TECHNICAL NOTE 01 / VERSION 1.0

Equal Totals Do Not Identify Equal Records

The exact boundary between a reconciliation check and a completeness proof

Kylian de Groot
5 September 2026

ABSTRACT

A reconciliation can reproduce both the number of records and their total amount while containing the wrong records. This note gives an explicit counterexample, proves the additional inclusion condition under which a count is sufficient, and characterises exact recovery from linear measurements on a finite record universe. It also separates bounded checksums from exact identity comparison. The results sharpen the use of count-and-total checks in financial ingestion: a mismatch can reject a candidate, while a match requires a stated model before it can establish completeness.

Expository note: elementary results with explicit proofs. No claim of mathematical novelty or measured product performance.

1. The claim a reconciliation is allowed to make

A payment feed arrives with a source count and a source total. Reproducing these values is a useful control. The question is whether the control identifies the source records, or merely rules out some possible mistakes. These are different mathematical claims. This note studies a fixed, finite source snapshot. Amounts are integer cents, identifiers are scoped to that snapshot, and a record includes the payload whose correctness matters.

Let U be a finite universe of immutable records. Each record u has a fixed amount v(u) in the integers. Let E be the source set and L the locally reconstructed set. Set semantics mean that duplicate records have already been removed by a valid identity rule. Define the count and the amount sum as follows.

$$N(A) = |A|, \; S(A) = \sum_{u \in A} v(u).$$

The target assertion is L = E. Equality of the summaries (N(L), S(L)) and (N(E), S(E)) is a necessary consequence of that assertion. Necessity alone does not establish the converse.

2. A four-record counterexample

Record	a	b	c	d
Amount in cents	100	400	200	300

Proposition 1. Equality of count and amount sum does not, in general, imply equality of record sets, even when every amount is strictly positive.

Proof. Take E = {a, b} and L = {c, d}. Both sets have cardinality two. Their sums are 100 + 400 = 500 and 200 + 300 = 500. The sets are disjoint and therefore unequal. All four amounts are positive. Thus neither positivity nor a simultaneous count check excludes substitutions.

The witness is about exact arithmetic, not floating-point error. There is no tolerance to tighten. A source statement of two records and 500 cents is compatible with both candidates. A repeated summary cannot distinguish candidates that have exactly the same summary.

3. The missing assumption that makes a count sufficient

Proposition 2. For finite sets L and E, if L is a subset of E and |L| = |E|, then L = E.

> *Proof.* Because L is a subset of E, the sets L and E minus L form a disjoint partition of E. Hence |E| = |L| + |E minus L|. Equality of the first two cardinalities implies |E minus L| = 0. The difference is empty, so E is a subset of L as well. Both inclusions give equality.

The inclusion premise is substantive. It means that every accepted local record has been shown to belong to the same source snapshot with the same relevant payload. Matching only an identifier does not establish that its amount or version is correct. Counting repeated deliveries does not establish set cardinality. Reading a changing source through several pages does not automatically establish a single snapshot.

Proposition 3. If L is a subset of E and v(u) is strictly positive for every u in E, equality S(L) = S(E) also implies L = E.

> *Proof.* The difference of the sums is the sum of v(u) over E minus L. A nonempty finite sum of strictly positive integers is strictly positive. Therefore a zero difference forces E minus L to be empty.

Both qualifications matter. A missing zero-valued record defeats the conclusion if zero amounts are allowed. A missing pair with amounts 100 and −100 defeats it for signed amounts. Count equality under verified inclusion still works in both cases because it counts records rather than their net financial effect.

4. An exact criterion for linear summaries

Number the records in U from 1 to m. Represent a candidate set by an indicator vector x in {0,1}^m. Let A be any fixed rational matrix whose columns correspond to these records. Counts, totals and partition totals are all examples of the measurement vector Ax.

Proposition 4. The map x ↦ Ax is injective on {0,1}^m if and only if the only vector z in {−1,0,1}^m satisfying Az = 0 is z = 0.

> *Proof.* If two different indicators x and y give Ax = Ay, their difference z = x − y is a nonzero vector with entries in {−1,0,1} and Az = 0. Conversely, given such a z, set x_i = 1 where z_i = 1 and zero elsewhere; set y_i = 1 where z_i = −1 and zero elsewhere. Then x and y are different binary vectors, x − y = z, and Ax = Ay. These two constructions prove both directions.

For the four-record example, choose A with first row (1,1,1,1) and second row (100,400,200,300). The vector z = (1,1,−1,−1) is the explicit witness: its count measurement is zero and its amount measurement is 100 + 400 − 200 − 300 = 0.

This is not a claim that every compressed-looking summary must fail. On a known universe, the single exact integer sum of the weights 1,2,4,...,2^(m−1) identifies a binary vector uniquely. To prove this, take the highest position at which two vectors differ. Its weight exceeds the sum of all lower weights, so lower differences cannot cancel it. Such an encoding requires up to m bits; it is an inventory encoding, not an ordinary monetary total.

5. Checksums, identity and the earlier working paper

Proposition 5. If m > b, no function from all subsets of an m-record universe to b-bit values is injective.

> *Proof.* There are 2^m subsets and at most 2^b output values. When 2^m > 2^b, assigning every subset a different output is impossible: at least two inputs share an output. This is the finite pigeonhole principle.

A cryptographic digest may make accidental or adversarial collisions sufficiently unlikely for a particular system. That is a different statement, with computational or probabilistic assumptions. It must not be silently substituted for mathematical injectivity. Likewise, an exact record comparison proves agreement with the chosen source snapshot, not that the source itself contains every real-world transaction.

Clarification to the earlier working paper, Absence of Errors Is Not Evidence of Completeness: its section 5 describes a reproduced count and sum as a completeness proof. Read without a verified-inclusion premise, that statement is too strong. Proposition 1 supplies a counterexample; Proposition 2 supplies a sufficient missing premise. This note is an explicit clarification dated 5 September 2026. The earlier PDF is retained as a historical version.

6. Consequence for a financial ingestion gate

A gate can reject a candidate as soon as an independently obtained count or total differs. To accept exact completeness, it must additionally establish an appropriate identity condition: verified membership plus equal cardinality, direct equality of canonical record sets, or an encoding whose injectivity is proved on the actual candidate domain. Each condition applies to a fixed scope and snapshot. A correct hourly proof is not a proof about an unexamined month.

All examples here are constructed witnesses. No result depends on customer records, a production incident, or a measured error rate. The proofs are finite set arguments and exact integer equalities.

References and relation to earlier work

Kylian de Groot. Absence of Errors Is Not Evidence of Completeness. Working paper, 2026; section 5 is clarified here.

Companion verification. Download the finite checks (Python 3). Exact integer and rational checks supplement the proofs; they do not establish universal claims beyond the checked domains.