibilities permit further compression, but the resulting description has a different scope of reuse. A globally smaller representation may rely on correlations or restrictions that another connection would invalidate.

Finally, the proof is indifferent to the social acceptability of the exported information. Its first step states a mathematical dependence: equal retained descriptions determine equal exports. That dependence can be satisfied by a carefully limited message or by an excessive disclosure. Evaluating those alternatives requires a purpose, an account of the parties affected, and justified constraints on information use. The formal result makes the dependence visible while leaving that evaluation to the later argument.

7 Effective Interfaces for a Fixed Receiving Description

This section derives a more economical condition for a one-way connection. Its objective is to identify exactly which distinctions among exported values matter to a fixed retained receiver state. The argument defines equivalence through the receiver's update, proves a necessary and sufficient factorization condition on a full product carrier, and then applies continuation closure to the sender's future use of that interface.

Let component A have configurations $a \in X_A$, observation o_A , and update $T_A(a, v)$ under external control $v \in V$. Let its raw export be $e(a) \in E$. Component B has configurations $b \in X_B$ and update $T_B(b, e, v)$. Its fixed retained description is a surjection $q_B : X_B \rightarrow Q_B$, which preserves its observation and every update for a fixed pair (e, v) . Thus two receiver configurations with the same retained description have the same retained successor when given the same exported value and control.

For each export e , define its effect on a receiver description $\beta \in Q_B$ and control v by taking any b with $q_B(b) = \beta$ and evaluating $q_B(T_B(b, e, v))$. The receiver's local preservation assumption makes this independent of the representative b . It therefore defines a function of (β, v) . Two export values are effectively equivalent when these functions agree, as stated in Equation 7.1.

$$e \sim_B e' \iff \forall b \in X_B \forall v \in V, \quad q_B(T_B(b, e, v)) = q_B(T_B(b, e', v)). \quad (7.1)$$

Equation 7.1 groups exports by their effect on the fixed receiver description. It is an equivalence relation because it compares functions by pointwise equality. Let $\rho : E \rightarrow E/\sim_B$ be the quotient map. The effective export of sender configuration a is $\rho(e(a))$.

The definition is relative in several respects. A receiver that retains a more detailed state can distinguish more effects. Additional external controls can expose a difference between exports that were previously equivalent. A changed receiving transition can alter the equivalence even when the raw export set remains unchanged. The effective interface is therefore a property of the declared relation between an export and a receiving task, rather than an intrinsic classification of the exported field.

Assume $q_A : X_A \rightarrow Q_A$ already preserves the sender's local observation and every update $T_A(\cdot, v)$. The coupled transition takes (a, b) to $(T_A(a, v), T_B(b, e(a), v))$. The export is read before the sender's update, preserving the earlier synchronous convention. The global output is any fixed function of the sender and receiver observations, with no additional direct reading of the raw export.

Proposition 7.1 (Fixed-receiver interface criterion). *On the full product carrier $X_A \times X_B$, the product description (q_A, q_B) preserves every coupled update exactly when $q_A(a) = q_A(a')$ entails $\rho(e(a)) = \rho(e(a'))$. Under the stated observation assumptions, this is also the criterion for recursive sufficiency of that fixed product description.*

Proof. For sufficiency, take a, a', b, b' with equal sender descriptions and equal receiver descriptions. Fix a control v . Sender preservation gives equal retained sender successors.

The effective-export condition says that $e(a)$ and $e(a')$ have the same effect on every receiver configuration. In particular, $q_B(T_B(b, e(a), v)) = q_B(T_B(b, e(a'), v))$. Receiver preservation for the fixed input $e(a')$ then gives $q_B(T_B(b, e(a'), v)) = q_B(T_B(b', e(a'), v))$. Combining these equalities proves

equality of the retained receiver successors. The product update is therefore well defined. The assumed local observation retention supplies the global observation condition.

For necessity, suppose two sender configurations share their description but their effective exports differ. By Equation 7.1, there is a receiver configuration b and a control v for which the retained receiver successors differ. The full product carrier contains both (a, b) and (a', b) . These configurations share the same product description, yet their retained successors after v differ in the receiver coordinate. This contradicts preservation of the product description. Hence equal sender descriptions must determine equal effective exports. $\square$

The necessity proof explains the full-product assumption. It needs the same witness receiver configuration to be compatible with both sender configurations. On a restricted correlated carrier, that pair of global configurations may be unavailable. A smaller interface can then suffice for the restricted procedure. Such sufficiency depends on the correlation remaining valid; the theorem should be reapplied if the carrier or connection changes.

The criterion also separates an interface from its encoding. Any message map $m : E \rightarrow M$ that permits exact recovery of the retained receiver update must separate exports with different effects, provided all receiver descriptions and controls remain possible. If $m(e) = m(e')$ while Equation 7.1 fails, the same message, receiver state, and control would require two different retained successors. Conversely, the effective class itself is an adequate message because its definition makes those successors agree. This gives a precise minimality statement about distinctions, without prescribing a particular data format.

Retaining the present effective export is only part of the sender's recursive task. A future sender operation may change that export in a way that depends on an omitted field. To obtain a coarsest reusable sender description for the fixed receiving task, apply the continuation construction to the paired observation $(o_A, \rho \circ e)$ under all declared sender controls. The resulting quotient preserves the local output, the current effective export, and their future updates.

Claim 7.2 (Sender construction for a fixed receiver). For the stated one-way connection, continuation equivalence for the paired sender observation $(o_A, \rho \circ e)$ is the coarsest sender grouping that preserves the sender's local task, supports every declared sender update, and makes the fixed product description recursively sufficient on the full product carrier.

Proof. The continuation proposition supplies a recursively sufficient quotient for the paired observation. It therefore preserves the effective export and satisfies the sufficient direction of the interface criterion. Conversely, any sender description meeting the stated requirements must preserve the effective export by the necessary direction of that criterion, must preserve the sender observation, and must be preserved by the sender updates. Greatestness of continuation equivalence places its kernel inside the paired-observation continuation relation. Thus it cannot merge a pair separated by the constructed quotient. $\square$

This construction refines the apparent choice between retaining a coarse local record and disclosing an entire configuration. A fixed receiving task may require only an effective message, while the sender retains additional internal information needed to update that message. Internal retention and external disclosure have different mathematical roles. Their practical justification can also differ. The theorem identifies a conditional information requirement; a governance argument must still decide whether the task, retention, and disclosure should occur.

8 An Intake-and-Routing Workflow Dossier

This section applies the preceding results to a finite fictional procedure. Its objective is to make the state distinctions, interface compression, and delayed relevance calculable in a single example. The discussion specifies an intake component and a routing component, derives their exact description sizes, introduces a new operation, and then examines a separate correlated-carrier case concerning allocation of retained information.

Table 1: Description sizes and separating tests in the fictional intake-and-routing procedure

Task regime	Retained tion	descrip- tion	Classes	Adequacy or separating test
Intake observation alone	r		2	All initial intake controls update the acknowledgment flag independently of the omitted bits
Initial coupled task	(r, b)		4	A routing request separates equal descriptions with different m
Initial coupled task with effective interface	(r, m, b)		8	Current output separates r, b ; routing separates m
Coupled task after adding promotion	(r, m, n, b)		16	Promotion followed by routing separates the previously omitted n

Let the intake configuration be $a = (r, m, n) \in \{0, 1\}^3$. The coordinate r records whether an acknowledgment flag is set. The coordinate m is a routing bit. The coordinate n is an auxiliary bit that the first routing procedure never reads. These are stipulated data roles. The bits carry no claim about a person's worth, eligibility, or legally recognized status. The acknowledgment flag records one procedural condition and supplies no implicit permission or precondition for the other commands.

The intake observation is $o_A(a) = r$. Controls can set or clear r , set either value of m , set either value of n , leave the configuration unchanged, or request routing. In the initial procedure, each setter changes only its named coordinate. A routing request leaves the intake configuration unchanged. Every command is total on the declared carrier. Partial availability would require the additional conditions discussed later.

The receiver has state $b \in \{0, 1\}$, which it both retains and displays. The raw export is (m, n) . On a routing request, the receiver sets b to m ; on every other control, it retains b . The global observation is (r, b) . Because all values of the four bits are admitted, the full global carrier has sixteen configurations.

Table 1 records the descriptions and the tests that determine their adequacy. The table includes a failed four-class proposal, so a smaller class count should be read together with its adequacy column. The counts are exact consequences of this fictional transition system. They supply no estimate of the state complexity of an actual institution.

For the intake task alone, $q_A(a) = r$ is sufficient. Every command either sets r to a declared value or leaves it unchanged. The omitted bits never affect a later acknowledgment observation under the initial control repertoire. Two acknowledgment values must remain distinct because the current observation separates them. The two-class description is therefore coarsest for the standalone task.

For the coupled task, the four-class proposal (r, b) fails. Fix r and b and vary m . The current output agrees, but a routing request sets the receiver to different values. The failure is exactly the hidden-export distinction from the earlier example.

The raw export has four values, but its effective interface has only two classes. Exports $(m, 0)$ and $(m, 1)$ induce the same retained receiver update under every control. On a routing request, the common value m determines the successor; on other controls, the receiver holds its state. Exports with different m are distinguished by a routing request. Consequently, $\rho(m, n)$ can be represented by m alone.

The sender can therefore retain (r, m) , and the product can retain (r, m, b) . Every initial control updates these coordinates independently of n . This establishes sufficiency of the eight-class global description. To establish coarseness, take two different triples. If they differ in r or b , the current output separates them. If they agree on those coordinates and differ in m , a routing request separates their outputs. Every difference among the eight triples is thus witnessed by a finite continuation. Continuation equivalence has exactly these eight classes.

Now add a control called promotion that copies the auxiliary bit into the routing bit, setting m to n while retaining r and n . The receiver holds b during that control. This is a stipulated procedural change. Its name describes a data movement and carries no evaluative claim about the promoted value.

Example 8.1 (Delayed relevance of an auxiliary field). Take intake configurations $(1, 0, 0)$ and $(1, 0, 1)$ with receiver state 0. Their old retained global descriptions agree. Promotion leaves the current global output unchanged in both cases but produces different routing bits. A subsequent routing request produces receiver outputs 0 and 1. The two-control continuation therefore separates configurations that the old eight-class description merged.

The added control makes the old description fail its update condition before the final difference becomes visible. Immediately after promotion, the retained routing coordinate m differs, even though the global output (r, b) still agrees. The later routing request exposes that retained-state difference at the observation level. This separates a failure of recursive sufficiency from a failure already visible in the current snapshot.

After adding promotion, all sixteen configurations require separate continuation classes. Differences in r or b are visible immediately. If those coordinates agree while m differs, routing separates the pair. If r , m , and b agree while n differs, promotion followed by routing separates the pair. These cases exhaust distinct configurations. The identity description is sufficient because it retains the full stipulated state, and the separating tests show that every pair must remain distinct for the enlarged task.

The change illustrates a carefully bounded claim about latent relevance. The auxiliary bit existed in the original carrier, but the original task and operations gave it no route to the declared output. The new command created such a route. The result concerns a changed modeled dependence structure. It does not show that every omitted feature will eventually become relevant, or that institutions should retain every feature in anticipation of unspecified future uses. An unrestricted future-task demand would erase the scope conditions that make sufficiency informative.

The dossier also separates internal retention from transmitted information. Under the enlarged repertoire, the sender needs n to update its future routing bit after promotion. The receiver still needs only the current effective export m to execute a routing request. The sender's internal state can therefore be more detailed than the message sent across the interface. Any practical retention or disclosure policy would need an independent account of necessity, access, correction, and acceptable future use.

A separate miniature case shows a limit of product minimality. Let two components have binary configurations a and b , with identity dynamics, but restrict the admissible global carrier to $C = \{(0, 0), (1, 1)\}$. Let the global task observe a , which equals b throughout this carrier. One allocation retains a in the first component and a constant in the second. Another retains a constant in the first component and b in the second. Both determine the global task and support the identity dynamics.

Neither allocation is componentwise coarser than the other. A family coarser than both would retain a constant in each component, and would merge the two observable global configurations. Therefore, among allocations permitted to redistribute retention for this global task, there is no componentwise coarsest sufficient family. The induced sufficient partition of C is nevertheless the same two-class partition in both allocations. The ambiguity concerns where information is retained, rather than the existence of a coarsest global behavioral description.

This final case depends on the carrier correlation and permits redistribution of local reporting obligations. If a component has an independent obligation to report its own bit, replacing that component's description by a constant would violate the local task even though the global observation remains recoverable elsewhere. If a new operation breaks the equality of the bits, the carrier must expand or the operation must be restricted. The allocation result therefore supports a question about institutional coordination; it leaves the legitimacy and stability of any particular allocation open.

9 Observation Revision and Recovery from Earlier Records

This section examines the transition from an old sufficient description to a finer one. Its objective is to distinguish constructing a new state space from knowing which new state applies to an earlier record. The discussion derives an exact recovery condition, introduces a set-valued representation of unresolved possibilities, and explains the consequences for historical reassessment and ongoing operation.

Suppose the earlier procedure retained $q_{\text{old}} : X \rightarrow Q_{\text{old}}$ and a revised task requires $q_{\text{new}} : X \rightarrow Q_{\text{new}}$. Both maps are defined on the same carrier for this comparison. Exact record conversion would be a function $f : Q_{\text{old}} \rightarrow Q_{\text{new}}$ satisfying $q_{\text{new}} = f \circ q_{\text{old}}$. The requirement says that the earlier record alone determines the new description of the same configuration.

Proposition 9.1 (Exact record conversion). *An exact conversion from an old description to a new description exists precisely when the new description is constant on every fibre of the old description. Equivalently, configurations sharing an old record must also share their new description.*

Proof. If the conversion exists, two configurations with the same old record are sent through the same value of f , so their new descriptions agree. Conversely, if the new description is constant on each old-record fibre, assign to an old record the common new description of its representatives. Surjectivity supplies a representative, and fibre constancy makes the assignment independent of that choice. The resulting function gives the required conversion. $\square$

The result is elementary, but its interpretation prevents a consequential mistake. A modeler can define a finer partition of the carrier without possessing the evidence needed to place an old case in a particular new class. The existence of the new classes establishes a representational possibility. It leaves the occupancy of those classes unresolved when the old record merged the relevant alternatives.

In the initial coupled dossier, an old record (r, m, b) omits n . After promotion becomes an allowed control, the new sufficient description includes n . Two configurations with different n have the same old record and different new descriptions. Exact conversion from that record alone therefore fails. A conversion procedure that fills the missing bit with a default creates an administrative convention; it requires a separate argument if the assigned bit is to be treated as evidence about the earlier configuration.

The available uncertainty can be represented without selecting a default. For an old record z , define $\Gamma(z)$ to be the set of new descriptions of configurations whose old description equals z . In the dossier, that set contains the two alternatives obtained by setting n to 0 or 1. The set is exact relative to the common carrier and the two description maps. Its interpretation remains conditional on the earlier record being accurate and on the carrier containing the actual possibility.

Additional evidence can narrow the set. A trustworthy retained source field may determine n . A documented operation history may determine it under a sufficiently supported transition model. A new observation may reveal its current value while leaving its earlier value uncertain. These forms of evidence have different temporal targets. Knowing the present auxiliary bit after several changes can leave the bit at the time of the old record unresolved.

Ongoing operation can proceed with possible-state information if the task permits it. Advance each possible new state under the observed control, collect its possible successors, and then retain those consistent with the new observation. The resulting record expresses uncertainty about occupancy while using the revised state description. A decision rule that depends on all possible states agreeing differs from a rule that chooses one representative or assigns probabilities. Those choices require their own operational and, where relevant, normative justification.

A change of carrier complicates the comparison further. The old and new models may represent different units, permit different configurations, or reinterpret a field. The fibre proposition then applies only after an explicit alignment identifies which configurations or histories are being compared. Similar labels in two databases do not establish that alignment. A record migration should therefore preserve the meaning of fields, the scope of conversion rules, and unresolved cases in addition to retaining a technical identifier.

Historical reassessment can also require a target outside both workflow descriptions. A revised procedure may distinguish which route a case would take now, while a later inquiry asks why a participant took a particular route earlier. The current sufficient state need not retain the actual sequence of earlier interventions, reasons, or communications. Recovering that history requires a historical target and an evidence record adequate for it. Predictive sufficiency for future operation and sufficiency for historical explanation should remain separately stated.

The argument supports a measured approach to retention. Anticipating a specific authorized future task can justify examining whether present records would support it. An unspecified demand to answer every future question would require a much broader retention commitment and could conflict with other interests. The mathematical result identifies the consequences of information loss for a declared conversation. It leaves the choice of present and future record obligations to an accountable practical judgment.

10 Epistemological Interpretation and Institutional Authority

This section examines the philosophical significance of the formal results. Its objective is to distinguish the warrant for a state description from the authority to impose its task, classifications, and information demands. The discussion develops task-relative adequacy, the separation of behavior and identity, and a conditional argument for making institutional state descriptions answerable to affected participants.

Recursive sufficiency has a clear epistemological role. It states that a retained description determines selected observations under selected operations in a specified model. The statement can be established by proof when the model is explicit. Its empirical use additionally depends on whether the model captures the relevant configurations and transitions. A successful proof therefore identifies a conditional warrant, with the interpretation and evidence premises remaining visible.

The dependence on a task creates a form of legitimate plurality. A service may use one description for issuing acknowledgments, another for routing cases, and another for auditing delays. Each description can have a different coarsest sufficient partition. Their coexistence need not indicate inconsistency. Inconsistency arises when a description is used for a task whose required distinctions it fails to preserve, or when incompatible interpretations are assigned to a field without an explicit reconciliation.

Plurality also creates questions about coordination. A participant may encounter the consequences of several descriptions at once, even when each institution evaluates only its local task. The composition theorem shows one way in which local adequacy can fail at the level of connected operation. A broader practical assessment must also consider burdens, access, correction, and the meaning of the combined procedure. Those concerns can remain invisible to a model whose global output records only successful routing.

Behavioral equivalence should consequently be interpreted with care. Two files may be interchangeable for one update procedure while representing different people. Two interventions may produce the same observed route while differing in authorization or reasons. A person may acquire a new workflow description while remaining the same participant in an ongoing relation. The quotient construction compares represented configurations for a task; identifying persons, events, or continuing responsibilities requires further criteria.

The same caution applies to historical formation. An exported bit can summarize a classification, but the bit alone may leave open whether the classification arose from an agreed procedure, an error, a coercive demand, or a later correction. If that formation history matters to the inquiry, the observation and evidence regime must preserve an appropriate historical target. A current predictive equivalence can merge such histories because their future outputs coincide under the declared operations. That merger expresses the limits of the task rather than a conclusion about the histories' significance.

The institutional issue can be formulated through explicit premises. Suppose a procedure materially affects participants' opportunities or burdens. Suppose further that those participants have a justified interest in understanding and contesting the grounds of those effects. Under these premises, an institution's use of a state description should be answerable concerning the task it serves, the distinctions it retains, the information it exchanges, and the correction routes available when its premises fail. This is a proposed normative argument; its premises require defense in a particular institutional setting.

Claim 10.1 (Conditional answerability of a description regime). If affected participants have standing to contest the grounds of a procedure's material effects, mathematical sufficiency for an institution-selected output leaves open an obligation to justify that output, the permitted operations, and the associated information exchange to those participants.

The inferential steps are straightforward. Sufficiency is established relative to the selected output and operations. The participants' standing concerns the grounds of the resulting effects, including the selection of that task. A proof that assumes the selection cannot by itself discharge a demand to justify the selection. The institution therefore needs an additional argument connecting its task and information practices to reasons that address the participants' standing. The claim's force depends on those normative premises, rather than on a hidden moral interpretation of an equivalence relation.

This argument permits several institutional responses to a newly discovered dependence. The institution may establish a justified need for an effective message and specify a limited disclosure. It may preserve information locally while transmitting only the required effect class. It may replace a receiving task that demands an unjustified distinction. It may suspend a connection until errors and contestation routes are understood. The theorem supplies a map of representational dependencies that can inform this choice.

A distinction between need and authorization is central. Under a fixed task, the interface criterion can show that certain message classes must remain distinct for an exact retained update. The result establishes a conditional information need for that task. It leaves open whether the institution should perform the task, whether the information can properly be obtained, and who may decide those questions. Mathematical necessity inside a chosen design cannot serve as its own practical authorization.

The effective-interface result also makes room for institutional restraint. Different raw exports can have the same relevant effect on a receiver. Sending a detailed raw field may therefore exceed the information needed for the declared update. Even an effective message may remain sensitive or objectionable in a particular context, so mathematical compression supplies only one consideration. The practical assessment must address what the message reveals, what further uses it permits, and whether the underlying classification is justified.

Finally, the analysis treats category revision as an accountable intellectual activity. Discovering a missed distinction can improve a model. Discovering that a retained distinction serves an objectionable purpose can motivate a different model or procedure. Both forms of revision are intelligible within the framework because it separates mathematical adequacy from the selection and justification of a task. The relational perspective helps identify the dependencies under examination while stronger claims about the ultimate constitution of social reality remain suspended.

11 Objections and Extensions of the Coupling Model

This section tests the principal assumptions that limit the formal results. Its objective is to distinguish extensions that require additional state from objections that require changing the task or interpretation. The discussion considers partial availability, asynchronous timing, restricted carriers, empirical uncertainty, and the risk of turning descriptive convenience into institutional authority.

The first objection concerns total transitions. Institutional procedures frequently distinguish a permitted operation, a refused request, and a request that cannot practically be carried out. A total transition table can represent these cases only under an explicit convention. If every attempted command produces a recorded result, refusal may be represented by a designated output and successor. If an unavailable command has no represented successor, the model instead uses a partial transition.

For partial transitions, quotient descent requires agreement about definedness as well as agreement about retained successors. Two configurations grouped by a description must permit the same declared input, and, when the input is permitted, their successors must share their description. To adapt the composition proof, first use export retention to obtain the same selected local input. Then use definedness preservation to show that the connected step is available from both representatives or from neither. Finally, use successor preservation when the step is available. This gives the same representative-independence argument on the defined part of the coupled transition.

A totalized failure state is another possible convention. Add a designated failure observation and send a blocked input to a formal failure configuration. That construction can make availability observable to the quotient procedure. Its interpretation should remain explicit: a formal failure configuration

represents the selected account of a blocked attempt, rather than a claim that an actual institution enters one permanent physical condition after every refusal. If blocked and unattempted operations are observationally indistinguishable, the observation must represent that fact.

The second objection concerns synchrony. Equation 6.1 uses exports from before a step. An actual procedure may update one component, transmit its new value, and then update another. Such a sequence can have different behavior from the synchronous model. The proper repair is to represent the order of component updates as part of the input or scheduler state. A scheduler that remembers which component has already acted requires enough retained memory to determine the next permitted action.

Order can matter even when each local transition is deterministic. In the dossier, promotion followed by routing can differ from routing followed by promotion because the receiver reads the routing bit at the moment of its own update. The operation sequence must therefore be retained at the level needed by the task. Treating an unordered collection of commands as a complete description would require a separate proof that the relevant operations commute for the declared observations.

The third objection concerns hidden channels. The composition theorem assumes that selectors depend only on declared exports and the external control. A human operator may consult an additional record, infer a value from timing, or use a communication outside the modeled channel. An application would need to investigate such dependencies. In a fully specified formal model, they can be introduced as additional inputs or state variables. In empirical work, their omission remains a model-adequacy question that a quotient calculation cannot resolve by itself.

The fourth objection concerns restricted carriers. A particular workflow may admit only configurations satisfying an invariant relation among components. Restricting the carrier can remove counterexamples that exist on the full product and can support smaller descriptions. This is legitimate when the restriction has adequate grounds and remains closed under the allowed transitions. If an operation sends a permitted configuration outside the restricted carrier, either the carrier must expand or the operation's availability must change. Merely discarding the inconvenient successor would alter the modeled procedure without explanation.

The fifth objection concerns the receiver description fixed in the effective-interface theorem. A chosen q_B may preserve more distinctions than the global task ultimately needs. The theorem then characterizes the interface required to maintain that chosen receiver state, which can be stronger than the interface required for a coarser global output. Jointly redesigning sender and receiver descriptions is a separate optimization problem. The correlated-carrier example shows why the location of retained information can remain underdetermined even when a coarsest global behavioral partition exists.

The sixth objection concerns empirical determinism. A deterministic transition law can be an idealized specification rather than a description of actual practice. Staff interpretation, changing rules, unrecorded choices, and incomplete inputs may make observed behavior depart from the specification. One response is to enrich the input or state until the model becomes adequate for the intended use. Another is to use nondeterministic or probabilistic transitions. Each response changes the evidential burden and the appropriate sufficiency condition.

The formal approach also risks excessive detail. A model that adds every conceivable variable can become difficult to interpret, impossible to identify, and practically intrusive. The present construction offers a counterweight by asking which distinctions a declared task and its allowed continuations actually require. That discipline remains useful only while the task itself stays open to criticism. An institution could otherwise select an excessively expansive task and use its information requirements to rationalize excessive retention.

Finally, a model can be mathematically consistent while institutionally misleading. Its categories may exclude participants' experiences, its outputs may reward an undesirable objective, and its interface may distribute burdens unfairly. The answerability argument therefore remains separate from the composition proof. Formal analysis contributes by revealing dependencies and counterconditions. Its practical role depends on the wider process through which purposes, evidence, affected interests, and alternative designs are assessed.

12 Open Questions and Evaluation Obligations

This section identifies the work required to extend the analytic construction and assess its practical usefulness. Its objective is to distinguish mathematical open problems, empirical identification tasks, and institutional evaluation. The discussion proposes finite model checks, a workflow study design, and separate questions about changing receivers, distributed retention, and justified information use.

Open Problem 12.1 (Interface revision across receiving tasks). Develop a comparison procedure for effective interfaces when the receiving transition, retained state, or control repertoire changes. The procedure should identify which earlier distinctions remain sufficient, which require refinement, and which become dispensable under the revised task.

For a fixed carrier and transition law, adding receiver tests can refine the effective-export relation. Changing the law can produce a different relation without a simple refinement order. An old and a new interface can merge different pairs of raw values. A useful comparison should therefore report the relevant assumptions and separating cases, instead of relying on a single measure of interface size.

Open Problem 12.2 (Joint allocation of retained information). Characterize sufficient component-description families under explicit carrier correlations, local reporting obligations, and permitted couplings. Determine conditions under which a componentwise coarsest family exists, and describe the alternatives when adequate retention can be distributed in several incomparable ways.

The correlated binary example establishes the possibility of incomparable allocations under a minimal global task. A broader treatment would need to distinguish the partition induced on the global carrier from the locations at which its information is retained. It would also need to account for failures of access, component replacement, and changes in the correlations that permit compression. Those practical conditions can make one mathematically adequate allocation more robust than another.

Open Problem 12.3 (Empirical identification of effective dependence). Establish an evidence procedure for assessing whether an observed workflow's receiving behavior depends only on a proposed effective interface. The procedure should distinguish a proved property of a specification from an empirically supported claim about actual practice, and retain uncertainty about unobserved controls and hidden channels.

Finite verification can test the internal claims without supplying empirical validation. For small carriers, enumerate transition maps, export maps, and quotient descriptions. Compare the local effective-interface condition with an independent test that groups global configurations by their retained product value and checks whether each group has a unique retained successor under each control. Include both full-product and correlated-carrier cases so that the necessity assumption receives direct scrutiny.

The dossier admits an additional independent check. Explore pairs of configurations under common control words and record the first word producing different observations. Such pair exploration can establish whether a proposed merger has a separating continuation without using the partition-refinement implementation itself. The initial procedure should yield eight continuation classes, while the enlarged procedure should yield sixteen. A failed four-class description should have a one-step routing witness, and the newly relevant auxiliary bit should have a promotion-then-routing witness.

An empirical study would begin with a bounded procedure and an explicit observation contract. The study should document which outputs the task requires, which controls are available, what information each selector uses, and which states or traces remain uncertain. Model construction should involve both procedural documentation and examination of actual cases. Divergences between the specified procedure and observed practice should become evidence for model revision, with the source of each change recorded.

A useful comparison could evaluate three designs: an existing local reporting arrangement, a connection using raw exported fields, and a connection using a proposed effective interface. Evaluation should distinguish update correctness, unexplained exceptions, information transmitted, internal information retained, correction effort, and participants' ability to contest a consequential classification. A design that

reduces message size while making correction harder would require a more complex assessment than an efficiency score alone.

The study should also include changes of task. A new receiver control, an additional output requirement, or a revised source field can reveal the limits of an interface that performs well under the initial contract. Such tests should be justified within the study's scope. Introducing arbitrary future uses solely to defeat every compact description would undermine the task-relative comparison. The research question concerns robustness across a declared family of plausible changes.

Institutional evaluation requires a separate account of justified purposes and affected interests. A study can show that a message suffices for a receiving update while leaving the legitimacy of that update disputed. Participants may object to the underlying classification or to the use of a record for a newly connected purpose. An evaluation should preserve those objections as part of the assessment, rather than coding all resistance as a failure of implementation.

Further philosophical comparison may refine the account of state, observation, and institutional categories. The present paper treats them as working representational roles and suspends a stronger ontology. Alternative approaches may give greater emphasis to practice, interpretation, embodiment, or the history of classification. Their relevance should be assessed through the specific questions they help answer, with source positions verified before attribution. The current formal results can then function as limited analytic inputs within a wider discussion.

13 Conclusion

This section states the completed contribution and its remaining obligations. Its objective is to connect the local sufficiency construction, the coupling results, and their philosophical interpretation. The conclusion first identifies the conditional mathematical outcomes, then records the evidence and authority questions that remain open for institutional applications.

A sufficient local description retains the distinctions needed to recover selected observations and update those observations through declared operations. Continuation equivalence supplies a coarsest grouping for that fixed task. Connecting components introduces an additional demand: the retained product description must determine the effects of their interaction. The hidden-field example shows how that demand can fail even when the separate local descriptions satisfy their original specifications.

The composition theorem establishes one sufficient repair. When local descriptions preserve their observations, exported interfaces, and all declared local-input updates, selectors based on those exports produce a well-defined synchronous update on the product description. The proof identifies its assumptions directly, including the timing convention, external-control class, and global observation task.

For a fixed receiver, the effective-interface theorem gives a sharper condition. Raw export values may be merged when they produce the same retained receiver update under every declared receiver state and control. On the full product carrier, preservation of that effective export is exactly the extra condition needed by an otherwise adequate sender description. Applying continuation closure to the sender's local output and effective export then retains the information needed for future operation.

The finite dossier shows the practical meaning of those distinctions within a stipulated system. Its standalone intake task has two classes, its initial connected task requires eight global classes, and an added promotion operation makes all sixteen configurations distinguishable. The receiver can still use a smaller effective message than the sender's internal state. A separate correlated-carrier example shows that sufficient information can be allocated among components in several incomparable ways.

These results support precise questions about a proposed workflow connection. They identify which dependence a compact record omits, which interface distinctions a fixed receiver requires, and which evidence an old record fails to supply after refinement. They also locate the assumptions that changes of task, control, carrier, or interpretation can invalidate.

The philosophical conclusion remains conditional and task-relative. A quotient state supplies a representation for a declared inquiry. Empirical adequacy, historical identity, the legitimacy of classification, and authority to exchange information require additional grounds. The paper offers its constructions

for criticism and further comparison, with the aim of making those grounds easier to distinguish and discuss.

DRAFT

References

- [1] Luca de Alfaro and Thomas A. Henzinger. Interface automata. In *Proceedings of the 8th European Software Engineering Conference*, pages 109–120. ACM, 2001.
- [2] Bart Jacobs and Jan Rutten. A tutorial on (Co)Algebras and (Co)Induction. *EATCS Bulletin*, 62:222–259, 1997.
- [3] Steven M. LaValle. *Planning Algorithms*. Cambridge University Press, 2006. Chapter 11. Author-hosted text: <https://lavalle.pl/planning/ch11.pdf>.
- [4] Stanford University, CS103. Lecture 18: Nonregular languages. Winter 2025 course slides, 2025. <https://web.stanford.edu/class/archive/cs/cs103/cs103.1254/lectures/18/Condensed%20Lecture%20Slides.pdf>.

DRAFT