# **Pre-Coding Essentials (Component: crates/vm\_pipeline/src/tabulate.rs, Version/FormulaID: VM-ENGINE v0) — 52/89**

## **1) Goal & Success**

Goal: Implement **TABULATE** stage: per-Unit computation of **UnitScores** according to VM-VAR-001 (plurality, approval, score, ranked\_irv, ranked\_condorcet). Record audit artifacts (IRV round log, Condorcet pairwise matrix).

Success: Output matches Doc 5’s **UnitScores** contract and feeds **ALLOCATE** deterministically; denominators and rounding follow Doc 4; no network; stable ordering.

## **2) Scope**

In: LoadedContext’s per-Unit tallies + ParameterSet. Out: UnitScores (per Unit: scores, turnout, optional RoundLog/PairwiseMatrix). **Do not** apply allocation, thresholds, gates, or frontier here.

## **3) Inputs → Outputs (with schemas/IDs)**

Inputs: BallotTally (shape varies by ballot type), Options (order\_index fixed), Units, Params. IDs/ordering per Annex B Part 0.

Output: **UnitScores** per Unit:

scores{Option→natural tally}; turnout{ballots\_cast, invalid\_or\_blank, valid\_ballots}; audit: RoundLog (IRV) / PairwiseMatrix (Condorcet). Consumed by **ALLOCATE**.

## **4) Entities/Tables (minimal)**

## **5) Variables (used here)**

## **6) Functions (signatures only)**

rust

CopyEdit

pub struct UnitScores {

pub scores: BTreeMap<OptionId, u64>,

pub turnout: Turnout, // {ballots\_cast, invalid\_or\_blank, valid\_ballots}

pub round\_log: Option<IrvRoundLog>,

pub pairwise: Option<PairwiseMatrix>,

}

pub fn tabulate\_all(ctx: &LoadedContext, p: &Params) -> BTreeMap<UnitId, UnitScores>;

fn tabulate\_plurality(unit\_in: &UnitInput) -> UnitScores;

fn tabulate\_approval(unit\_in: &UnitInput) -> UnitScores;

fn tabulate\_score(unit\_in: &UnitInput, p: &Params) -> UnitScores;

fn tabulate\_ranked\_irv(unit\_in: &UnitInput, p: &Params) -> (UnitScores, Option<TieContext>);

fn tabulate\_ranked\_condorcet(unit\_in: &UnitInput, p: &Params) -> UnitScores; // completion via VM-VAR-005

*(TieContext is recorded if an IRV elimination tie blocks progress; pipeline can act in* ***RESOLVE\_TIES*** *later.)*

## **7) Algorithm Outline (by ballot type)**

**Plurality**: scores[opt] = votes. Support% for gates later uses valid\_ballots as denominator.

**Approval**: scores[opt] = approvals. **Approval gate** later uses **approval rate = approvals\_for\_change / valid\_ballots** (fixed).

**Score**: scores[opt] = Σ scores (apply normalization if VM-VAR-004=linear). Gate support (if binary change) uses spec’d ratio; means may be reported later, not used for allocation.

**Ranked IRV**: iterate rounds; eliminate lowest **continuing** tally; transfer next preferences; **denominator shrinks** when ballots exhaust (fixed policy). Emit RoundLog. If lowest-tally tie blocks elimination, return TieContext (no RNG here).

**Ranked Condorcet**: build pairwise matrix; if Condorcet winner exists, that’s the unit winner; else apply VM-VAR-005 completion. Emit matrix.

**Blank/invalid handling (all types):** count in ballots\_cast, excluded from valid\_ballots; if VM-VAR-007=on, inclusion only affects **gates** later, not tabulation.

## **8) State Flow**

Pipeline: **LOAD → VALIDATE → TABULATE → ALLOCATE** → …; UnitScores feed allocation; step order is fixed.

## **9) Determinism & Numeric Rules**

Integer/rational math; **round half to even** only where defined (none in tabulation except score normalization math if needed). Stable orders: Units by ID; Options by (order\_index, id). No RNG in this stage.

## **10) Edge Cases & Failure Policy**

Unknown ballot type → typed error.

Ranked IRV with all ballots exhausted → last continuing set decides per IRV rules; if still ambiguous, record TieContext for later resolution stage.

Zero valid\_ballots in a unit → scores all zero; downstream allocation/labels handle.

Tally sanity issues should have been caught in **VALIDATE**; guard asserts remain.

## **11) Test Checklist (must pass)**

**VM-TST-001** pipeline path: approval → Sainte-Laguë later yields **1/2/3/4** with our UnitScores.

**VM-TST-002** supports plurality tallies feeding WTA later (m=1).

**Ranked fixtures (VM-TST-010/011)**: IRV RoundLog shows shrinking continuing denominator; Condorcet completion per VM-VAR-005.

Determinism: shuffling option/unit input order yields identical UnitScores after stable ordering. Defaults per Annex B Part 0 respected.