# Survivor → Millions: DVOA Merge Step‑by‑Step Guide

This walkthrough explains exactly how your weekly DVOA snapshot flows into **Survivor** and can be reused by **Millions** with the same column logic. It’s written to be friendly for newer Python users and shows what to run, what to expect, and how to troubleshoot.

## 0) What you run each week (high level)

(1) Run your ingest: ingest\_ftn\_dvoa\_snapshot.py → data/ftn/dvoa/dvoa\_weekly.csv  
(2) Join/validate for Millions: scripts/validate\_and\_join\_dvoa.py → data/millions\_planner\_with\_dvoa.csv  
(3) Generate Survivor roadmap (your generator) → picks/survivor/survivor\_roadmap\_expanded.csv

**Single source of truth:** picks/survivor/survivor\_roadmap\_expanded.csv already carries the standardized DVOA columns. Millions can read the same columns to stay consistent.

## 1) Inputs and expected columns

**A. DVOA snapshot (CSV)**

* Path (default): data/ftn/dvoa/dvoa\_weekly.csv
* Flexible column names; the adapter auto-detects:
  + Team key: team / team\_abbr / abbr
  + Total DVOA: total\_dvoa / overall\_dvoa / dvoa / team\_dvoa
  + Offensive: off\_dvoa (or similar)
  + Defensive: def\_dvoa (or similar)
  + Optional: season, week

**B. Team alias map (CSV)**

* Path: data/seeds/team\_aliases.csv
* Purpose: normalize everything to canonical abbreviations (e.g., **WSH** not WAS; **JAX** not JAC).

**C. Planner / Survivor schedule**

* Millions: data/millions\_planner.csv
* Survivor: picks/survivor/survivor\_roadmap\_expanded.csv (final)

## 2) Normalization rules (identical for Survivor & Millions)

1. **Team codes** → Apply team\_aliases.csv to both the DVOA file **and** your planner/survivor file on the columns team and opponent.
2. **Week & season** → Filter to --season and --week *if those columns exist*; otherwise stamp them from CLI args.
3. **Total DVOA fallback** → If total\_dvoa is missing but off\_dvoa and def\_dvoa exist, compute total\_dvoa = off\_dvoa − def\_dvoa.

## 3) Join logic (how the columns are created)

We form two joins to attach DVOA for **team** and **opponent**:

1. Load planner/survivor as p (must have team and opponent).
2. Load and normalize DVOA as d (one row per team).
3. Merge **team DVOA**:
   * p = p.merge(d.add\_prefix("team\_"), left\_on="team", right\_on="team\_team", how="left")
   * Drops team\_team helper column.
4. Merge **opponent DVOA**:
   * p = p.merge(d.add\_prefix("opp\_"), left\_on="opponent", right\_on="opp\_team", how="left")
   * Drops opp\_team helper column.
5. Create **diff**:
   * dvoa\_diff = team\_total\_dvoa − opp\_total\_dvoa

**Columns you’ll see:**

* team\_total\_dvoa, opp\_total\_dvoa
* (optionally) team\_off\_dvoa, team\_def\_dvoa, opp\_off\_dvoa, opp\_def\_dvoa
* dvoa\_diff

This is the same schema we standardized for Survivor and Millions, so they stay perfectly in sync.

## 4) Commands you can run

### A) Millions – validate & join

# Activate your venv first  
.\venv\Scripts\Activate.ps1  
  
python -m scripts.validate\_and\_join\_dvoa `  
 --season 2025 `  
 --week 1 `  
 --planner data/millions\_planner.csv `  
 --dvoa data/ftn/dvoa/dvoa\_weekly.csv `  
 --aliases data/seeds/team\_aliases.csv `  
 --out data/millions\_planner\_with\_dvoa.csv `  
 --show

**What you should see:** a short schema report (which columns were mapped), then a small preview with team\_total\_dvoa, opp\_total\_dvoa, and dvoa\_diff.

### B) Survivor – ensure clean expanded file (only if needed)

If you ever see the “first 4 preloaded rows” or non-numeric week rows:

python -m scripts.survivor\_dvoa\_align\_and\_clean `  
 --in picks/survivor/survivor\_roadmap\_expanded.csv `  
 --aliases data/seeds/team\_aliases.csv `  
 --drop-top 4 `  
 --out picks/survivor/survivor\_roadmap\_expanded\_clean.csv `  
 --show

**Normal case:** you don’t need the \_clean file day-to-day. The master remains survivor\_roadmap\_expanded.csv.

## 5) Sanity checks (quick wins)

* **Team count**: exactly 32 teams present for a given week in the merged output. If < 28, check for alias mismatches.
* **Value ranges**: DVOA usually sits roughly between **−60% and +60%**; anything beyond that likely means a mapping error or bad week filter.
* **Freshness**: The adapter adds updated\_at so you can confirm last update time.

## 6) Troubleshooting

* **Missing** \*\* or \*\*: Some schedule exports are game-level (home/away) only. Duplicate each game into two rows (one per team) before the join, or use your existing generator that already emits team-level rows.
* **Aliases not applied**: Make sure data/seeds/team\_aliases.csv exists and includes entries like WAS→WSH, JAC→JAX, NOR→NO, OAK→LV, etc.
* **Different column names**: The adapter auto-detects common variants. If your ingest changes naming, update the candidate lists in the script once and everything stays stable.
* **Preloaded rows in Survivor**: Use --drop-top 4 and/or the numeric week filter. This removes header-like rows without touching real data.

## 7) Reusing Survivor output in Millions (optional)

If you prefer to avoid two separate joins, Millions can read from `` directly (it already has the standardized DVOA columns). In that case, just select the columns you need for Millions and skip a second DVOA join.

## 8) Next small enhancements (nice-to-haves)

* Add a **pre-commit check** that fails if the weekly DVOA file doesn’t have 32 normalized teams.
* Add a tiny **unit test** that validates dvoa\_diff = team\_total\_dvoa − opp\_total\_dvoa for a sample row.
* Attach a source column in the DVOA snapshot (e.g., ftn) for auditing.

**You’re set.** Run the adapter for Millions, keep the Survivor roadmap as the master, and both tools will stay aligned on the exact same DVOA columns.