CompoundWise.io

METHODOLOGY & MONEY-MATH REFERENCE

How every number is made — and why you can trust it.

A complete, plain-language walk through every engine and calculation inside CompoundWise: what each one does, why it exists, exactly how it is computed, and the rule or standard it answers to. Written to be read end-to-end by a first-time investor and audited line-by-line by a CFA.

DOCUMENT	AUDIENCE	SCOPE
The Methodology	Investors · CFAs · End users	Every subtool, exhaustively

JURISDICTION

Canada (CRA / Income Tax Act)

30	14	~5,300	1
engines & calculations, one method	valuation models, blended or withheld	backend tests gating every ship	source of truth per number

Start here

How to read this document

CompoundWise is a research platform for long-horizon investors: it scores companies on business quality and price, estimates what they are worth, tracks a real portfolio to the penny with Canadian tax correctness, and turns all of that into a short list of decisions. This document opens the hood on *all* of it.

Every engine below is explained in the same five-part rhythm, so a beginner is never left behind and a professional can skip straight to the math:

- **IN PLAIN WORDS** — the idea with no jargon, the "lemonade-stand" version.
- **UNDER THE HOOD** — the exact formula, thresholds, and edge-cases, with the source file named.
- **HOW IT CONFORMS** — the CRA rule / **Income Tax Act** section, or the professional standard (CFA Institute / GIPS) it satisfies.
- **WHY YOU CAN TRUST IT** — the single source of truth in code, and the automated tests that lock it.
- **THE HONEST LIMIT** — what the engine deliberately does *not* claim. This is the most important voice in the whole document.

NEW TO INVESTING

Start with plain words

Read the *In plain words* voice in each chapter and follow the figures. Skip the formulas — you'll still understand every decision the tool makes and why.

CFA · AUDITOR

Jump to the math

Go straight to *Under the hood* and *How it conforms*. Every rule cites its ITA section or GIPS/CFA standard, with the test that locks it in Appendix C.

ENGINEER

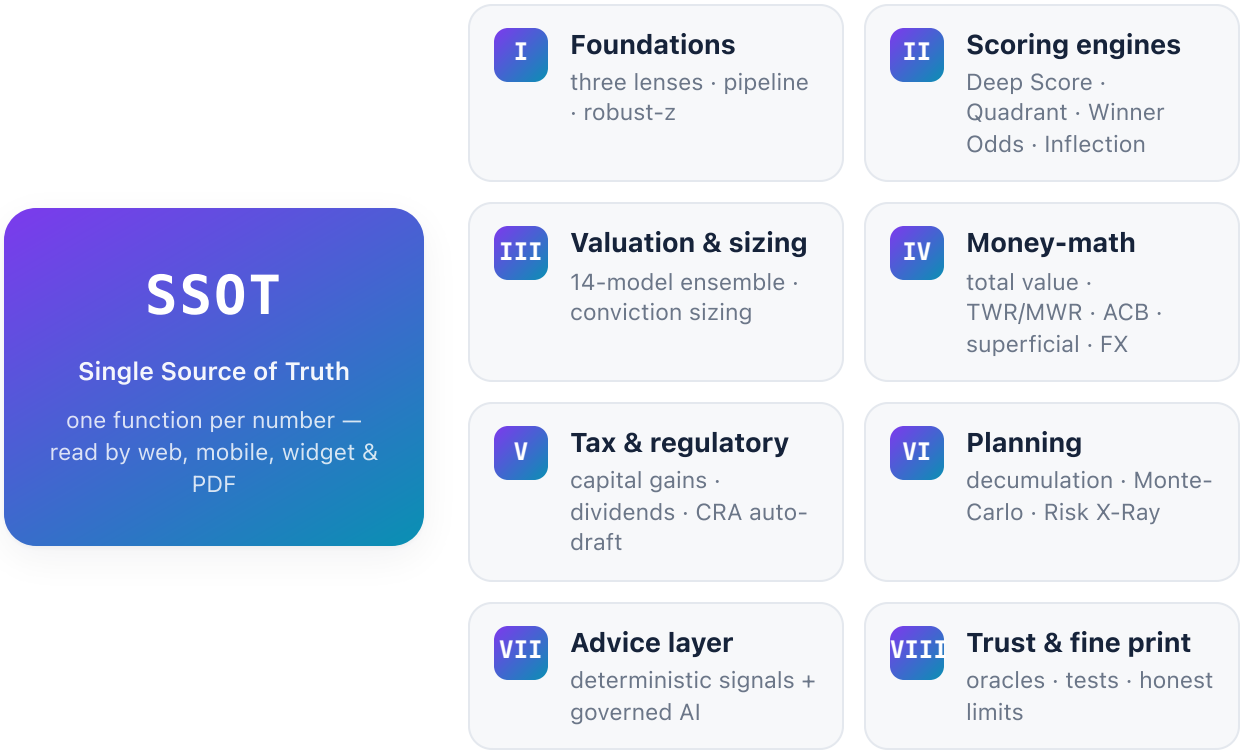
Trace it to source

Each engine names its one source-of-truth function and test file. The formula index (Appendix B) maps every number to the code that owns it.

THE ONE IDEA THAT TIES IT TOGETHER

CompoundWise is built on a rule its authors call **SSOT — a Single Source of Truth**. Every number that matters is computed in exactly *one* place in the code, and every screen (web, mobile, PDF report) reads that same place. That is why your "total value" is identical everywhere, and why a tax figure on the Advisor page can never quietly disagree with the Tax Centre. Throughout this document, "the SSOT for X" means "the one function that owns X."

FIGURE · THE SYSTEM AT A GLANCE — ONE SOURCE OF TRUTH, EIGHT DOMAINS



Everything in this document hangs off one rule: each number is computed once, in a single function, and every surface reads it. The eight domains map to **Parts I–VIII** below.

Terms shown with a dotted underline carry a hover-definition; all of them are collected in Appendix A. Code locations are cited as `module.py` so any claim can be traced to the exact source.

PART I · FOUNDATIONS

1 **What CompoundWise is (and is not)** — three lenses, by design

2 **Where the numbers come from** — data sources & the scheduled pipeline

3 **The shared statistics core** — winsorized robust-z & percentile rank

PART II · THE SCORING ENGINES

4 **Deep Score** — the Screener's quality-value engine

5 **The Value-vs-Quality Map** — the four quadrants

6 **Winner Odds** — the durable-compounder ranker

7 **Why Winner Odds ships rank-only** — the honesty gate

8 **Emerging Compounders** — the Inflection engine

9 **Discovery** — the open-universe bridge

PART III · VALUATION & SIZING

10 **What's it worth?** — the fair-value ensemble

11 **How much to buy** — position sizing

PART IV · THE MONEY-MATH

12 **Total Value** — the equity SSOT

13 **Am I beating the index?** — TWR, MWR & the benchmark twin

14 **The performance curve** — splits, CDRs, reconstruction

15 **Adjusted Cost Base** — the CRA cost-base replay

16 **Superficial-loss rules** — the regulatory showpiece

17 **In-kind & account transfers** — deemed dispositions

18 **The FX chokepoint** — one currency source of truth

PART V · TAX & REGULATORY

19 **Capital-gains tax**

20 **Dividend tax** — gross-up, DTC & foreign credit

21 **The tax-rules config & CRA auto-draft**

PART VI · PLANNING ENGINES

22 **Retirement income** — decumulation

23 **Will it last?** — Monte-Carlo simulation

24 **True net worth & asset mix**

25 **Risk X-Ray** — concentration & effective bets

PART VII · THE ADVICE LAYER

26 **The Advisor** — deterministic signals + AI narrative

27 **Strategy & Rebalance**

28 **Action Center, Market Pulse & macro**

PART VIII · TRUST & THE FINE PRINT

29 **How we know it's right**

30 **Disclosures & deliberate limitations**

APPENDICES

A **Glossary** — every term, defined

B **Formula index & source map**

C **Regulatory & Standards Conformance Matrix**

PART I

Foundations

Before any single tool: what the platform is trying to do, where its raw numbers come from, and the one piece of statistics that every ranking engine shares.

Chapter 1

What CompoundWise is — and what it is not

A research aid for people who want to own good businesses for 5–10 years — not a trading app.

CompoundWise screens publicly traded companies on **business quality** — profitability, growth durability, financial safety — and on **price**, then presents a scored, structured view to support a decision. It also tracks your real portfolio with Canadian-tax accuracy and helps you plan. It deliberately does *not* tell you to trade in and out; it is engineered to filter short-term noise and focus attention on the businesses most likely to compound wealth over years.

The most important design decision in the whole product is that it looks at every company through **three different lenses that answer three different questions**. They are kept mathematically separate on purpose — a great business you already recognise, a business that is *accelerating* before the crowd notices, and a curated rank of possible future big winners are not the same thing, and a tool that blurs them misleads you.

FIGURE 1 · THREE LENSES, THREE QUESTIONS — KEPT ORTHOGONAL BY DESIGN

SAME COMPANY · THREE ANSWERS

DEEP SCORE

The Screener engine

"How good is this business, right now?"

Reads trailing **levels** & averages — proven quality vs peers.

Abstains only when unmeasurable.

INFLECTION

Emerging Compounders

"Is it accelerating before the crowd?"

Reads the **change** (the 2nd derivative) — catches turns early.

Confidence-banded.

WINNER ODDS

The durable-compounder ranker

"Where does it rank for the next 10×?"

A cross-sectional multi-factor **rank** — a curated shortlist.

Abstains on thin data.

A mature, high-but-flat compounder can score **high** on Deep Score and **near-zero** on Inflection at the same time — that disagreement is a feature, not a bug. Keeping the lenses separate is what lets the product say precise, honest things. `Sources: deep_score_engine.py · inflection.py · winner_engine.py`

IN PLAIN WORDS

Imagine three scouts watching the same team. One grades the players on their career stats so far (Deep Score). One watches for the rookie whose last three games are suddenly much better (Inflection). One ranks everyone by their odds of becoming a superstar, but refuses to quote exact odds because the season is too young to be sure (Winner Odds). CompoundWise runs all three scouts and never lets one pretend to be another.

What it deliberately is not

- **Not a robo-advisor and not advice.** It does not take custody of money, place trades, or issue personal recommendations you must follow. It is a research and bookkeeping aid; every action is yours.
- **Not a day-trading or options tool.** The whole design horizon is 5–10 years. There is no intraday signal, no leverage, no derivatives modelling.
- **Not a black box.** Every score decomposes into named factors with visible weights, and every dollar figure traces to one function in open code. This document exists so that claim can be checked.

THE HONEST LIMIT

No engine here forecasts a price. Deep Score is an *assessment of standing today*, Inflection is a *detection of change*, and Winner Odds is a *rank*. None is a promise about returns. Where a claim *could* be mistaken for a forecast, the product suppresses it — that discipline is the subject of Chapters 7 and 30.

Chapter 2

Where the numbers come from — data sources & the scheduled pipeline

Good analysis starts with clean inputs and a disciplined refresh; here is exactly how the raw data arrives and in what order it is processed.

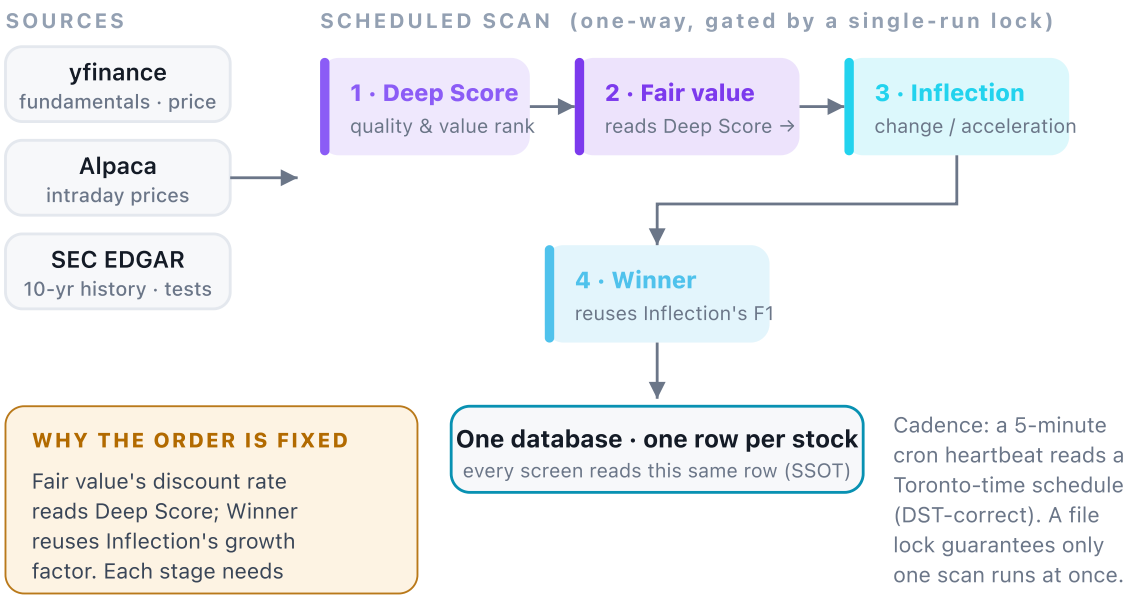
Nothing downstream can be trustworthy if the inputs are wrong, so CompoundWise separates **live market data** (prices, fundamentals, analyst estimates) from **deep historical statements** (used only to test the models), and runs a fixed scheduled sequence so that each engine only ever reads finished work from the engine before it.

The two data sources, and why there are two

- **Live data — yfinance & Alpaca.** Each scan the scanner pulls the company "info" block (market cap, price, sector, beta, analyst targets), the income / balance / cash-flow statements, the insider-trading feed, and the analyst-estimate frames. Prices refresh every few minutes during market hours. `scanner.py · alpaca_prices.py`
- **Deep history — SEC** EDGAR**.** For U.S. companies the platform can read up to a decade of official 10-K filings straight from the SEC's XBRL feed, validated to match yfinance "to the dollar." For U.S. filers it is also the *primary* source of the Deep Score's multi-year history; the four-year yfinance statements are the fallback for foreign private issuers, for filers whose tags the reader cannot use, and for any filer when the SEC cannot be reached — an unreachable SEC degrades that night's history, it never drops a company from the scan. It is capped at the most recent ~10 years and keyed on each fact's start/end dates so a filing's comparatives can't collapse two years into one. `sec_edgar.py` *How lines are read (2026-09-16, verified against SEC bulk companyfacts for every U.S. name in the universe):* a total is the per-year maximum across its tags, never one winning tag — revenue tagged only under the ASC 606 slice, or a utility's plant tagged only as non-utility PPE, was reading a fraction of the true figure; banks and broker-dealers use their own revenue definition (net interest income plus non-interest income, or revenues net of interest expense); EBIT is derived as pretax income plus interest where no operating-income line is filed; a filer that never tags income tax is a pass-through whose pretax income is its net income; a year with no figure is **unknown, never 0**; a fiscal year exists when a statement exists — a pre-revenue filer's reported zero revenue is a measurement beside its real balance sheet and cash burn, not a missing year (2026-09-17); where a filer tags no revenue at all, **net income drives the years**, so a clinical-stage biotech or an agency mortgage REIT is read rather than discarded, with revenue left unknown; the annual report may be a **20-F or 40-**

F, since a foreign issuer reporting under US GAAP files the same facts; a partnership's equity is read from partners' capital, which is what an LP files; a history whose revenue tag lapsed more than two fiscal years ago is unusable rather than served as recent; and a share count is restated in today's shares by its *filing vintage*: each year's count comes from the year's own 10-K (a later filing's equity roll-forward carries the prior year-end unrestated, so "filed after the split" never meant "restated"; a count that disagrees with its own filing's weighted average by more than 1.5× is replaced by the average, which ASC 260 requires restated), and a split is applied to the counts filed before it only where the counts confirm it (a factor within 15% of 1 is never a split), or where a clean ratio is not contradicted — the vendor's split series also carries spin-off price factors (EQT's 1.837) that never changed a share count. A blanket multiplier had counted RLI's 2:1 twice; a jump test at the split date had missed the step a mixed-vintage series carries inside it. Share units (thousands against units) are reconciled *after* that restatement, so a large cumulative split (NVIDIA's 40×) is never mistaken for a unit error; a revenue year filed only under a tag the filer has since retired fills a gap inside the live years, and a retired tag extends them backwards only where its years run contiguously into the live ones (tag succession, as when a lessor REIT's `Revenues` gives way to ASC 842 lease income, which is now read as revenue); and the ticker-to-CIK map falls back to the SEC's plain ticker list for the filers its JSON omits (AvalonBay, Equity Residential).

FIGURE 2 · THE SCHEDULED PIPELINE — A FIXED, ONE-DIRECTIONAL ORDER



The scan order is `Deep Score → Fair Value → Inflection → Winner`. It is one-directional so no engine ever reads a half-computed input, and it is protected by an advisory lock so two scans can never collide. `Source: scanner.py (run order ~L1421–1434) · scan_dispatch.py · deploy/gcp/install-scan-cron.sh`

UNDER THE HOOD · DATA HYGIENE THAT PREVENTS FAMOUS BUGS

Two guards live at the data door. **Phantom cross-listings** — U.S. companies whose Canadian depositary receipts (CDR) can arrive carrying the underlying's multi-trillion-dollar market cap next to the receipt's small price (the notorious "NVDA.TO shows $6.4 trillion" artifact) — are held on a small reviewed list and skipped at fetch, with any stale rows purged. **Missing values** become a true `None` that drops out of an average, never a fake `0` that would masquerade as data. `scanner.py`

WHY YOU CAN TRUST IT

The refresh schedule lives in a database table read by a heartbeat, not hard-coded, so it stays correct across daylight-saving changes. Every scan is single-flighted by a lock. And because everything lands in one row per stock that every surface reads, a number cannot differ between the web app, the phone app, and a PDF report.

UNDER THE HOOD · EVERY PRICE SAYS WHEN IT WAS TRUE

Each stored close carries the moment it was observed and the feed it came from (Alpaca IEX or Yahoo Finance, stamped by the intraday refresh; the scan's date when no refresh has touched it). Held prices show **"as of ..."**, **"live"** only for a quote under an hour old while the exchange is open (still delayed, not real time), and an amber **"stale"** once a newer session has opened on that holding's own exchange calendar — so a Canadian price is not stale on a TSX holiday. A holding with no price shows no date at all, never a made-up one. Each macro figure names its source — the Bank of Canada, FRED, market data, or an AI web-search extraction, which says so where it is read. One decider makes the stale/live call for every screen. `price_asof.py` · `tests/test_price_asof.py`

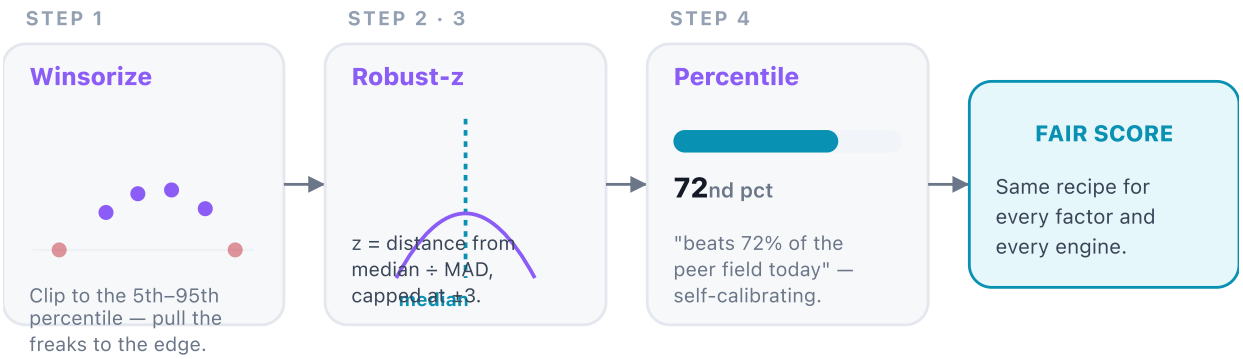
Chapter 3

The shared statistics core — winsorized robust-z & percentile rank

The one piece of math every ranking engine reuses: a fair, outlier-proof way to say "how far above or below the typical peer is this?"

Comparing raw numbers across companies is a trap: one company with a freak 900% return-on-equity would dominate any simple average and make everyone else look identical. So all three scoring engines route through a single, robust routine that (1) clips the wild extremes, (2) measures distance from the *typical* peer using median-based math a single outlier can't distort, and (3) converts that to an easy-to-read percentile. Building it once means every engine is fair in exactly the same way. `cross_sectional_stats.py`

FIGURE 3 · FROM A MESSY RAW FACTOR TO A FAIR 0–100 PERCENTILE



Because the score is a **rank within today's actual field**, it always uses the full 0–100 range — it can't silently drift the way a fixed-threshold score does when the whole market gets more expensive. `cross_sectional_stats.py · winsorized_moments, robust_z, fractional_percentile`

UNDER THE HOOD

```
# 1. Winsorize the cross-section, then measure spread robustly clip
each value to [ 5th pct , 95th pct ] location = median(clipped)
scale   = MAD × 1.4826 # MAD = median absolute deviation;
                       # ×1.4826 makes it match a normal σ # 2. Turn
one value into a robust z-score (direction-aware) z = clamp( (clip(x) -
median) / scale , -3 , +3 ) × direction # 3. Rank it inside the peer
field percentile = count(peers ≤ x) / n # 0 … 1
```

Why median and MAD, not mean and standard deviation? The mean and standard deviation are themselves yanked around by outliers — the very thing we're trying to neutralise. The median (middle value) and MAD (typical distance from the middle) are unmoved by a handful of extremes. Two safety valves: a factor needs at least **5** real values to be scored at all, and if a factor is so clustered that its MAD collapses to zero the routine falls back to a normal standard deviation rather than dividing by zero.

HOW IT CONFORMS · METHODOLOGICAL STANDARD

Winsorization and robust (median/MAD) standardisation are textbook **robust-statistics** practice for cross-sectional factor work — the same family of techniques used in quantitative equity research to keep a few bad data points from dominating a ranking. Nothing here is proprietary or hidden; it is a defensible, conventional way to compare companies fairly. The percentile step is a plain empirical rank.

WHY YOU CAN TRUST IT

This routine exists in exactly one file and is imported by Deep Score, Winner Odds, and Inflection alike, so "fairness" is defined once and cannot drift between engines. Its behaviour — including the ≥5-point rule and the zero-MAD fallback — is pinned by unit tests.

PART II

The scoring engines

The heart of the product: how a company earns a Deep Score, where it lands on the Value-vs-Quality Map, how it is ranked for future upside, how acceleration is detected, and how new names enter the universe.

Chapter 4

Deep Score — the Screener's quality-value engine

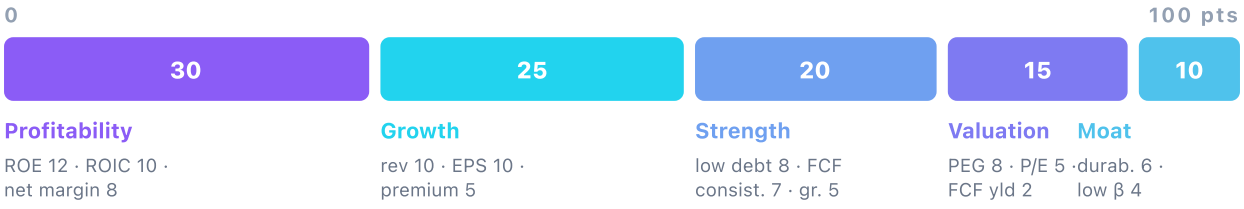
A single 0–100 checkup that answers "how good is this business, and how reasonably priced, compared with companies like it?"

Deep Score is the number that powers the Screener. It grades every company on five things that decide long-run compounding — **profitability, growth, financial strength, valuation, and moat** — and blends them into one score out of 100 with a plain label (STRONG BUY down to AVOID). Crucially, it grades each company against *its own kind*: a bank is judged against banks, a small miner against small miners. A 15% profit margin is spectacular for a grocer and mediocre for a software firm; comparing everyone to one universal bar would be nonsense.

The five pillars and their 100 points

The scoreboard is fixed and public. Fourteen factors sit in five pillars; the points show how much each matters. Every factor belongs to exactly one pillar — an old version of the scorer double-counted a couple of factors, and a test now forbids that.

FIGURE 4 · THE DEEP SCORE SCOREBOARD — 14 FACTORS, 5 PILLARS, 100 POINTS



Profitability & Returns (does the business earn well on its capital?) carries the most weight; **Moat** the least. Within a pillar, the small numbers are each factor's prior weight. `Source: deep_score_model.py — FACTORS, PILLAR_MAX`

Pillar (max pts)	Factor	Plain meaning	Wt (measured IC, 2026-09-28)
Profitability & Returns (30)	ROE	Profit per dollar owners put in	12
	ROIC	Profit per dollar of all capital	10
	Net margin	Cents of profit per sales dollar	8
Growth (25)	Revenue growth	Are sales getting bigger?	12
	Earnings growth	Are profits getting bigger?	13
Financial Strength (20)	Low leverage (D/E)	Not drowning in debt	8
	FCF consistency	How often it produces real cash — positive years as a share of the years actually MEASURED, never of the calendar. A year whose free cash flow cannot be computed is unknown, not a failure; a company with no measurable year abstains from the factor rather than scoring the worst value in the field.	7
	FCF growth	Is that cash growing?	5

Per-class factor sets. Banks, lenders, insurers, equity REITs, regulated utilities, commodity producers and software companies are each scored on a factor set built for their economics (the tables below) and ranked only against their own kind. **Balance-sheet financials** — banks, deposit-funded lenders, broker-dealers, insurers and mortgage REITs — are scored on a balance-sheet basis for this pillar and for FCF yield: **assets to tangible equity** in place of D/E, **profitable-year consistency** in place of FCF consistency, **book value per share growth** in place of FCF growth, and **tangible book yield** in place of FCF yield. Free cash flow is deposit, loan and reserve flow for these businesses, not cash generation, and their leverage is the business model. They are ranked only against their own kind (spread lenders as one peer group, insurers as another), never inside the wider Financial Services sector — the substitution is only sound inside a cohort measured on one basis. A name is a spread lender when net interest income is at least 10% of its revenue (or, when that line is missing from the vendor feed, when its industry is a bank); an insurer by industry; a mortgage REIT by industry. **A fund is not a lender.** A closed-end fund or business development company earns interest income exactly as a bank does, so the income test

Pillar (max pts)	Factor	Plain meaning	Wt (measured IC, 2026-09-28)
alone cannot tell them apart — the balance sheet can: a lender funds assets with liabilities (11.6x assets to tangible equity, median, on the 2026-09-17 scan) while a fund holds a portfolio against its own capital (1.7x, and capped near 3x by the Investment Company Act). A name that presents as a lender by income but carries under 3x, or whose own SEC filings report the portfolio it holds and the income it throws off, is classed an **investment company** and is **not scored at all** — it has no return on equity, operating margin or revenue growth in the sense these five pillars mean, and it receives no fair value either, because its worth is the net asset value of what it holds and none of the models here computes that. Before this rule, 36 such names sat in the bank cohort and six published as Strong Buy. **That rule is an American one, and a reviewed list carries the rest.** The 3× leverage line comes from the U.S. Investment Company Act and the filing test reads SEC filings, so a Canadian closed-end fund or split-share corporation — which has no SEC filer at all — is structurally invisible to both. Those are classed by a reviewed list instead (`data/class_review/reviewed_classification.csv`), where each entry carries a cited source, the reviewer, and a date by which it must be re-checked, and a reviewed classification outranks every automatic detector. On 2026-09-24 two such funds were found publishing fair values at MEDIUM confidence — a trusted tier, so they were driving position sizing — and were added. Capital adequacy is measured *unweighted*: CET1, risk-weighted assets and loan-loss provisions are regulatory disclosures absent from the vendor feed, so Canadian banks score worse than US regionals partly because insured-mortgage risk weighting is invisible here.			
Valuation (15)	PEG	Price fair for the growth?	8
	P/E	Price vs profits	5
	FCF yield	Cash return at today's price	2
Moat & Quality (10)	Margin durability	Margins holding vs their own history	6
	Earnings variability	How steady its own profits are, year to year	4
Why beta is no longer a quality factor. Until 2026-09-17 this pillar scored *low beta* — how calm a share price is against the market. That is *systematic* risk, and the fair-value engine already charges for it: the discount rate is `risk-free + beta × equity risk premium + a quality adjustment`, and the quality adjustment is built from this very pillar. A low-beta company therefore earned a quality credit that lowered its own discount rate and raised its own fair value — beta priced twice in one equation. It is now priced once, where it belongs, in the cost of equity. Its four points went to **earnings variability**, which measures how steady *this company's own* profits			

Pillar (max pts)	Factor	Plain meaning	Wt (measured IC, 2026-09-28)

have been with no reference to the market — the same descriptor MSCI's quality index uses, and the kind of stability a pillar about business quality should be measuring.

Each class below is a full five-pillar set with the same pillar maxima as the default; the composite, the label cliffs and the quality score are unchanged. A class is ranked only against its own kind. Generated from `deep_score_model.FACTOR_SETS` by `scripts/render_factor_sets.py`.

Banks, lenders and broker-dealers (spread lenders, incl. mortgage REITs)

Pillar	Factor	Direction	Points
Profitability & Returns	Return on equity	higher is better	12
Profitability & Returns	Return on tangible equity	higher is better	10
Profitability & Returns	Net margin	higher is better	8
Growth	Revenue growth	higher is better	12
Growth	Earnings growth	higher is better	13
Financial Strength	Low leverage (assets/tangible equity)	lower is better	8
Financial Strength	Profitable-year consistency	higher is better	7
Financial Strength	Book value per share growth	higher is better	5
Valuation	P/E ratio	lower is better	6
Valuation	Tangible book yield	higher is better	6
Valuation	PEG (growth-adjusted)	lower is better	3
Moat & Quality	Margin durability (current vs. history)	higher is better	6
Moat & Quality	Earnings variability (own history)	lower is better	4

Insurers

Pillar	Factor	Direction	Points
Profitability & Returns	Return on equity	higher is better	12
Profitability & Returns	Claims / total revenue (loss-ratio proxy)	lower is better	10
Profitability & Returns	Net margin	higher is better	8
Growth	Premium and revenue growth	higher is better	10
Growth	Earnings growth	higher is better	8
Growth	Book value per share growth	higher is better	7
Financial Strength	Low financial leverage (debt / capital)	lower is better	10
Financial Strength	Profitable-year consistency	higher is better	10
Valuation	P/E ratio	lower is better	6
Valuation	Tangible book yield	higher is better	6
Valuation	PEG (growth-adjusted)	lower is better	3
Moat & Quality	Margin durability (current vs. history)	higher is better	6
Moat & Quality	Earnings variability (own history)	lower is better	4

Equity REITs

Pillar	Factor	Direction	Points
Profitability & Returns	FFO return on equity	higher is better	12
Profitability & Returns	FFO margin	higher is better	10
Profitability & Returns	Operating margin	higher is better	8
Growth	Revenue growth	higher is better	10
Growth	FFO per share growth	higher is better	15
Financial Strength	Low debt / EBITDA	lower is better	8
Financial Strength	Interest coverage (EBIT / interest)	higher is better	7
Financial Strength	FFO payout (lower is safer)	lower is better	5
Valuation	FFO yield (1 / P/FFO)	higher is better	8
Valuation	Dividend yield	higher is better	4
Valuation	Tangible book yield	higher is better	3
Moat & Quality	Operating margin vs. history	higher is better	6
Moat & Quality	Earnings variability (own history)	lower is better	4

Utilities (regulated, merchant and renewable)

Pillar	Factor	Direction	Points
Profitability & Returns	Earned return on equity	higher is better	12
Profitability & Returns	Return on invested capital	higher is better	10
Profitability & Returns	Net margin	higher is better	8
Growth	Asset-base growth (net PPE)	higher is better	12
Growth	Earnings growth	higher is better	8
Growth	Revenue growth	higher is better	5
Financial Strength	Operating cash flow / debt	higher is better	8
Financial Strength	Interest coverage (EBIT / interest)	higher is better	7
Financial Strength	Earnings payout (lower is safer)	lower is better	5
Valuation	P/E ratio	lower is better	7
Valuation	Dividend yield	higher is better	5
Valuation	PEG (growth-adjusted)	lower is better	3
Moat & Quality	Margin durability (current vs. history)	higher is better	6
Moat & Quality	Earnings variability (own history)	lower is better	4

Dividend facts — what is known, and who reads which figure (2026-10-13; the valuation basis 2026-10-15). The scan stores a **valuation dividend**, decided once per company by one rule: the declared **forward rate** when the source gives one; **0.0 only when the source states the company pays nothing** — a zero trailing rate *and* a zero yield, with no positive forward rate (a **known zero**); otherwise the source's **forward dividend yield × price** — the **implied** basis, in the share price's own currency — *if it passes a cross-check*; otherwise unknown. Between the 13 and 14 October 2026 scans the source stopped sending the forward rate for paying companies (1,659 of the 1,673 payers lost it, and 1,120 of those fair values moved by more than 1% because the dividend-discount model could not run), so the implied basis now carries most payers. **Why not the dividend paid over the last 12 months.** The source reports that figure in the company's *home* currency — a euro ADR's reads about 0.86× its dollar dividend, a Japanese ADR's is in yen — and it lags cuts and raises. A first design valued it and was withdrawn before release when a review found exactly that. **The cross-check.** The yield (a percent, printed to two decimals) is refused when that rounding alone could move the dividend by more than 10% (a yield below 0.05%). Where the source states a positive trailing 12-month dividend, it must sit within **0.33–3×**

of the implied figure, and the implied figure must sit within the same band of the last valuation dividend stored for the company in the previous 30 scans (a figure that itself passed — so a value rejected one night is rejected again the next, rather than slipping through because the night before stored nothing; none in that window, or a known zero, is no reference). The band is wide on purpose: the implied figure is right on its own for almost every payer, so the check exists to catch *gross* disagreement — one quarter reported as the year (JNJ: trailing 1.30 against an implied 5.52, so JNJ is left unknown), a dividend in yen, won or pence against a dollar price, a yield reported as a fraction. Measured against the forward rate stored on 13 October over 1,673 payers, it accepts 1,563 implied dividends, 3 of them more than 20% off (0.2%, all on yields under 0.3%), and leaves 89 payers unknown whose implied figure was within 10% (5.3%), mostly Japanese and Korean ADRs whose trailing figure is in their home currency; a 0.80–1.25 band caught no more wrong figures and turned away 281 right ones. Re-run over the 1,659 payers that lost the forward rate (exported production fields and a live read of the source on 15 October, not a live re-scan), 1,560 get a valuation dividend back on the implied basis and 99 stay unknown — 88 whose trailing figure disagrees, 8 with no yield, 3 with a yield too coarse to use. A dividend left unknown is never a guess, and the scan log names the company and the reason. **The limit we cannot close:** the yield is the source's own forward figure and we cannot verify the date it was set. **An implied dividend says so wherever it is read**: the stock page's models panel prints a note (web and Android), the screener and the Tax Centre's withholding table mark it *impl.*, the Deep Score's dividend factors read "(dividend implied from yield)", and the RRSP-migration signal quotes the yield as "implied from Yahoo's forward yield". The **trailing rate** as the source reports it is also stored raw, and is read only for the cross-check and for whether a company pays. **One decider says whether a company pays**: a payer when the valuation dividend or the raw trailing rate is positive, a non-payer on a known zero, unknown otherwise — and every reader of "pays" (the suitability check for an income or preservation objective, the candidate screen) asks it. The **tax-placement yield** reads the valuation dividend only, so a payer whose dividend is unknown is *not assessed* there rather than placed on the wrong figure. **Valuation reads the valuation dividend only**: the dividend-discount model, sustainable growth (retention × ROE) and every other fair-value input never read the trailing rate. In the Deep Score the **dividend-yield** factor scores a known zero as a real 0% yield, while the **payout-safety** factors (FFO payout, earnings payout — lower is safer) **abstain** on a company that pays nothing, because a 0% payout would otherwise rank a non-payer, a suspended payer included, as the safest in its cohort. A known limit, kept by decision: sustainable growth still reads an *unknown* valuation dividend as no dividend (100% retention) — since 15 October, the 99 payers left unknown. `dividend_facts.valuation_dividend; scanner.fetch_fundamentals; tests/invariants/test_pays_dividend_one_decider.py`

Commodity producers

Pillar	Factor	Direction	Points
Profitability & Returns	Operating margin (current)	higher is better	10
Profitability & Returns	Through-cycle operating margin	higher is better	8
Profitability & Returns	Return on equity (multi-year)	higher is better	12
Growth	Book value per share growth	higher is better	10
Growth	Revenue growth	higher is better	8
Growth	Earnings growth	higher is better	7
Financial Strength	Low net debt / EBITDA	lower is better	8
Financial Strength	FCF consistency	higher is better	7
Financial Strength	Interest coverage (EBIT / interest)	higher is better	5
Valuation	FCF yield (current)	higher is better	8
Valuation	EV / through-cycle operating earnings	lower is better	4
Valuation	Tangible book yield	higher is better	3
Moat & Quality	Margin stability (low variability)	lower is better	6
Moat & Quality	Earnings variability (own history)	lower is better	4

Software

Pillar	Factor	Direction	Points
Profitability & Returns	Return on invested capital	higher is better	12
Profitability & Returns	SBC-adjusted FCF margin	higher is better	10
Profitability & Returns	Gross margin	higher is better	8
Growth	Rule of 40 (growth + FCF margin)	higher is better	20
Growth	Analyst +1y revenue growth	higher is better	5
Financial Strength	Low leverage (D/E)	lower is better	8
Financial Strength	FCF consistency	higher is better	7
Financial Strength	FCF growth	higher is better	5
Valuation	EV / revenue	lower is better	6
Valuation	FCF yield	higher is better	5
Valuation	PEG (growth-adjusted)	lower is better	4
Moat & Quality	Margin durability (current vs. history)	higher is better	6
Moat & Quality	Earnings variability (own history)	lower is better	4

Each class runs the models that mean something for its economics, at the policy weights below, on its own growth axis; the default class is valued exactly as before. Generated from `valuation/class_policy.py` by `scripts/render_factor_sets.py`.

Class	Anchor and models (weight)	Growth axis
Banks, lenders and broker-dealers (spread lenders, incl. mortgage REITs)	Justified P/TBV (2), Residual Income (1.5), Gordon Growth DDM (1), P/B Peer (1), P/E Peer (1), EPV (0.5), Graham Number (0.5)	retention × ROE (sustainable growth)
Insurers	Justified P/B (2), Residual Income (1.5), Gordon Growth DDM (1), P/B Peer (1), P/E Peer (1), EPV (0.5), Graham Number (0.5)	retention × ROE (sustainable growth)
Equity REITs	P/FFO Peer (2), FFO Perpetuity (1.5), EV/EBITDA Peer (1), Gordon Growth DDM (1), P/B Peer (0.5)	FFO per share growth
Utilities (regulated, merchant and renewable)	Gordon Growth DDM (2), P/E Peer (1.5), EV/EBITDA Peer (1), EV/Net PPE (1), Residual Income (1), EPV (0.5), Graham Number (0.5), P/B Peer (0.5)	rate-base (net PPE) growth
Commodity producers	Normalized EV/EBITDA (2), FCFF Perpetuity (1), FCFF Two-Stage (1), P/B Peer (1), P/E Peer (1), EPV (0.5), EV/EBITDA Peer (0.5), Gordon Growth DDM (0.5), Graham Number (0.5), Residual Income (0.5)	free-cash-flow history
Software	EV/Revenue (growth-adjusted) (2), EV/EBITDA Peer (1), FCFF Exit Multiple (1), FCFF Perpetuity (1), FCFF Two-Stage (1), P/FCF Peer (1), EPV (0.5), Lynch Fair Value (0.5), P/E Peer (0.5), P/S Peer (0.5)	free-cash-flow history
Closed-end funds and BDCs (investment companies)	*Not valued — a fund is a portfolio, and its worth is the net asset value of what it holds. No model here computes that, so no fair value is published.*	—

Class anchors: **Justified P/TBV** — FV = tangible book value per share x clamp((ROTE - g) / (k_e - g)) — the bank relation.; **Justified P/B** — FV = book value per share x clamp((ROE - g) / (k_e - g)) — the insurer relation (Nissim:; **P/FFO Peer** — FV = class-median P/FFO x FFO per share (Nareit). The REIT multiple, in place of P/E.; **FFO Perpetuity** — FV = AFFO/share x (1+g) / (k_e - g), AFFO = AFFO_RATIO x FFO, g = FFO/share CAGR capped.; **EV/Net PPE** — FV = (m x net PPE - debt + cash) / shares, m = class-median EV / net PPE — the regulated; **Normalized EV/EBITDA** — EV/EBITDA on THROUGH-CYCLE EBITDA (cycle-average operating margin x current revenue + D&A); **EV/Revenue (growth-adjusted)** — EV/Revenue = a + b x Rule-of-40, fitted inside the software cohort (static fallback), so

Discount-rate basis. One basis, since 2026-09-18: Damodaran's January-2026 implied mature-market premium of 4.23% plus a country premium (US 0.23%, CA 0.00%); the domicile's cached 10-year yield as the risk-free rate (US via ^TNX, Canada via the Bank of Canada), read from the app's macro cache and never fetched on a valuation path, with a static **per-domicile** yield standing in when the cache is absent, older than 72 hours or outside 2–7% — 4.3% for a U.S. row and 3.83% for a Canadian one, the Government of Canada 10-year benchmark (Bank of Canada series BD.CDN.10YR.DQ.YLD, observation 2026-09-22), and the source string says so when it is used. Until 2026-09-24 a single 4.3% stood in for *both*, so a stale cache silently put every Canadian row back on the U.S. rate and the Canadian basis ceased to exist until the cache refreshed; and an equity beta of 0.67 × the Damodaran industry levered beta + 0.33 × the firm's. A 5.5% historical premium was the alternative until a switch retired that day.

What the confidence tier is measured on. The engine discards any model whose answer falls outside a third to three times the share price — a band wide enough that crossing it means the input is broken. Until 2026-09-24 the tier then measured how much the *survivors* agreed, which is a circular test: the models that disagreed had just been removed for disagreeing. Re-run on the 2026-09-23 scan of 2,891 names, 1,031 of the 2,035 carrying a published tier would carry a different one had the disagreement been measured before that filter, and 85 of the 216 top-confidence labels were not earned. The tier now reads the disagreement that actually existed, and both figures are published so a reader can see the difference rather than take the label on trust. **The thresholds moved with it, and had to.** They were set when the typical spread was 0.27; on the new basis it is 0.46, so the old cut-off for "too much disagreement to publish" sat near the middle of the distribution instead of catching its tail — left unmoved it would have withdrawn 840 of the 2,035 published values as arithmetic rather than judgement. They were re-fitted by measurement against the live engine, and coverage lands at 1,997 names against 2,035. **The change is not fewer labels but different ones:** 200 top-confidence names against 216, now the ones whose models genuinely agreed. **And a clamped multiple no longer counts as a second opinion:** where a bank or insurer multiple lands on its [0.5, 3.0] rail it is a constant, which agrees with whatever sits beside it — it still informs the estimate, but a name whose corroboration rests on one cannot reach top confidence, which costs 18 of the 218 names that would otherwise get there. No fair value moves: 0 of 2,891 point estimates change. These cut-offs are a house calibration fitted to this book; no external convention sets them.

When the fair value is made — and why it can appear or vanish within a day. Because of that band, the blend is a function of the share price: as the price moves, a model can cross a third or three times it, the median the outlier trim is taken around shifts with it, and the number of corroborating families can fall below two. Until 2026-10-07 the blend was made once per scan and the price was then refreshed every two minutes underneath it, so the fair value on screen had been made at an earlier price — and the stock page’s models panel valued the stock a second time on vendor inputs, so one page could show two answers. Measured on 2026-10-07: of the 918 names with no published fair value, 10 were computable at the price then on screen; 1

published value no longer was; and 73 of the 1,971 that stayed computable moved by more than 5%. The fair value is now re-blended whenever the price is refreshed, over the whole peer cohort (a peer's market value moves the multiples too), it records the close it was made at, and every surface — the stock page, the models panel, the screener, the alerts, the apps — reads that one stored record. **What this does not smooth.** There is no hysteresis: a name whose models straddle the band's edge can gain and lose its estimate within a session. Micron on 2026-10-07 had no estimate at every price tested between $900 and $1,050 and about $1,590 at $1,070, because its models on normalised earnings ($85–$351) and its two on current EBITDA ($1,885 and $1,920) sit far apart, and which of them survive the band depends on the price. Where a name has no estimate, the page names the rule that withheld it. The verdict beside the blended value states the same margin of safety as the stock page's tile, (fair value – price) ÷ fair value: until 2026-10-07 it quoted the upside, fair value ÷ price – 1, so one page read "Undervalued +46.7%" beside a "+31.8%" margin of safety for the same price. No external convention sets this; it is a property of the house blend. `ensemble_store.py; price_refresh.refresh_prices; tests/invariants/test_fair_value_one_number.py`

Whose money a ratio is measured on. Preferred shareholders have a claim ahead of the common shareholder, so wherever one side of a ratio is common-only the other must be too. Three places did not match. **Return on tangible common equity** used net income *before* preferred dividends against an explicitly common denominator — of 201 banks publishing a fair value, 59 carry preferred, and the return was overstated by a median 3.6%, by 5% or more on 14, and by 220% on one. **The payout behind sustainable growth** divided the common dividend by total earnings per share, overstating what a bank retains. And **REIT funds from operations per share** divided the Nareit total by the common share count. All three now use the figure the common shareholder has a claim on, and one piece of code performs the subtraction. **The published effect is small, which is the honest number:** four fair values of 2,891 moved on the first two and twenty-five on the third, because the justified multiple is clamped and 24 of the 61 affected banks sit on a clamp rail where a change to the return cannot move the published figure. **One claim is still not subtracted, by decision:** the bridge from enterprise value to equity removes debt and adds cash, and leaves preferred and minority interests with the common shareholder on 167 of the 2,106 rows that publish a value. No balance sheet figure for either exists in the data — only the dividend paid — and turning a dividend into a balance needs an assumed rate, which would put an invented number where a visible gap is more honest. Each affected row says so on its own model panel.

What each model claims, and where it says so. A model named for a published convention is making a claim about method. Until 2026-09-24 none of the 21 recorded whether its formula actually followed the convention its name invokes, and three did not: **EPV** names Bruce Greenwald and capitalises normalised earnings per share at the cost of *equity*, where his construction is an enterprise calculation — normalised after-tax operating profit at the weighted cost of capital, then adjusted for debt and cash. **Graham Revised** described its denominator as the AAA corporate

yield and used the government 10-year, which is lower, so the model reads roughly 18% high (for a measured median blend effect of −0.67%; a Moody's feed was declined as not worth a new dependency for that). **Residual Income** damps its excess-return term by a house persistence factor of 0.5 that is no part of the standard formula. Each of the 21 now declares one of three things — it follows the convention, it is a *proxy* for it with a note saying what it really computes, or it invokes no external convention at all (a peer median claims nothing) — and a proxy is **labelled on the per-model panel itself**, where the model's name appears, rather than only in the register. Eight of the twenty-one are proxies. Where the source behind a convention is not something we hold a retrievable copy of, the entry records that too rather than implying a citation.

Whose FFO a REIT's per-share figures are built on. Nareit's Funds From Operations starts from net income attributable to the parent, which is *before* preferred dividends. That is the right numerator against revenue or against total equity, because both sides include the preferred — and the wrong one the moment the other side is common-only, as it is for a per-share figure or a market capitalisation. Until 2026-09-24 the engine divided the Nareit total by the **common** share count: of 119 equity REITs on the 2026-09-23 scan, 22 carry a preferred line and 21 of those publish a fair value, with FFO per share overstated by a median 1.8% and up to 20.1% — and it fed *both* REIT anchors, so the error landed twice in one blend. The peer P/FFO multiple had the mirror of the same fault, pairing market capitalisation with the total. Both now use FFO available to common; the class median was unchanged by it, because only 22 of 119 are affected and the median sits elsewhere.

What growth is measured over, and which way it may point. Until 2026-09-24 the projected growth rate was biased upward in four independent places, and this document named none of them while naming all three guards that pull a fair value down — which reassured a reader about conservatism that existed and said nothing of the optimism beside it. All four are fixed. **The span is the years that elapsed**, not the count of profitable ones: a loss year between two good ones used to compress the time a business had to compound, so cash flow of 100, −50, −50, 121 read as 21%/yr where three years had actually passed and the rate is 6.6%. **A shrinking business is projected to shrink**: growth was floored at zero, so a company whose cash flow fell every year was valued as flat forever — 972 of the 2,753 names with two or more years of history were shrinking. (Damodaran, *Valuing Distressed and Declining Companies*, 2009: the perpetual growth rate "in some cases ... can even be negative ... the firm will continue to exist but get progressively smaller over time". The −10% bound itself is our judgement, not his; he prescribes none.) **The base is what the company reported**, loss years included, rather than an average of only the good ones. And **the growth blend may move down**: historical and analyst-implied growth are blended 40/60, where the analyst's view used to be kept only when it was the more optimistic of the two. **Where a rate cannot be computed at all** — fewer than two profitable years, so there are not two points to compound between — the model now declines to value the row rather than substituting zero. A company that merely *ends* in a loss is not that case: it still has a measured rate between its last two profitable years, and a negative one is now a permitted answer. Re-run

over the 2,891-name scan of 2026-09-23 these changes moved 542 fair values, withdrew 133, and changed 102 of 2,577 buy/sell verdicts.

The discount-rate floor. The cost of capital is floored at 7.0%, which is not a chosen number: it is the 2.5% terminal growth plus a 4.5-point spread, so the Gordon denominator can never narrow to the point where far-future cash dominates the appraisal. Since 2026-09-23 **the same 4.5-point spread guards every other model that divides by (cost of equity – growth)** — the dividend-discount model, residual income, the justified book-value multiples and the REIT perpetuity — and no model may project growth above the risk-free rate, following Damodaran's rule that a stable growth rate should not exceed the riskless rate used in the valuation, because nothing outgrows the economy forever. Before that, growth caps written in one module (6% for a bank's retention × ROE) and the discount floor written in another could sit a single point apart, and a bank's justified multiple read up to 39× tangible book. **What that cap does, in both directions.** It is usually described as pulling inflated appraisals down, and mostly it does — re-run over the 2,891-row scan of 2026-09-23, it lowered the fair value on **553 of the 2,089** names valued both ways (median –12.5%). But the justified-multiple relation (ROE – g) / (k – g) is *decreasing* in growth whenever a company earns less than its cost of equity, so lowering g RAISES the multiple for exactly the weakest businesses: it lifted **103 of 2,089** (median +11.3%, largest +247%). A company earning 6% on 7%-cost equity moves from 0.50× book to 0.78×. Across the scan the change moved **107 of 2,577** buy/sell verdicts — 62 from ADD to HOLD, 24 HOLD to TRIM, 17 HOLD to ADD and 4 TRIM to HOLD — and that count is published here because a change that moves advice should be counted in the units advice is given in, not only in percentage moves of the estimate. It was a flat 8.5% until 2026-09-18, which stopped guarding the tail once the premium beneath it fell — it was clamping 29.6% of priced rows, a third of them losing more than the whole 100 bp quality adjustment. Generated from `valuation/discount.py` and `valuation/models/dcf.py` by `scripts/render_factor_sets.py`.

> **IN PLAIN WORDS**
>
> Think of a report card with five subjects. In each subject the company is graded on a curve *against its classmates* (companies in its sector and size), not against an impossible universal standard. Add the five subject grades and you get one score out of 100. A 72 doesn't mean "72% good" — it means "this business ranks near the top third of its peers on quality-and-value, today."

How the five grades are actually computed

Each pillar grade is built in two moves. First, inside a peer group, every factor is turned into the robust-z score from Chapter 3, and a weighted average of those z-scores gives the company's

raw standing in that pillar. Second — and this is the clever part — that raw standing is turned into a **percentile rank against the same peer group** and multiplied by the pillar's points.

UNDER THE HOOD · TWO-PHASE PILLAR SCORING

```
# Phase 1 — raw standing in a pillar (present factors only) pillar_mz =
Σ( weight · robust_z ) / Σ( weight ) # Phase 2 — rank that standing
across the peer cohort, scale to points pillar_score = PILLAR_MAX ×
percentile_rank( pillar_mz within cohort ) # Phase 3 — rank the
COMPOSITE across the whole day's universe cohort_composite = Σ
pillar_score deep_score = clamp( round( 100 × percentile_rank(
cohort_composite across the universe ) ) , 0 , 100 ) # A sum of five
uniform pillar ranks is BELL-SHAPED, not uniform, so the fixed # label
cliffs landed nowhere near the fractions they name (STRONG BUY caught #
0.6% of a 1,015-name scan where the spacing implies 15%). Ranking the
composite # makes a cliff mean what it says — and means the headline is
NOT the sum of the # bars: the bars are your standing in your cohort,
the headline across everyone. # A cohort with <5 rankable peers on a
pillar → neutral midpoint for that pillar, # applied identically to
every member, because it is a property of the COHORT. # A pillar THIS
company cannot fill is different: the row is left UNSCORED # entirely,
never given half the points free (see "when the model declines").
```

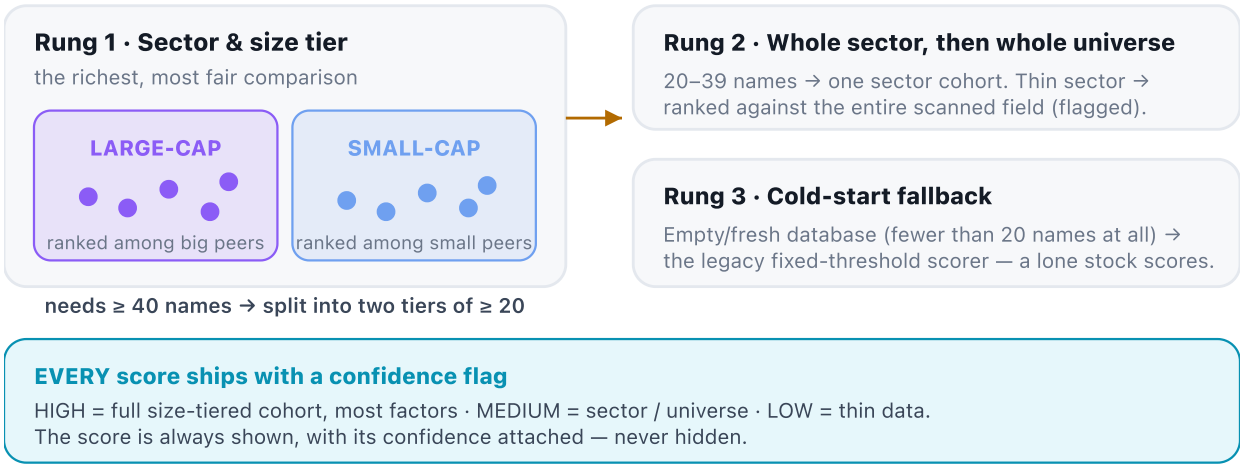
Why rank the pillars instead of mapping z-scores straight to points? A fixed "z → score" line sounds simpler, but it made the top of the scale unreachable: on a real 1,000-stock run, *zero* companies landed in STRONG BUY. Percentile ranking is **self-calibrating** — it always spreads the day's field across the full 0–100 range, so the labels downstream tools rely on stay meaningful. `deep_score_engine.py — _score_group, _raw_pillar_mz`

The peer group: sector, then size

"Companies like it" is defined in two layers. First by **sector** (a sector needs at least 20 scored names to form a fair cohort). Then, if a sector is big enough (40+ names with a valid market cap), it is split into **large-cap and small-cap tiers** using a robust median of *log* market cap — so a company is ranked against peers of similar size too. If a sector is too thin, the model falls back gracefully rather than giving up. **That fallback has an exception worth stating plainly:** a business class with too few scoreable names to form its own cohort is ranked on the *universe* rung, and there every name is scored on the DEFAULT operating-company factor set — not on its

class's own factors. So "a class is ranked only against its own kind" holds wherever the class is large enough to be a cohort, and where it is not, the row is measured on the general set rather than withheld. The alternative — carrying class-specific units such as FFO or a loss ratio into a general pool — is the mixing the peer group exists to prevent, which is why the basis changes with the rung. `deep_score_engine.py — score_cohort, thin_indices`

FIGURE 5 · PEER-COHORT FORMATION & THE 3-RUNG FALLBACK (DEEP SCORE ABSTAINS ONLY WHEN THE COMPANY CANNOT BE MEASURED)



Unlike Winner Odds (which abstains on thin data), Deep Score **never trades availability for confidence**: a thin peer group widens the cohort and lowers the `deep_confidence` flag rather than withholding the score. It withholds only where there is nothing to measure, in two cases, and in both the name is shown unscored (N/A) and left out of its peers' cohort rather than placed at the midpoint. **No financial history at all** — no usable annual statement (2026-09-17: 108 of the 1,922 names the universe expansion added, pre-revenue filers whose years the reader had discarded; they had been scoring a median 56 on nothing). And **a pillar the company cannot fill** — the score is these five pillars, and one of them handed over at the neutral midpoint is half its points awarded free: on the same universe that moved 104 names, 89 of them downward by a median of 4 points, while pushing the best-ranked of them up into STRONG BUY on no measurable growth. It costs nothing on established companies: none of the 1,008 hand-curated names is missing a pillar, and the 104 are clinical-stage biotechs and pre-production miners. `Source: deep_score_engine.py — score_cohort, _size_split, _confidence, legacy_fallback`

From score to label

The final 0–100 becomes a plain verdict at fixed cut-offs. These exact cliffs are hard-wired because other tools (position sizing, the Discovery approval gate) depend on them.

FIGURE 6 · THE LABEL LADDER

Cut-offs at 85 / 70 / 55 / 40. `Source: deep_score_model.py — LABEL_CLIFFS`

HOW IT CONFORMS · METHODOLOGICAL STANDARD

Deep Score follows accepted **factor-investing and robust-statistics practice**: economically-motivated factors, a transparent weighting, peer-relative (sector- and size-neutral) standardisation, and outlier control. It makes no return forecast and no personalised recommendation, so it raises no advice-registration question — it is a research screen, presented with its own confidence disclosure.

THE HONEST LIMIT

Deep Score is an **assessment of a company's standing today, not a prediction of its price**. It reads trailing levels, so a genuinely improving business can look mediocre until the averages catch up (that gap is exactly what the Inflection engine in Chapter 8 is for). And until 6 October 2026 it had **not been tested against forward returns at all**. This page once said such a test was impossible; the point-in-time population harness built on 27 September 2026 (every US 10-K filer FY2009–2022, scored from what each had filed at the time — the test Winner Odds went through in Chapter 7) made it possible, and it has now been run. What it found, and what it did not test, is in the next paragraph, and every screen that shows a STRONG BUY / BUY / AVOID label or a sizing band carries the short version: the fundamentals were tested and ranked later returns modestly; the valuation inputs, the label cut-offs, Canadian listings and the sizing bands were not. The product still never calls Deep Score “back-tested,” “validated” or predictive.

The population test (6 October 2026). Registered in advance — the design was committed before any run read a return — and judged by the same certification rule Winner Odds passes. Every US 10-K filer FY2009–2022 was scored by the shipped engine as of each fiscal year, from what it had filed by then; its outcome is the three-year return from the first trading day after its annual report (a bankruptcy inside the window counts as –100%) *minus the average of its own peer group and size tier that year*, the quantity the score claims to rank. Size comes from the public float each company states on its own annual report, the one market value in the record a later share split cannot distort. Of 65,318 company-years, 40,916 could be scored and 23,694 graded against their peers (the rest had no price path, were acquired or went dark with no price, or traded under $1). **The fundamentals core — the four business pillars, 85 of the 100 points, valuation left out — passed**: a rank correlation with the peer-relative return of +0.10, the top third beating the bottom third by 6.7 percentage points over three years, positive in both halves of the history (+0.07 for 2009–2014, +0.11 for 2018–2022), a t-statistic of 3.10 against a bar of 3.0 and a deflated Sharpe ratio of 0.99. It is a **narrow pass and a modest effect**: two of the fourteen years ran the wrong way (2011 and 2022), and over three years the names the 85-point core would have labelled STRONG BUY beat their peers by 3.1 points on average while those it would have labelled AVOID trailed by 2.8 — descriptive figures for the four-pillar rank, not a test of the shipped five-pillar labels or their cut-offs. The pre-registration did not say how names tied on the whole-number score are split at the tercile edges; the recorded verdict uses the order the history stores them in, and with t that close to the bar the pass is narrow. The five-pillar version, with a valuation pillar rebuilt from the

public float, did **not** pass (t 2.89). Not tested at all: the valuation inputs the live score uses (the vendor's P/E, PEG and free-cash-flow yield — PEG needs an analyst forecast that has no point-in-time record), the exact label cut-offs, Canadian listings (they are not in the SEC record), the live vendor-sector peer groups (an SIC map stands in, and it leaves 12.3% of the filers — 13.0% of the company-years — in a catch-all Industrials peer group, among them biological-research labs, coal miners and education companies; remapping them would be a new trial for all three engines, so the share is disclosed instead), the per-share growth factors, the sizing bands, and any probability. The test's own figures — the funnel, every certification metric, the year-by-year spreads and the label averages — are written by scripts/backtest_deep_score_population.py to scripts/backtest_deep_score_result.json; the model register (Deep Score 2.1.3) carries them with the data-coverage counts measured alongside.

One company, one verdict. A business listed on two exchanges, or with two share classes, used to enter its peer group twice and — because an American ticker is read from the regulator's filings and a Canadian one from the vendor's frames — could arrive with two different histories and publish two different answers for the same company. The engine now groups listings by company name, lets the listing with the deeper history vote, and gives every other listing that same verdict, recording which ticker it came from. Both tickers stay scanned, searchable and readable on their own page, so a Canadian holder still sees the symbol they own; the ranked screens show the business once, because two rows carrying one answer would spend two places on a shortlist that is meant to name different ideas.

Two limits worth naming. An auto or equipment maker with a captive finance arm (GM Financial, Ford Credit, Cat Financial, John Deere Financial) carries that lender's debt on its consolidated balance sheet, and no feed line separates it, so its leverage is scored as filed — the Street looks through to the industrial company; the register discloses this rather than adjusting it. And insurers are scored on *financial* leverage — debt over debt plus shareholders' equity, the rating-agency measure — never on assets over equity, which would count policyholder liabilities as leverage and rank every life insurer last by construction; banks, whose balance-sheet leverage is their risk, keep assets over tangible equity.

The Growth pillar decides the top of the ranking, and it is built entirely on the past. Both of its inputs — revenue growth and earnings growth — are historical, and 25 of the composite's 100 points rest on them. Measured on the production scan of 24 September 2026, over 2,577 scored names: the rank correlation between the published score and the same score with the pillar removed is 0.97, which reads as harmless —

but 45 of the top 50 names are in that set *only* because of the pillar, and 6 of the 8 names the live screen surfaced were lifted by it, averaging 20.6 of its 25 points against a universe mean of 12.5. Two findings bear on that directly. Chan, Karceski and Lakonishok (*Journal of Finance*, 2003) find no persistence in long-term earnings growth beyond chance. Lakonishok, Shleifer and Vishny (*Journal of Finance*, 1994) find that buying low forecast growth and selling high forecast growth earns superior returns over five years — the glamour end underperforms.

The pillar is therefore **disclosed rather than re-weighted**, and the reason is that whether its sign is wrong has **not yet been tested on its own**. The population test described above has now run (6 October 2026): the four business pillars *together* ranked later peer-relative returns and passed the certification bar narrowly, and the Growth pillar is 25 of those 85 points. It did not measure the pillar by itself, and a per-pillar read on the same history would be a new trial, counted as one. The forward grading of the 1,059 names scored on 8 June 2026 ran on 14 October 2026, two days after its 126-day window closed, under a reading rule fixed before any return was read. Over the 998 names it could grade, the pillar's points showed **no detectable sign either way** against each name's return relative to its own peer group and size tier: rank correlation +0.04, with a 95% interval of −0.02 to +0.11, and taking the pillar out of the composite would not have changed how well its ranking matched those returns by a measurable amount. Fourteen names could not be priced, and had they fallen in line with their Growth scores the reading would have tipped to a positive sign. That is one window in one market regime — a diagnostic, not the test — and it changes no weight. Until the pillar is tested on its own, changing the weights on the strength of the literature alone would replace weights the population test measured together with weights nothing has measured. Instead, whenever a recommended name owes the pillar more than an evenly-scoring name would (above its own weight share of 25%), the report says so in the recommendation itself, names both papers, and calls the rank a screen rather than a forecast.

Grading the live labels forward (from late October 2026). Separately from that historical test, the owner's validation report grades the labels the live screen actually published, scan by scan, **126 and 252 calendar days later** (about four and eight months): each name's total return between its own stored dividend- and split-adjusted closes, and each stored fair value against the share price carried forward by the split-only closes. Until 15 October 2026 it read the next scan's raw quote, so a split inside the window graded as a crash and a dividend was left out. Every name it cannot grade is **named with the reason** — no stored close at the scan; stored closes that end before maturity (not refreshed since, or no longer trading); none in the days from maturity; no

split-only close — and the next cohort’s ungradable names are listed before its grading date. Each band’s graded count is printed beside the names it labelled, so a median over a few survivors is not read as the band’s. The first cohort, the 15 June 2026 scan, matures on 19 October 2026 and is graded once a close dated on or after 24 October — the last day its maturity close may fall on — is stored, so no name is dropped for a close still to come, and a name whose own stored closes stop earlier is named as stale rather than as missing its maturity close. A few months of one market is a report card, not a test of the score, and it is measurement only: it never adjusts a weight.

ETFs are scored by a separate ladder — cost, long-term performance, fund size, income — and performance is judged within the fund's asset class (equity, fixed income, cash, balanced, commodity), inferred from the vendor category or, for Canadian listings that carry none, from the fund's name. A bond or cash fund is never asked for an equity fund's 15% a year, and a fund too young to have a five- or ten-year return has that window scored as unknown, with the score re-based on the pillars it has, rather than as zero. A fund whose expense ratio is not on file is the one deliberate exception: its cost pillar scores as the worst fee, not as unknown, so a gap in the data can never rank it above a fund that discloses what it charges (the breakdown says the fee is not on file).

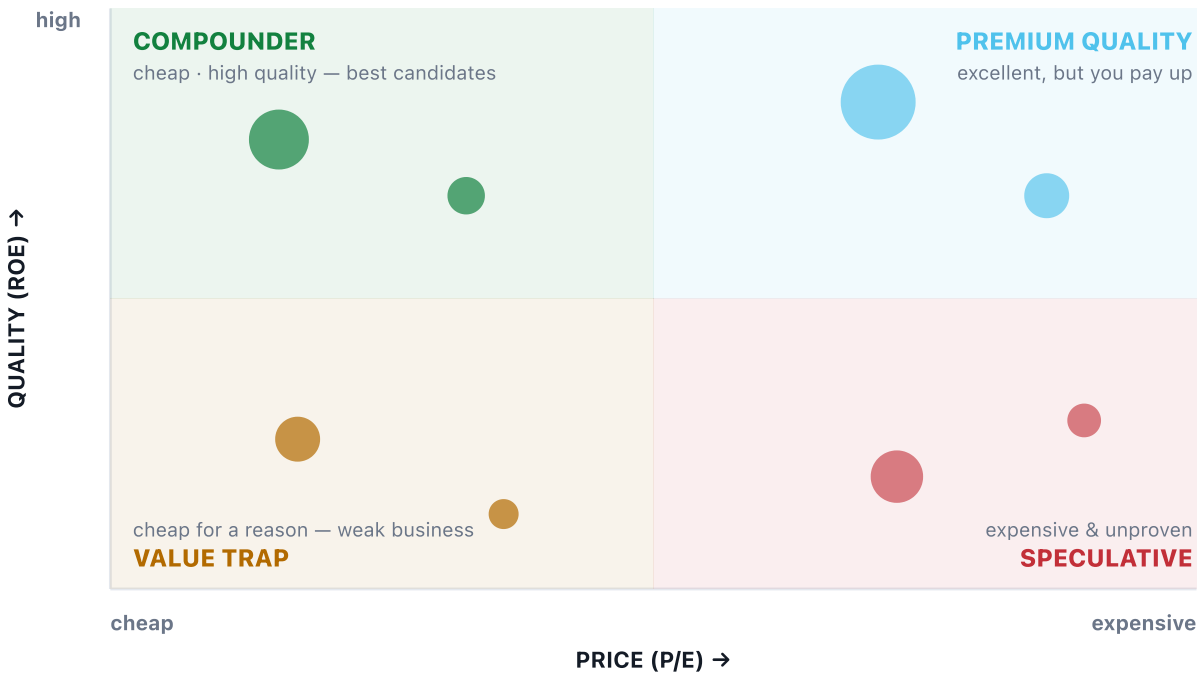
Chapter 5

The Value-vs-Quality Map — the four quadrants

A single picture that separates "cheap because it's good value" from "cheap because it's broken."

The Screener's signature chart plots every company on two axes you can read at a glance — how **expensive** it is (left = cheap) and how **high-quality** it is (up = better) — with bubble size showing company size. But the *colour* of each dot is smarter than its position: it is decided **relative to the company's own sector**, so a stock sitting in the visually "expensive" zone can still be coloured a green Compounder if it's cheap and excellent *for its industry*.

FIGURE 7 · THE VALUE-VS-QUALITY MAP



Bubble size = market cap. The axes show raw price and quality; the **colour is sector-relative**, which is why the coloured regions and the visual position can disagree — and why there are deliberately no universal gridlines drawn on the chart. `Sources: deep_score_engine.py — compute_quadrant · web/components/charts/PlotlyQuadrantImpl.tsx`

UNDER THE HOOD · HOW THE COLOUR (QUADRANT) IS DECIDED

```
# uses the COHORT-RELATIVE z-scores, not raw P/E and ROE val_zs = every
valuation lens THIS CLASS scores that the name has # default: P/E, FCF
yield, PEG · REIT: FFO yield, dividend yield, ... if val_zs is empty: #
no lens at all → cheapness unknowable quadrant = SPECULATIVE else:
quality_high = (quality_z > 0.5) and (deep_score ≥ 55) # quality_z is
the CLASS's own lens: ROE, or FFO-ROE for a # REIT, or through-cycle
margin for a commodity producer cheap = mean(val_zs) > 0 # cheaper than
the cohort on the blend Compounder = quality_high and cheap Premium
Quality = quality_high and not cheap Value Trap = not quality_high and
cheap Speculative = not quality_high and not cheap
```

Two deliberate choices: quality must clear a bar *above* the median (`roe_z > 0.5` , roughly the top third) **and** the overall composite must be at least HOLD-grade, so a bottom-of-the-class stock can never masquerade as a "Compounder." And a company with no real earnings is judged *Speculative* outright, rather than being flattered as "infinitely cheap." `deep_score_engine.py — compute_quadrant`

IN PLAIN WORDS

Two questions, four corners. Is it *good* (up) and is it *cheap* (left)? Good-and-cheap is a Compounder — the sweet spot. Good-but-pricey is Premium Quality — you're paying for the quality. Cheap-but-weak is a Value Trap — cheap for a reason. Pricey-and-weak is Speculative. Judging "good" and "cheap" against the company's own industry is what stops you from calling every utility "low growth" or every software firm "expensive."

THE HONEST LIMIT

The map is a *starting point for research*, not a buy list. A green Compounder can still be a mistake (the model can't see a pending lawsuit or a fading product), and a Premium-Quality name can be worth every penny. The quadrant tells you *which question to ask next*, not what to do.

Chapter 6

Winner Odds — the durable-compounder ranker

A ranked shortlist of the companies whose accounts look most like businesses that kept compounding for years, and least like the ones that failed. Not a multi-bagger finder — tested, and it is not one (see the closure at the end of this section).

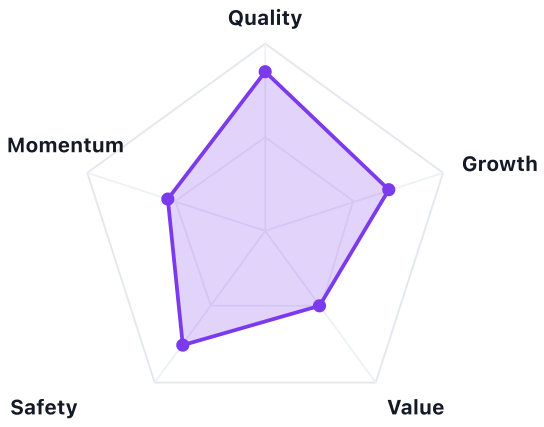
Where Deep Score grades *every* company on general quality-and-value, Winner Odds is pickier: it looks for the durable, high-return, still-growing businesses that keep compounding and rarely fail. The population test (closure below) shows that is what it finds — and that it does not find multi-baggers: top-tier names rose fourfold in five years no more often than anyone else. It scores each name on seven core factors drawn straight from the "great-investor canon" — Buffett & Munger and Greenblatt on returns on capital, Novy-Marx on gross profitability, Fisher and Lynch on growth runway — and produces a **percentile rank and a tier** (Strong candidate → Watch → Unlikely; three tiers since 2026-09-28, Strong candidate being the top quartile). It deliberately keeps *price* and *momentum* to the side as an "you decide" overlay, never mixed into the quality-and-growth rank.

Role	Factor	What it captures (and whose idea)	Wt
Core · Quality	ROIC	High returns on all capital (Buffett–Greenblatt)	0.20
	Gross profitability	Gross profit ÷ total assets, same fiscal year (gross profit is read as filed, or derived as revenue – cost of revenue where the filer tags the cost line and no gross-profit line — 810 of 2,583 US names on the 2026-09-27 scan had none until 2026-09-28) — the cleanest quality signal (Novy-Marx 2013, as published; until 2026-09-27 this computed gross *margin*)	0.13
	Piotroski F	All nine published signals, asset-scaled (until 2026-09-27: seven, on equity) A filer with no borrowing anywhere in its history scores zero on the leverage signal rather than losing the whole score (Piotroski's rule: one if leverage fell, zero otherwise; 2026-09-28). Two borrowing labels (convertible notes payable, unsecured long-term debt) were missing from the list that separates “no debt” from “debt under a label not read”, so eight names on the 2026-09-28 book, ServiceNow and Zscaler among them, scored as debt-free; they are now unknown on this signal. Reading such labels as the debt figure was tested the same day and *declined*: it lowered the population test (rank-IC 0.244 to 0.238, deflated Sharpe below the bar).	0.20
	Margin stability	How steady the net margin has been over the last five years — lower spread is better (house definition; until 2026-09-27 this was named "durability" and rewarded margin *expansion*, i.e. a cycle peak)	0.30
	Low accruals	(Net income – operating cash flow) ÷ average assets (Sloan; until 2026-09-27 scaled by |net income| and charged capex)	0 — shown, unweighted

Role	Factor	What it captures (and whose idea)	Wt
Core · Growth	Growth acceleration	Growth that is speeding up (reused from Inflection)	0 — shown, unweighted
	Rule of 40	Growth% + margin% ≥ 40 (healthy growth)	0.17
Gate	Altman Z	Bankruptcy-distress guard — a hard cap, never summed	—
Overlay	Earnings yield (1/P/E) · FCF yield · PEG · upside to fair value · 6-month price return · insider net buying — an unknown input is shown as unknown, never as the cheapest name	Value & momentum — shown *beside* the rank, never inside it	—

Where the weights come from (2026-09-28). Until this date the weights were hand-set priors that nobody had tested. Each core factor was then measured on the SEC 10-K population test: its rank information coefficient against three-year forward returns, per independent three-year block, FY2009–2022. Five factors carry signal well past the t = 3 bar the multiple-testing literature demands (margin stability +0.25, ROIC +0.17, Piotroski +0.17, Rule of 40 +0.14, gross profitability +0.11); the weights are those coefficients, normalised, which is Grinold's rule for combining signals. Growth acceleration carries none (t = 0.4) and Sloan accruals run *against* their published direction here (t = −8.9, consistent with the literature on the anomaly's decay), so both stay on the scorecard and in the five-of-seven presence floor at weight zero — measured and shown, never flipped, never fitted. The scheme was read on blocks never used to fit it, both ways round: rank-IC 0.162 → 0.196 on the recent half and 0.140 → 0.173 on the early half against the old priors, with top-decile bankruptcy at 0.5% against 3.2% for the rest. Every configuration tried is a recorded trial, so the Deflated Sharpe below counts them. The share of names that go on to gain 150% or more did not move under any scheme: this ranker finds compounders that survive, not lottery tickets.

FIGURE 8 · A WINNER ODDS SCORECARD (RADAR) AND THE RANK-TIER LADDER



50 = the universe median on each spoke

PERCENTILE → TIER

Unlikely	< 50th
Watch	≥ 50th
Strong candidate	≥ 75th (top quartile)

Three tiers since 2026-09-28: the 90th-percentile cut could n

HARD GATES override the tier
Distress → Unlikely · Cyclical peak → Watch

Each spoke is a pillar score, `clip(50 + 25·mean-z, 0, 100)`, so 50 is the peer-group median (the name's sector, or its business class). The tier comes from the company's percentile in the composite — unless a hard gate (distress or a commodity-cycle peak) caps it. `Sources:`
`winner_engine.py — _score_row, _pillar_scorecard, score_universe · winner_model.py — rank_tier`

UNDER THE HOOD

```
robust_z = z of the factor within the name's peer group # its sector,
or its business class — the Deep Score's cohort; under 20 names → the
universe composite = Σcore( weight · robust_z ) / Σcore weight # present
core factors only not ranked if the class is bank / insurer / REIT /
utility / fund # graded by the Deep Score's class set instead one
listing per company votes; its other listings copy the result # GOOG
and GOOGL agree and take one slot abstain if fewer than 5 of the 7 core
factors are present # thin data → no rank at all abstain if the
universe has < 20 names # no real field to rank against percentile =
count( composites ≤ this ) / n tier = rank_tier(percentile) # while
CALIBRATED is False; ranked ACROSS the universe if |percentile − cut| <
0.03: tier = last scan's tier # boundary buffer, adjacent tiers only if
altman_z < 1.10: tier = "Unlikely" # Altman Z″ distress zone; skipped
for banks, insurers, REITs, utilities, funds elif altman_z is None:
flag "solvency not assessed" # an input is missing: the gate did not
judge the name; the tier stands, the badge says so if cyclical_peak:
tier = min(tier, "Watch") # timing risk
```

Value and momentum factors are computed but held *outside* the composite as overlays. The design view is that "is it cheap?" and "is it in favour?" are the owner's calls — mixing them into a quality rank would contaminate a clean signal with a timing bet.

`winner_model.py — CORE_FACTORS, OVERLAY_FACTORS, GATE_FACTORS`

IN PLAIN WORDS

Picture a talent scout's shortlist for "most likely to become a superstar." The scout only grades on the traits that actually predict superstars — durable high returns, real cash profits, a long runway to keep growing — and refuses to grade anyone with too little track record (fewer than six readable traits, or too few peers to compare against). Two automatic red flags — looming financial distress, or a commodity company at the top of its cycle — knock a name down no matter how good the raw score looks.

HOW IT CONFORMS · METHODOLOGICAL STANDARD

Winner Odds is a transparent multi-factor cross-sectional model with published factors, each weighted by its measured information coefficient on the population test, and robust standardisation — squarely inside accepted quantitative-equity practice (and the CFA curriculum's treatment of factor models). Because it emits a rank and a qualitative tier rather than a probability or a personalised recommendation, it makes no forecast it cannot support — the subject of the next chapter.

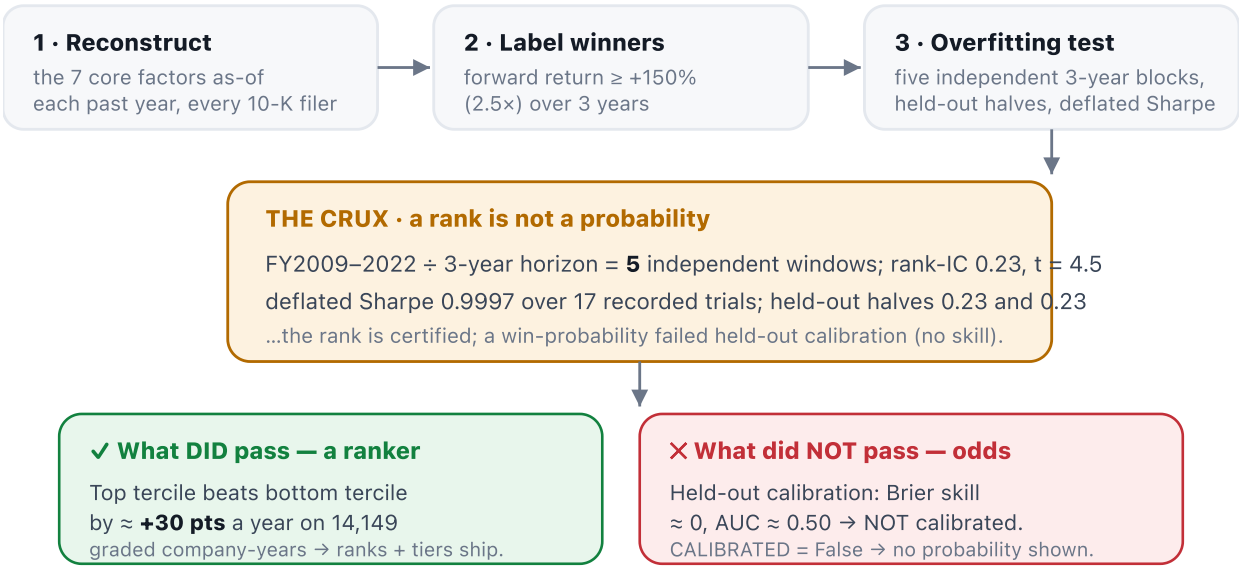
Chapter 7

Why Winner Odds ships rank-only — the honesty gate

The single most important integrity decision in the product: the engine refuses to print a win-probability it cannot honestly certify.

It would be easy — and dishonest — to slap a "68% chance of winning" on each name. CompoundWise won't. A probability is only trustworthy if it has passed a strict statistical certification on *independent, survivorship-free* data, and the available history simply isn't deep enough to pass that bar yet. So the engine ships a **rank and a tier**, and a flag in the code, `CALIBRATED = False`, guarantees no probability number is ever emitted. In the authors' words: "a calibrated-false model that emits a probability is a bug."

FIGURE 9 · THE CERTIFICATION GATE — WHY ODDS DON'T SHIP (YET)



The first test was a 38-name basket chosen with hindsight: in-sample, survivorship-biased, one window, and it separated the best- from the worst-ranked names. The test that now decides is the population one described below: every US 10-K filer, FY2009–2022, point in time, bankruptcies and delistings from each filer's own SEC record, five independent three-year blocks, every configuration recorded. It validates the RANK on every gate but the base rate; it does not fit a probability, and none is shown.

UNDER THE HOOD · THE CERTIFICATION BAR

To flip `CALIBRATED` to true, the backtest must clear *all* of: at least **5 independent (non-overlapping) windows**; a t-statistic **≥ 3.0** (Harvey–Liu–Zhu); a **Deflated Sharpe ≥ 0.95** (Bailey–López de Prado) whose trial count is read from a recorded trial log of every configuration ever run; an **external base rate** pinned to a source (none is today — the 12% the code carried until 2026-09-27 cited "breadth studies" that were never pinned); and an in-sample rank-IC of at least 0.05; a rank-IC of at least 0.05 on *each* half of the history when the weighting is fitted on the other three blocks and read on the two it never saw, both ways round (Bailey–López de Prado: a result must hold out of sample, and a certification with no held-out measurement on file is refused rather than assumed); and a positive tercile spread. Until 2026-09-28 the in-sample IC also had to sit *below* 0.15 on the premise that a bigger number signals overfitting; that premise came from monthly-horizon studies, no source was found for a three-year horizon, and survivorship in this population runs the other way (the unpriced names concentrate in the bottom tiers), so the owner retired the ceiling in favour of the held-out test. Only then are a logistic slope and an intercept fitted — the intercept is pinned so the average probability equals the adopted **base rate** (8.3%, adopted 2026-09-28: the share of graded company-years on the population test that gained 150% or more in three years, bounded 6.8–24.5% by the exits still unpriced; a measured house figure, since no external study states such a rate), never fitted to flatter the test basket.

`scripts/backtest_winner.py — certification gate`

THE FIRST POPULATION TEST (2026-09-27) — AND WHY IT DID NOT CERTIFY

Beyond the 38-name basket, the ranker was run over **every US 10-K filer** in the SEC's free Financial Statement Data Sets (2009 onward), each company scored as of each fiscal year from what it had filed by then, at the earliest filing, with the entry priced after the 10-K and a three-year horizon — by the same engine that ships. Of 64,835 company-years, 17,962 could be ranked and 10,346 graded (9,812 priced survivors plus 534 bankruptcies counted at –100%). The rank-IC was 0.158 and the top tercile beat the bottom by 16 points a year over five independent three-year blocks (t = 3.5). It is **not certified**: the IC sits above the plausibility band the gate demands, no external base rate is established, and — the finding that matters most to a reader — **the published tiers are not monotone**: "Plausible" names (+58%) beat "Strong candidate" names (+44%), which sat beside "Watch" (+44%). Two of thirteen years ran negative. The survivorship claim is partial: 6,135 ranked names (34%) had neither a price path nor an exit event and are excluded and counted, so every figure above is an upper bound. The empirical base rate of a +150% three-year outcome was 8.9–10.3% of graded names (5.5% if every unpriced name is a loser). Source: `scripts/backtest_winner_population.py`, `scripts/history_*.py`.

Follow-up (2026-09-27, later the same day). Two things were checked before anything was changed. First, is the Plausible-over-Strong gap real? Across the five independent three-year blocks Strong beat Plausible in one and trailed in four; the mean block difference is –0.08 with t = –1.2, and a bootstrap interval on the pooled gap spans zero. The *medians* are monotone (Strong 0.32, Plausible 0.31, Watch 0.28, Unlikely 0.09); the mean gap comes from a right tail of small-cap Plausible names (+1.01 versus +0.36 for small-cap Strong), while large caps run the expected way. Bankruptcy within three years falls from 4.7% in Unlikely to 1.5% in Strong, and from 7.3% in the bottom decile to 1.5% in the top. The share of names reaching +150% is flat at 8–10% in every tier: the ranker separates average outcomes and failures, not the right tail. The tiers were therefore **left as they are**; moving the cuts to fit one uncertified test would be the tuning the Deflated Sharpe exists to penalise. Second, the missing third. Every filer's own SEC submissions feed was read (13,950 filers, all forms, all 8-K item codes), replacing a full-text harvest that had found 700 delistings in 17 years and none for 2013–2016. A Form 25 alone is not an exit (a SPAC delists its units on the merger and keeps filing), so an exit requires the filer to have stopped filing annual reports. With that record the unpriced share fell from 34% to 21% of ranked names: 1,556 were acquired and 978 went dark, both carried as bounded buckets because no deal price is parsed; the graded statistics did not move (rank-IC 0.157, tercile spread +0.157, t = 3.5). Still **not certified**, for the same two reasons. One defect in the test

itself is disclosed rather than fixed: the reconstructed filings carry no Altman Z″ (two balance-sheet tags were never harvested), so the distress gate that ships was inert in this test — every name ranked as if solvency were unknown. **Closed on 2026-09-28:** the two tags were backfilled, the gate now judges 43,545 of 79,582 reconstructed filings, and with the measured weights described under the factor table the full-history rank-IC is 0.191, the tiers are monotone in mean for the first time (Strong +0.49, Plausible +0.47, Watch +0.42, Unlikely +0.41) and the long/short Sharpe over the five blocks is 2.85 with a Deflated Sharpe of 0.998 over six recorded trials. Still not certified: the IC sits above the house plausibility band, whose premise for a three-year horizon is itself unsourced, and no external base rate exists. Later the same day gross profit was derived as revenue – cost of revenue for filers who never tag a gross-profit line: 14% more company-years could be ranked (20,521), the rank-IC read 0.197, the tiers stayed monotone (Strong +0.52, Plausible +0.47, Watch +0.41, Unlikely +0.39) and the long/short Sharpe over the five blocks was 4.48. Reading a filer with no borrowing on its balance sheet as carrying zero long-term debt, at the filing, then ranked a further 4,421 company-years and taught the test to refuse a sub-penny quote as an entry price (482 excluded): on that final population the rank-IC is 0.265, the top tercile beats the bottom by 28 points a year, the long/short Sharpe is 1.87 over the five blocks (t = 4.2) with a Deflated Sharpe of 0.976 over eleven recorded trials, and the Strong and Plausible tiers sat level with each other by median rather than in order. **Three decisions followed on 2026-09-28.** The 90th-percentile cut was removed, so "Strong candidate" is the top quartile and the tiers are three. The certification rule stopped failing a rank-IC above 0.15 and instead demands the held-out test described above, which the model passes (0.277 and 0.234). And acquired or gone-dark names are now graded at their last print from a feed that keeps delisted symbols (838 company-years, reachable only for windows after mid-2020): rank-IC 0.267, the top tercile 30 points a year above the bottom, long/short Sharpe 2.08 over the five blocks, Deflated Sharpe 0.974 over thirteen recorded trials, tiers in order. With the measured base rate adopted (8.3%, bounded 6.8–24.5%), **every gate of the rule holds: the ranking is certified.** A probability is a separate question and it failed: fitted on three blocks and judged on the two it never saw, a win-probability curve had no skill over the base rate on either half (Brier skill ≈ 0, AUC ≈ 0.50) — the share of names reaching +150% is flat across the ranking, 6.8% to 9.3% by decile. So the product claim is exactly this: Winner Odds orders companies by durable quality and steers away from failures, and it does not tell you which will become multi-baggers. No odds are shown.

IN PLAIN WORDS

You can rank ten horses confidently from a few races, but you can't honestly quote each one's win *percentage* until you've seen many independent seasons. CompoundWise has enough evidence to rank, and openly says so — and not enough to quote odds, and openly says *that* too. A separate AI feature that writes a "bull vs bear" note has a firewall that strips out any win-probability language the model tries to sneak in.

WHY YOU CAN TRUST IT

The refusal is enforced in code, not policy: serializers and the screener null out the odds field whenever `CALIBRATED` is false, and the AI thesis-writer runs a regex scrub that deletes probability phrasing. Meanwhile a monthly, survivorship-free **forward log** quietly accumulates real out-of-sample outcomes — the honest, slow path toward one day earning the right to show odds. `winner_thesis.py · scanner.py — snapshot_winner_predictions`

THE HONEST LIMIT

The **+64-point** tercile spread is a *discrimination* result on a hand-picked, survivorship-biased basket — proof the factor *ordering* is real, read directionally only. It is **not** evidence of out-of-sample alpha, and the product never presents it as such.

WHAT IT CAN AND CANNOT DO — TESTED ON THE WHOLE POPULATION (CLOSURE 2026-09-28)

In plain words. Every US company that filed an annual report from 2009 to 2022 — about 14,000 companies, *including* the ones that later went bankrupt or were delisted — was scored using only what it had filed at the time, and its shares were followed for five years (figures measured on the 2026-09-28 history, before the 2026-10-06 repair, and not yet re-derived). The top quartile's typical name gained **+55%**, with 6 in 100 losing 70% or more and 2 in 100 going bankrupt. The bottom quartile's typical name **lost 54%**, with 44 in 100 losing 70% or more and 22 in 100 going bankrupt. Everyone together: +30%, 18 in 100, 9 in 100. That separation is stable on years the model never saw, and it is the tool's strength: it finds the businesses that keep compounding and it avoids the ones that fail.

A correction to the test itself (2026-09-28, register 1.16.0). The historical data behind this test stored only 15 of the 43 borrowing labels that the "no debt" rule checks, so in the test, companies whose debt sat under the other 28 labels were read as debt-free on the leverage check — more generously than the live product reads them. With all 43 stored, the certified figures are lower and still pass every gate: rank correlation with the three-year outcome 0.244 (was 0.267), top-minus-bottom spread +0.244 (was +0.300), long/short t-statistic 4.71, deflated Sharpe 0.966 over 15 recorded trials (bar 0.95), held-out halves 0.250 and 0.219. The five-year figures above are the corrected ones. The probability test still fails, so the product stays rank-only.

Re-run on a repaired history (2026-10-06, register 1.16.1): still certified. Two further defects in the same history were repaired for the Deep Score's own test — the industry map read drug makers as chemical companies and aircraft makers as retailers, and the SEC's data set for the first quarter of 2023 had never loaded, so most fiscal-2022 annual reports carried the next year's filing date — and this test was run again as a recorded trial. Every gate still holds: rank correlation 0.234, top-minus-bottom spread +0.203, long/short t-statistic 4.52, deflated Sharpe 0.9997 over 17 recorded trials, held-out halves 0.228 and 0.226. The five-year figures above were not re-derived on the repaired history.

What it cannot do is find the next ten-bagger. Top-quartile names rose fourfold or more within five years about **7 times in 100** — the same as everyone else. We then tried, under a plan and a pass mark fixed before any result was seen, to build a separate multi-bagger detector from ten point-in-time preconditions: small float, profitability, cheapness, quarterly acceleration (every 10-Q, 2009–2026), operating leverage, no dilution, insider open-market buying (every Form 4, 2006–2026), balance-sheet safety

and a rising count. No configuration met the pass mark on both held-out halves; the best flagged fewer than one name in a hundred and caught under 2% of the eventual four-baggers. The preconditions remove failures (wipe-outs fall from 29% to 8% as more are met) and leave the four-bagger rate flat. NVIDIA is the clean example: from its 2012 report it sat in the Strong candidate or Watch tier every year, because it was a well-run, growing, debt-free business — and so did about 1,400 other top-tier names over those years, most of which simply compounded. The tool saw NVIDIA's quality at the time; nothing in the data could see the size of the run. Register entries 1.12.0–1.14.0 carry every figure and every recorded trial.

Two more things were checked the same day (register 1.15.0). *Would refreshing the rank every quarter help?* No: scoring the same companies at 48 quarter-ends with trailing-twelve-month figures from their 10-Qs gave the same rank quality as the annual reports (rank-IC +0.220 either way, the refreshed version ahead in 26 of 48 quarters), so the rank keeps its annual inputs. *Do commodity names at a price peak get in?* Yes, and they do worse than other top-tier names: over five years a top-quartile mining or energy name had a median gain of +29% with 9 in 100 losing 70% or more, against +62% and 6 in 100 for the top quartile of every other sector — still better than the population, so they stay in the tier, but a reader should know a Strong candidate in those sectors has historically delivered about half the gain.

Chapter 8

Emerging Compounders — the Inflection engine

Catch the business that is bending upward before its trailing averages — and the crowd — notice.

WHICH TOOL, WHEN — AND WHAT THE POPULATION TEST SAYS ABOUT THIS ONE (2026-09-28)

In plain words. **Deep Score** asks “is this a good business today, at a fair price?” — a grade for one stock. **Emerging Compounders** asks “is this business getting better, faster, right now?” — a score built only from trends, indifferent to whether the company is already excellent. **Winners** asks “which companies have the makings of a long holding that will not blow up?” — a certified rank of proven quality, the default long-term shortlist.

The Inflection Score was scored, fundamentals only, on every US 10-K filer FY2009–2022 from what each had filed at the time and followed for five years (the same population and rules as the Winner Odds test). Its year-by-year rank correlation with the five-year return is positive in 10 of 11 years but weak (mean +0.079; Winner Odds on the same rows +0.140). Names labelled Emerging or Inflecting under engine 2.0.0 had a five-year median of +45% with 12 in 100 losing 70% or more, against +36% and 15 in 100 for names with no inflection. The edge is real and modest, and most of it lives in the overlap with quality: an Emerging name inside the Winner top quartile made +71% with 7 in 100 wiped out; an Emerging name outside it made +36% with 15 in 100. Growth acceleration, then the score’s largest ingredient, carried no lift on its own, and insider buying was anti-predictive in every year (names awarded its full points had a five-year median of −20% with 32 in 100 wiped out). **Both weights were changed the same day** (engine version 2.0.0): growth acceleration from 30 to 10 points, insider buying to no points while it stays displayed as a fact. Because most names score zero on those two factors, cutting their weight lifted everyone’s raw score and would have changed what the label bands mean, so the score is re-based onto the published scale so that each label keeps the share of the universe it had before (a recorded house decision; the ordering is unchanged). On the population the new configuration ranks five-year returns better in 10 of 11 years (rank correlation +0.058 to +0.096). Read Emerging as a lens on the Winner shortlist — which quality names are accelerating now — not as a rival ranking, and never as a multi-bagger finder. The earlier walk-forward (scripts/backtest_inflection.py) was on a curated, survivorship-biased basket; this population read supersedes it as the evidence of record.

Measured against the Winner Odds bar, and what passes (2026-09-28, later the same day). Engine 2.0.0 was run through the one certification rule Winner Odds passes, on the same population and grading: it fails every gate (long/short t 1.68 against 3.0 required; deflated Sharpe near zero; held-out halves 0.059 and 0.032 against 0.05 each). A pre-registered program then tried, in a fixed order with every variant counted

against the deflated Sharpe: scoring each fundamental trend as a percentile against the company's sector in that year (V2); replacing growth acceleration with an earnings surprise from the quarterly record (V3); restricting to the Winner top half (V4); measuring the weights by rank correlation on training years and judging on held-out years (V5); and the two shippable candidates with weights measured on all years (V7 with the earnings surprise, V8 without). **V7 certified on that run** (it is not certified on the 2026-10-06 re-run, deflated Sharpe 0.48 — see below): rank-IC 0.169, top-minus-bottom spread +0.248, t 8.12 over five independent windows, deflated Sharpe 0.958 over the eight variants, held-out halves 0.171 and 0.140. V8 misses only the deflated-Sharpe gate (0.949). Two things a reader should know. First, V7 is only moderately related to Winner Odds (rank correlation +0.32 on names both score): its strength is largest on the 10,587 names Winner Odds does not rank at all (rank-IC 0.170 there; its bottom quartile lost a median 23% over three years with 33 in 100 wiped out) and smallest inside the Winner top quartile (0.043), so it complements Winners rather than repeating it. Second, like Winners it works by finding failures more than by finding the tail: over five years its top quartile made +38% with 14 in 100 wiped out, its bottom –31% with 37 in 100, and both quartiles rose fourfold at about the same rate. V7 **shipped as engine 3.0.0** the same day: a cross-sectional pass at scan time (the same shape as the Winner pass) turns each weighted trend into a sector-peer percentile, the earnings surprise is read from the filer's own quarterly facts in EDGAR, the weights are the measured ones in Figure 10, the five untestable or unhelpful signals stay on the page as facts, and the weighted percentile is re-based so every label keeps the share of the book it had that day. The 2026-09-28 figures are V7's; they describe the fundamentals-only score and do not include the displayed facts, which carry no weight. Every run is in scripts/inflection_program.py and the trial record.

The stability signals were tested next, and left out (2026-09-29). The one well-supported idea the data could still test was Mohanram's G-score, the eight signals the accounting literature uses to sort growth companies: profitability, cash-flow profitability, cash flow above reported profit, steady quarterly earnings, steady sales growth, and how much the company spends on research, capital equipment and advertising. Six were already in the filing record; research and advertising spend were added to it that day. Each was read the same way as the six weighted trends — as a percentile against sector peers — and added to the shipped score through the same program, with the pass mark fixed beforehand: it would replace the shipped score only if it certified *and* beat it on both held-out halves. It did not certify. The combined score's rank correlation with the three-year outcome roughly doubled (0.331 against 0.169) and both held-out halves rose (0.379 and 0.286), but the edge swung so much from one three-year

window to the next (top-minus-bottom spreads from +0.17 to +0.60, against +0.18 to +0.34 for the shipped score) that the deflated Sharpe read 0.169 against the 0.95 required. Research, advertising and accruals earned no weight at all. There is a second reason it would have been declined even had it passed: the gain is mostly Winner Odds wearing different clothes. Profitability levels and earnings stability are what the Winners rank already measures, the combined score correlates 0.69 with the Winner percentile (the shipped score 0.32), and inside the Winner top quartile it reads only 0.069. Emerging Compounders exists to read *change* that the Winners list does not, and the stability signals find failures rather than the tail: on their own, the bottom quartile lost a median 55% over three years with 43 in 100 wiped out, the top quartile made +32% with 3 in 100 — and rose fourfold less often. Nothing shipped; the record is richer for the next test.

Re-run on a corrected history (2026-10-06): not certified. Two defects in the history every population test here reads were repaired before the Deep Score's own test: the industry map read drug makers as chemical companies and aircraft makers as retailers, and the SEC's data set for the first quarter of 2023 had never loaded, so most companies' fiscal-2022 annual reports carried the next year's filing date. The same pre-registered program was re-run on the repaired history, every variant counted again — except that V1, the old engine 2.0.0, is read from its frozen 2026-09-28 scores (the current engine no longer computes them) and only its outcomes were re-graded. **Engine 3.0.0 (V7) is not certified on the 2026-10-06 re-run**: its ranks still separate — rank correlation with the three-year outcome 0.163, top-minus-bottom spread +0.224, t 7.43 over five independent windows, held-out halves 0.159 and 0.136 — but the deflated Sharpe ratio is 0.48 against the 0.95 required. Two things moved it, and either alone would fail the bar: V7's own long/short Sharpe fell from 3.63 to 3.32, and the spread of Sharpe ratios across the thirteen variants rose from 2.32 to 3.89, most of it from V5, which is read on only four held-out windows. Whether that variant belongs in the deflation on the same footing is an **open investigation**. The engine stays live and every surface now says it is not certified; the figures are in scripts/inflection_program_result.json and the model register (inflection 3.0.1).

Insider buying was given its best chance, and still carries no weight (2026-09-29). The research literature offers one reason raw insider buying might mislead: most insider trades are routine — the same person trading in the same month every year — and only the rest carry information (Cohen, Malloy & Pomorski). Their rule was applied to every officer and director in the SEC's insider filings: an insider is classified only with a trade in each of the three preceding years, routine if one calendar month appears in all three, opportunistic otherwise. Before the filter, companies whose insiders were net

> buyers in the year before the annual report went on to a three-year median of –1.2% with 23 in 100 wiped out, against +24.9% and 9 in 100 for net sellers (rank correlation –0.106). After it, opportunistic net buyers made +6.9% with 17 in 100 wiped out against +32.3% and 4 in 100 for opportunistic net sellers (–0.030, negative in both held-out halves). The filter removes most of the damage and none of the sign. A signal whose correlation is not positive earns no weight, so the score is unchanged and insider buying stays on the page as a fact to investigate. One caution in fairness to the paper: it measures the month after a trade, this test measures three years after the annual report — the finding is that insider buying is not a three-year signal, not that the paper is wrong.

Deep Score reads how good a company *is*; the Inflection engine reads how fast it is *changing*. It looks for the tell-tale signs of a business hitting an inflection point — growth that is accelerating, margins widening, operating leverage kicking in, returns improving, cash turning positive — and scores them out of 100. A mature, excellent-but-flat company scores near zero here even while Deep Score loves it; that disagreement (Figure 1) is the entire point.

Eleven signals feed the "Inflection Score" (engine 3.0.0, 2026-09-28). Six carry weight; each is the name's *percentile among its sector peers* on the night of the scan (a sector-year with fewer than twenty known names ranks against the whole universe), and the score is the weight-averaged percentile over the factors the name has data for — a data-thin name is neither punished nor flattered, just lower-confidence, and a name with fewer than four of the six known gets no number. The other five are shown as facts with no weight. The weighted percentile is then re-based so that each published band keeps the share of the book it had that day, and capped at 50 by heavy dilution or a cyclical margin peak.

FIGURE 10 · THE ELEVEN INFLECTION SIGNALS — SIX WEIGHTED, FIVE SHOWN AS FACTS (ENGINE 3.0.0, 2026-09-28)

Shown as facts, no weight:

Growth acceleration · Early price breakout · Insider buying

Forward growth · Estimate revisions

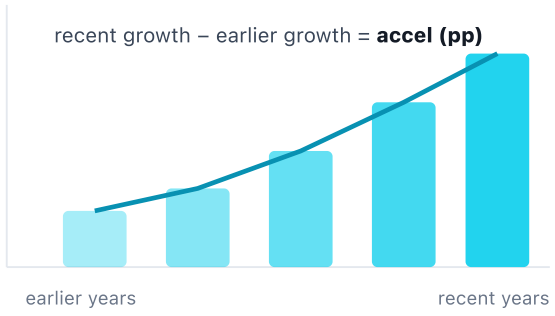
Each weighted signal is the name's percentile among its sector peers that night (a sector-year under 20 names ranks against the ur
the weight is the signal's rank correlation with the three-year outcome on the SEC 10-K population FY2009–2022, scaled to sum to

Weighted percentile → re-based to the published bands (each label keeps its 2026-09-28 share of the book) → capped at 50 by he

The heaviest weight sits on **F1 — growth acceleration**, the cleanest signature of an inflection. Green/teal signals come from the financials; blue signals (F9, F10) from analyst consensus; F8 (dilution) rewards discipline. `Source: inflection_factors.py — FACTORS · inflection.py`

FIGURE 11 · LEFT: WHAT "ACCELERATION" MEANS. RIGHT: THE CYCLICALITY GUARD.

ACCELERATION = A RISING GROWTH RATE



CYCLICALITY GUARD (materials / energy only)

1. **Troughed below 8% margin**
   in ≥ 2 years — a cyclical pattern
2. **Gave back ≥ 5pp from a prior high**
   a peak-to-trough drawdown once
3. **Back above 8% now**
   confirming it's at a peak, not a trough

All three → demote F1 ×0.4, cap score at 50.
Uses operating margin only. Reused by Winner Odds.

The guard exists because a commodity producer at the top of its cycle (think a fertiliser maker in a price spike) looks like a screaming inflection right before it gives it all back. It demotes rather than deletes, and abstains where history is too shallow to judge. `Source: inflection.py — _accel_pp, _cyclical_peak`

UNDER THE HOOD

```
score = round( 100 × Σ_scored earned / Σ_scored max ) # only signals it could measure gate: require F1 and (F2 or F4) scored, else "Insufficient signal" size gate: market cap ≥ $300M and revenue ≥ $50M # below this, % moves are noise cap: heavy sustained dilution (>15%/yr) → score capped at 50 cap: cyclical peak → score capped at 50 # bands ≥80 Inflecting · ≥65 Emerging · ≥50 Building · ≥35 Early · else None # confidence by COUNT of scored signals ≥5 high · ≥4 medium · else low
```

The robust primitives matter: growth steps whose base is near-zero or that flip into a loss are dropped (the ratio is meaningless there); with only two recent data points the engine takes the *minimum*, demanding both years show elevated growth so a one-off acquisition spike isn't mistaken for a trend; and the spliced trailing-twelve-month point is stripped out for balance-sheet ratios. Analyst signals F9/F10 need real coverage (≥3 analysts) or they go *unscored*, never a fake zero; only *upward* revisions score.

`inflection.py — _yoy_growths, _recent_estimate, _drop_ttm`

HOW IT CONFORMS · METHODOLOGICAL STANDARD

The engine is grounded in documented market anomalies — the **earnings-momentum / post-earnings-drift** and **estimate-revision** effects (F10), and quality-of-earnings via **accruals** — combined with robust trend measurement. Where it can be tested, a separate walk-forward harness reconstructs the fundamental factors from a decade of EDGAR filings and measures rank correlation with realised forward returns; it reports results as *discrimination, not alpha*, on the same honest footing as Chapter 7.

THE HONEST LIMIT · WHY F5 IS SPECIAL

F5 (price breakout) is a *market* signal, not a *business-quality* one, so it is deliberately (1) left out of the plain-English "why this is inflecting" thesis line, and (2) left out of the statistical back-test — price and insider activity have no clean point-in-time historical anchor to reconstruct. Since engine 3.0.0 it carries no weight: it is shown beside the score as a fact. The 3.0.0 score was tested as a rank on fundamentals only, on US 10-K filers FY2009–2022 (above): certified on 28 September 2026, not certified on the 2026-10-06 re-run under a corrected history (deflated Sharpe 0.48 against 0.95; the ranks still separate, rank correlation 0.16), and it reads US-filer history only.

Chapter 9

Discovery — the open-universe bridge

How brand-new names and themes enter the scored universe — from the open web, but never on blind trust.

A screener can only rank what it already tracks. Discovery is the bridge to everything else: an AI agent that reads *your* portfolio's concentration and a census of which sectors the universe already covers, then searches the web for real listed companies in the themes you're **thin** on — always keeping your growth tilt and never telling you to abandon a conviction you hold. Crucially, a proposal is **untrusted until it is validated and scored by the same engines as everything else**.

- **Grounded, not hallucinated.** The agent is given real tools: your actual sector weights and single-stock concentrations, and a SQL census of "names scored and median Deep Score per sector." It proposes into gaps and can never invent a fundamental number — only a ticker, a theme, and a cited source. `discovery_agent.py`
- **Resolved to a real symbol, then de-duplicated.** "BROADCOM" is resolved to `AVGO`; stale or unlistable names are dropped; anything you already hold or that's already scored is removed in code, not just by asking the model. `discovery_agent.py — _resolve_symbol`
- **Scored on approval, then permanent.** Approving a proposal runs the full scheduled treatment on demand — fetch fundamentals, compute the sector-relative Deep Score, run the fair-value ensemble, rescore the winner cross-section — and inserts the ticker into a custom universe so every future scheduled scan keeps it fresh. `scanner.py — score_ticker_on_demand`

> **IN PLAIN WORDS**
>
> It's a research assistant that already knows what you own, goes looking on the open web for good ideas in the corners where you're *under-invested*, and then — before any number reaches you — sends each idea through the exact same scoring machine as the rest of the universe. The web can suggest a name; only the engines can score it.

WHY YOU CAN TRUST IT

The approval screen surfaces the very disagreement this document keeps returning to: a name can come back *low* on Deep Score but *high* on Inflection — a classic emerging compounder, accelerating but not yet proven. You see both, with their eligibility flags, rather than a single blended verdict that would hide the tension. Web results are treated as untrusted source material, never as instructions.

PART III

Valuation & sizing

A score tells you how good a business is. These two engines answer the next two questions every investor faces: what is it worth, and how much should I own?

Chapter 10

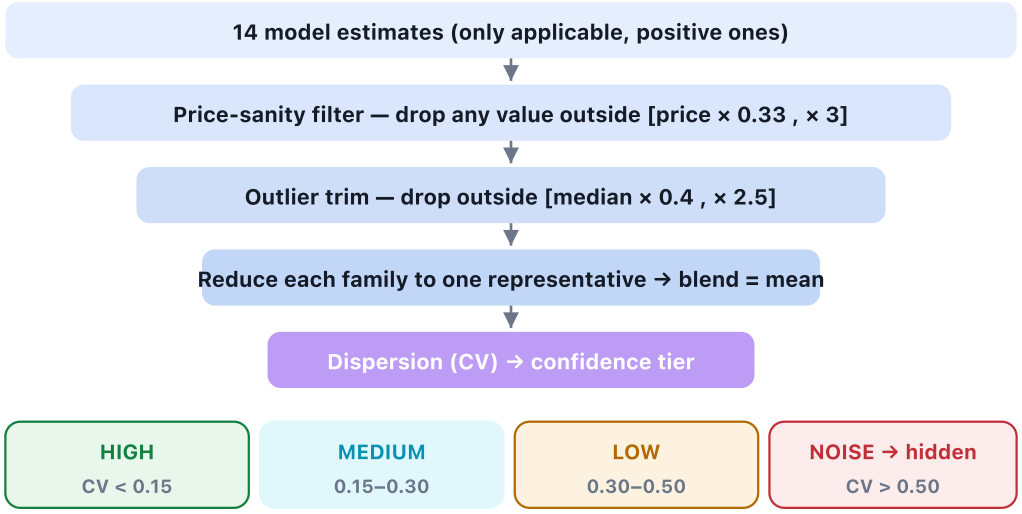
What's it worth? — the fair-value ensemble

Never trust one valuation model; run fourteen, throw out the nonsense, and only quote a number when they agree.

Valuing a company is genuinely hard, and any single method can be badly wrong for a given business — a dividend model is useless for a company that pays no dividend, an asset model misprices an asset-light software firm. So CompoundWise runs **fourteen independent valuation models across four families**, discards the ones that produce obviously broken numbers, blends what survives, and — the honest part — **refuses to show a fair value at all when the survivors disagree too much**. A trustworthy "no estimate" beats a confident fantasy.

Family	Models	How that family thinks about value
DCF (3)	FCFF perpetuity · exit-multiple · two-stage	The business is worth the cash it will throw off, in today's dollars
Multiples (6)	EV/EBITDA · P/E · P/B · P/S · P/FCF · Lynch fair value	Worth what similar companies trade for, per unit of earnings/sales/book
Asset / earnings-power (4)	Graham Number · Graham Revised · EPV · Residual income	Worth its assets, or its no-growth earnings power (a floor)
Dividend (1)	Gordon Growth DDM	Worth the growing stream of dividends it pays — read from the valuation dividend: the declared forward rate, or where the source gives none its forward dividend yield × price, cross-checked against the dividend paid over the last 12 months and labelled *implied*; a known zero is not a payer and the model does not run

FIGURE 12 · THE ENSEMBLE FUNNEL — 14 ESTIMATES IN, ONE TRUSTWORTHY NUMBER (OR NONE) OUT



CV is how tightly the family estimates agree. In the production path a fair value is only shown if at least **two independent families** corroborate it and their spread isn't NOISE — otherwise the name is shown with *no* fair value rather than a made-up one. `Source: valuation/aggregate.py — blend() · registry.py`

UNDER THE HOOD · THE DCF, AND HOW QUALITY BENDS THE DISCOUNT RATE

```
# The 3 DCF models discount TRUE unlevered free cash flow (FCFF) value
= Σ FCFF_t / (1 + WACC)^t + terminal value then EV − net debt → equity
cost_of_equity = risk_free + beta × ERP + quality_adj(quality_score)
                    # deep_score ≥80 lowers it, ≤40 raises it
WACC        = max( cost_of_equity blended with after-tax cost of debt ,
7.0% floor ) terminal growth = 2.5% projected growth capped at 12%
margin of safety = (fair_value − price) / fair_value
```

Three deliberate guards keep DCFs from exploding: a **terminal growth of 2.5%** (no company outgrows the economy forever), a **12% cap** on projected near-term growth (don't extrapolate a hot streak), a **–10% floor** on projected DECLINE (the same logic mirrored: do not extrapolate a fall off a cliff), and a **7.0% floor** on the discount rate — the 2.5% terminal growth plus a 4.5-point spread, so the denominator can never narrow to the point where far-future cash dominates the appraisal. And higher-quality businesses (a strong Deep Score) get a gently *lower* discount rate, because a durable business's future cash is genuinely more certain. `valuation/models/dcf.py — cost_of_equity, _wacc, constants`

HOW IT CONFORMS · CFA-CURRICULUM VALUATION

Every model in the ensemble is a standard, teachable valuation method straight from the **CFA equity-valuation curriculum** — DCF/FCFF, relative multiples, residual income, dividend discount, Graham/EPV asset methods. Blending multiple independent models and disclosing dispersion is exactly the discipline that curriculum urges: no single model is authoritative, and a wide spread is itself information.

THE HONEST LIMIT

A fair value is a *range built on assumptions*, not a price target. When the models can't agree, the product shows you nothing rather than a false precision — and even a HIGH-confidence estimate is only as good as the inputs (analyst growth, margins) behind it. Treat it as "roughly, is this cheap or dear?", never "this is the right price."

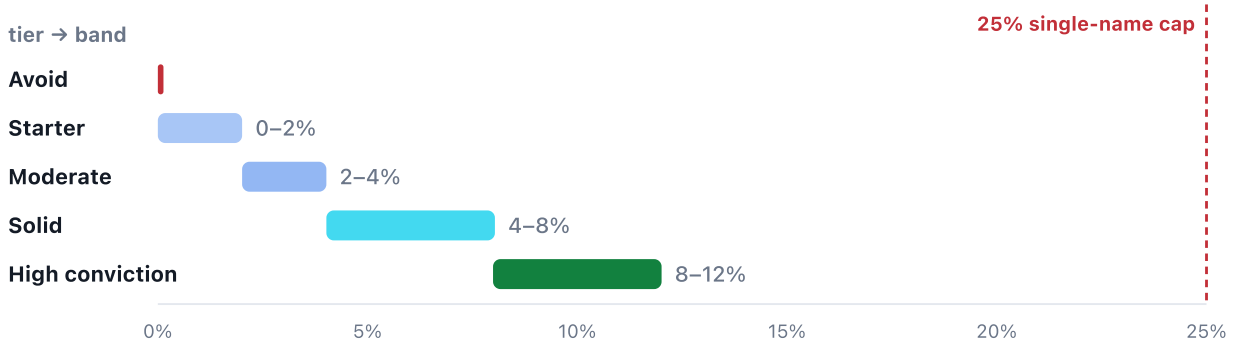
Chapter 11

How much to buy — position sizing

Turn a conviction level into a sensible weight band, with every adjustment named out loud.

Good investors size by conviction, but "conviction" has to be legible or it becomes a gut feeling. This engine maps the Deep Score to a suggested **weight band** — how much of your portfolio one name should be — and then nudges it up or down for the margin of safety, with every step written into a plain rationale. No AI, no hidden weights, and a ceiling that always stays well under the 25% single-name policy cap.

FIGURE 13 · THE SIZING LADDER — DEEP SCORE TIER → SUGGESTED WEIGHT BAND



Even maximum conviction tops out at 12% — comfortably below the 25% cap — so no single idea can sink the portfolio. `Source: sizing.py — BANDS, _TIER_CUTOFFS (cutoffs shared with Deep Score's 85/70/55/40)`

UNDER THE HOOD · THE ADJUSTMENTS, EACH SPOKEN ALOUD

- **Base tier** from the Deep Score, using the very same 85/70/55/40 cliffs as the labels — one source of truth.
- **+1 tier** if there's a **≥25% margin of safety** at trusted (HIGH/MEDIUM) fair-value confidence — a real discount earns a bigger position.
- **No promotion** for a deep discount on a *weak* business — "value-trap risk," said explicitly, so cheapness never rescues poor quality.
- **–1 tier** if the stock trades *above* fair value.
- **Sized on quality alone** when fair value is missing or low-confidence — valuation simply isn't counted rather than guessed.

A fund (ETF) returns "not applicable" — a diversified basket is an allocation choice, not a single-name conviction bet. `sizing.py — suggest_size, _MOS_UP_NOTCH`

HOW IT CONFORMS · PRUDENT-DIVERSIFICATION PRACTICE

Conviction-weighted sizing with a hard single-name cap is standard **risk-budgeting / prudent-diversification** practice. Making every adjustment explicit (and refusing to let cheapness promote a weak business) is a transparency and suitability safeguard — the user always sees *why* a size was suggested, and it is a suggestion, never an order.

PART IV

The money-math

From here on, the numbers are about *your* money — tracked to the penny, in two currencies, with Canadian tax correctness baked into every calculation. This is where "single source of truth" earns its keep.

Chapter 12

Total Value — the equity SSOT

One number for "what your portfolio is worth," computed once and shown identically everywhere.

The headline figure on every screen — your total value — is deliberately produced by a **single function**. Total value is simply the market value of your holdings plus your cash, in Canadian dollars. What makes it trustworthy is that the web app, the phone app, the widget, and every PDF all read that *same* function, so they can never disagree, and that the "how am I doing?" percentage is computed on a stated basis — gain over net contributions. Brokers differ in what they show; Questrade's own headline figure is a time-weighted rate, so the two will not always agree.

UNDER THE HOOD

```
market_value = Σ value_cad # holdings only, in CAD total_equity =
market_value + cash_cad_equivalent # the "how am I doing since I
started?" switch if net_deposits known and > 0: pnl = total_equity –
net_deposits # matches your broker return_% = pnl / net_deposits #
basis: "since inception" else: pnl = unrealized gain vs cost # basis:
"vs cost" today_pnl = Σ shares × day_change_native × fx # excludes any
name with no prior close
```

Two honesty touches: today's move *excludes* any holding with no prior-day close (it returns nothing, never a fake $0), and a hard "parity contract" requires the combined total to exactly equal the sum of the individual accounts. `advisor_engine.py – build_account_summaries, _pnl_vs_deposits, _combined_block`

HOW IT CONFORMS · GIPS / BROKER RECONCILIATION

Reporting return on a **since-inception, net-of-deposits** basis (equity minus what you put in) is the intuitive money-weighted view and is what brokerages like Questrade show — so the figure reconciles against your statement instead of quietly using a different definition. When deposit history is incomplete the app clearly relabels the basis as "vs cost" rather than pretending. Labelling the basis is itself a GIPS-spirit disclosure. The Tax report and the AI Tax & Structure Audit quote the same figure — (holdings + cash – net deposits) ÷ net deposits — with its basis labelled "since inception" or "vs cost".

Chapter 13

Am I beating the index? — TWR, MWR & the benchmark twin

The question most apps dodge, answered three honest ways from your real transaction ledger — no estimates, no AI.

"Am I actually good at this, or would I have done better in an index fund?" CompoundWise answers it three complementary ways, all reconstructed from your actual buys and sells: **what your dollars earned** (money-weighted), **what an index would have earned with the exact same deposits on the exact same days** (the benchmark twin), and **what your holdings did against the index with deposit-timing stripped out** (time-weighted) — a difference, not a measure of skill. Each answers a different question, and together they leave nowhere to hide.

FIGURE 14 · THREE HONEST ANSWERS TO ONE QUESTION

MWR / XIRR
"what my dollars earned"

Weights every dollar by how long it was invested. The return you actually lived, including timing.

Root-solved from your dated cash flows.

THE TWIN
"same money, index instead"

Replays every buy & sell, dollar-for-dollar, same dates, into XEQT or SPY. Its value vs yours is the gap — not a skill score.

Δ$ = you + your dividends reinvested in index − twin.

TWR
"holdings vs the index"

Chains daily returns with deposits/withdrawals removed — so a well-timed deposit can't flatter you.

The GIPS-consistent way to compare against an index.

The **twin** is the emotional gut-punch (a dollar figure: you vs the index with identical deposits); **TWR** is the like-for-like line against the index — a difference, not a measure of skill; **MWR** is your lived experience. All three come from the ledger — dividends reinvested symmetrically on both sides, oversold-twin capped. `Source: performance.py — _xirr, _twin_units, _chain_twr`

UNDER THE HOOD

```
# MWR / XIRR — the rate that makes your dated flows net to zero 0 = Σ flow_t / (1 + r)^(days_t / 365.25) # solved by bisection # TWR — chain daily sub-returns, strip the flows level ×= max( 0 , (V_t − Flow_t) / V_t−1 ) twr_annual = (1+twr)^1/years − 1 # annualised only with 365 days or more of history; under a year the cumulative return for the period is shown (GIPS 2020, 2.A.12) # Twin — same cashflows into one index fund units += amount / index_price_cad(date) alpha_% = MWR − twin_MWR Δ$ = holdings_today + your dividends as if reinvested in the index − twin_today # a gap, not skill
```

A subtle correctness point: the TWR flows come from the *reconstructed* curve (shares moved × that day's split-adjusted close), **not** the ledger's dollar amounts — because mixing split-adjusted prices with as-traded dollars would fabricate huge phantom losses on big buy days. The `max(0, …)` guard lets a full exit-to-cash-and-re-enter round trip restart the chain instead of cratering to zero. `performance.py`

HOW IT CONFORMS · GIPS & CFA PERFORMANCE STANDARDS

Time-weighted return is the method the Global Investment Performance Standards and the CFA curriculum prescribe for comparing a manager against a benchmark, precisely because it removes the distortion of deposit timing. **Money-weighted (IRR)** is the standard when the investor controls the cash flows — which you do. Presenting both, plus a same-cashflow benchmark and a clearly disclosed scope, is textbook honest performance reporting.

THE HONEST LIMIT

Scope is disclosed right on the panel: it covers your *holdings*, on a *total-return* basis. Dividends: the index twin reinvests its distributions; for the dollar comparison your dividends are treated as reinvested in the same index from the day you received them, so both sides are on a total-return basis. A benchmark whose first close is more than 7 days after your first transaction (XEQT launched in August 2019) is not compared — no twin, gap or alpha — and its line starts at its launch; a US-dollar benchmark (SPY) is drawn in Canadian dollars, like the book. An in-kind transfer between your own accounts (both legs sharing a transfer group) is not new money and is not a flow; a one-sided broker "Transfer in:" counts as money in at that day's market value, not its carried cost. The one real limit is that idle cash is excluded, so cash drag does not appear in the return, and an over-sold twin is capped so it can't go short. It answers "what did my holdings do against an index" — a gap, not a measure of skill, and over a short history most of a gap is luck — not "what is my whole financial life's return." For that, read the statement-basis table below.

The statement basis — your account return the way a CRM2 report calculates it

Canadian registrants send every client an annual investment performance report (NI 31-103 s.14.18–14.19, the "CRM2" report). Section 14.19 says what it contains: the market value of *all cash and securities* at the start and end of the period; deposits, withdrawals and transfers of cash and securities at *market value*; an annualised money-weighted return *net of charges*; for the last 12 months, 3, 5 and 10 years and since the account opened; and nothing under a year annualised (14.19(6)). The panel's **Your account returns — statement basis** table rebuilds exactly that from your ledger, per account and for the whole book, to today or to 31 December (the period a broker's report covers).

- **Cash is rebuilt day by day** from deposits, withdrawals, trades with their commissions, dividends, returns of capital, account fees, interest and currency conversions, and must **reconcile** with the balance on record today — your broker's for a synced account, yours otherwise — within C$1 or 0.05% of the account's value, whichever is larger, in each currency. 0.05% bounds what an unexplained residual can do to the cumulative return (about 0.05 percentage points). If it does not reconcile, the account shows **N/A with the reason**, and so does the whole book: a return built on a cash path known to be wrong is not shown. A manual account's balance counts only once someone has entered it (the stored default of 0 is not an entry).
- **Positions reconcile too.** Cash alone cannot see a wrong share count — an unrecorded 2-for-1 split moves no cash and halves the value. Each holding's share count must match the holdings on record within 0.001 shares, every split the market data shows while a holding

was held must be booked in the ledger (a reported ratio within 5% of 1 — Yahoo shows one on some Canadian ETFs' year-end distribution dates — is not treated as a split), and a synced account whose last sync found it does not match the broker is withheld with the broker's own words.

- **Flows at market.** A security moved in or out is valued at that day's close; at account scope a move between two of your own accounts is money out of one and into the other (as on each account's statement), and in the whole book the two legs cancel.
- **Statement prices.** Holdings are valued at closes adjusted for splits but *not* for dividends — the dividend is already cash in the account, and a dividend-adjusted price would count it twice.
- **Net of charges.** Commissions sit in the trade amounts; account fees, margin interest and the currency-conversion spread leave the cash. Broker interest, fees and conversions are booked as ledger rows (`$CASH` FEE / INTEREST / FX) by the Questrade and SnapTrade syncs and the Questrade and Wealthsimple file imports.

> **THE HONEST LIMIT**
>
> A cash path that reconciles today can still hide two past errors that cancel; a trade the broker has not yet listed in its activity feed reads as a mismatch until the next sync. A holding with no market price on a date makes the periods that need it N/A — s.14.19(7) would value it at zero; this app does not. **Interest income is not in the Tax Centre estimate** (your broker reports it on a T5), and an account fee is not treated as deductible.

Your policy mix, and returns by period

When an investment policy is on file, the default benchmark is **your own policy mix**: your equity / fixed-income / cash targets held in XEQT (global all-equity), XBB (Canadian universe bonds) and ZST (ultra-short bonds, standing in for cash — the savings-account fund CASH.TO only begins in November 2021), put back to target on the first trading day of every month. It is labelled a custom benchmark with its components and rebalancing (GIPS 2020 4.C.33). There is no splice to an older proxy: before its youngest weighted fund launched it is not compared, the same rule as any benchmark. Policy targets are not versioned, so the whole history is measured against the policy you hold today. XEQT and SPY remain one tap away.

Returns by period lists each calendar year and the trailing 1, 3, 5 and 10 years for you and the benchmark, computed from the same two curves the chart draws (GIPS 2020 4.A.1(b),(e)). Its basis is stated beside it (4.A.3): time-weighted, holdings only, distributions reinvested, *before* commissions and account fees; the benchmark is its total return after its own fund fee; both in Canadian dollars. Three years and longer are per year; a part-year is for the period. A benchmark

that had not launched at a period's start is N/A — never 0%. Sources: NI 31-103 (OSC consolidation, in force 2023-09-13) and GIPS 2020 for Firms, both pinned in docs/sources. GIPS compliance is not claimed — its calculation conventions are borrowed.

Chapter 14

The performance curve — splits, CDRs & reconstruction

The unglamorous plumbing that makes the curve true: handling stock splits and Canadian depositary receipts without fabricating losses.

A performance chart is only as honest as the price history behind it, and price history is full of traps. Two in particular — **stock splits** and **currency-hedged Canadian depositary receipts** — will silently corrupt a naïve curve. CompoundWise handles both with purpose-built reconstruction so the line reflects reality, not a data artifact.

- **Splits.** Price feeds show *split-adjusted* history, but your ledger records a split as extra shares on the split date only. Multiplying raw shares by adjusted prices therefore understates every pre-split day. The fix multiplies your share count on each past day by the product of every split ratio that came *after* it — using ratios derived from your own ledger's split entries, not the feed's unreliable CDR split data. `portfolio_core.py — _split_factor_series`
- **CDRs.** Batch price downloads silently drop some Canadian depositary receipts and lack their pre-launch history. Rather than flat-filling a gap (which made the curve visibly swing on every refresh), the panel retries each missing ticker individually and *reconstructs* genuine gaps from the U.S. underlying — a hedged CDR's Canadian price tracks the underlying's return, so it anchors a known CDR price and scales by the U.S. move. Real CDR prices always win where the feed has them. `portfolio_core.py — _complete_price_panel, _reconstruct_cdr_series`
- **Weekend-dated trades.** The curve is drawn on business days. A trade dated on a Saturday or Sunday counts from the next business day: each day takes the share count and cost base as they stood on or before it. Until 14 October 2026 such a trade was left off the curve altogether, which understated every day after it.

IN PLAIN WORDS

If you owned 100 shares that later split 4-for-1, a lazy chart would look at today's "400 shares" against yesterday's split-adjusted prices and decide you were down 75% before the split — a total fiction. This engine rewinds the share count correctly, and for Canadian proxies of U.S. stocks it rebuilds any missing price history from the real U.S. stock, so the curve doesn't jump around every time the page reloads.

WHY YOU CAN TRUST IT

Zero-equity days (a fully-cashed-out day) are dropped by default so the chart never shows a phantom –100% crash, and the whole performance engine is locked by golden tests that would fail the moment a split or CDR fix regressed. This machinery was built in response to real bugs and each fix carries its own test. `tests/test_performance.py`

Chapter 15

Adjusted Cost Base — the CRA cost-base replay

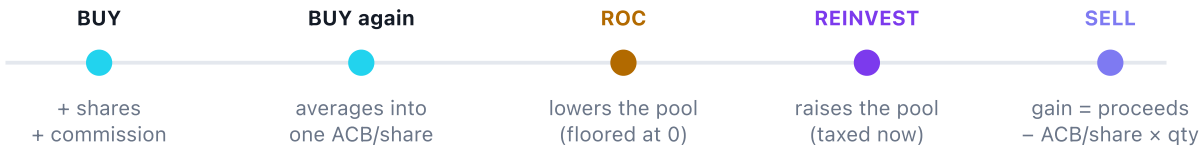
The running average of what you paid — the number the CRA makes you use to figure a taxable gain.

When you sell in a taxable account, your gain isn't "sale price minus what I paid for these particular shares." Canada requires the **average-cost method**: all identical shares share one running ACB per share. CompoundWise computes this by **replaying your entire transaction history in order** through one engine, applying each CRA rule as it goes — commissions add to cost, returns of capital reduce it, reinvested distributions raise it.

FIGURE 15 · THE ACB REPLAY — HOW ONE NUMBER EVOLVES THROUGH A POSITION'S LIFE

KEY RULE

Selling **never changes the per-share ACB** — it only removes shares at that average. Each buy locks in its own FX rate (Chapter 18).



One engine drives both the native-currency and the Canadian-dollar cost base — the CAD formula is literally the native one with each buy converted at *its own* purchase-date exchange rate. `Source: portfolio_db.py — _replay_position_acb, recalculate_acb_cad, holding_cad_acb`

HOW IT CONFORMS · INCOME TAX ACT S.47 (IDENTICAL PROPERTIES)

The average-cost pooling of identical shares is mandated by **ITA s.47**. Commissions capitalised into cost, **return of capital** reducing ACB (and triggering a gain if it drives ACB below zero), and reinvested/"phantom" distributions increasing ACB are all standard CRA cost-base mechanics. Doing this replay in one engine means every cost, gain, and tax figure in the app is computed on a compliant basis by construction.

One pool, every reader — since October 2026. The pool is the *taxpayer's*: every non-registered account holding the security feeds one average, and each registered plan is its own (the plan trust owns those shares). Booking already priced a sale at that pooled average; six other paths did not, and each now reads the same replay. The order preview's tax estimate and its "average cost after" for a buy (an *edit* is previewed as a replacement — the position, the pool and the superficial-loss deferrals are read without the row being edited, as booking replays them after the edit); the transfer preview; the price stamped on both legs of an in-kind move (so a move between two of your own non-registered accounts carries the pooled cost and books exactly $0 — it is not a disposition at all under the s.248(1) definition, there being no change in beneficial ownership); the broker transfer linking; the stored cost base on *every* account in the pool after any trade, delete, transfer or nightly exchange-rate fill in any one of them; and the T1135 in-year peak below. A sale is capped at the shares in *its own* account as well as the pool's, so a mistyped oversell in one account can never sell its sibling's shares. Measured on the production book before the change: no client yet holds one security in two taxable accounts, so no published number moved. `portfolio_db.py — position_acb_per_excluding, refresh_pool_holdings, _replay_position_acb · portfolio_core.py — _t1135_peak_foreign_cost · tests/invariants/test_s47_pool_one_decider.py`

WHY YOU CAN TRUST IT

There is exactly one cost-base engine, and one display function (`holding_cad_acb`) that every Canadian-dollar cost/gain surface routes through — so the Advisor, the Tax Centre, the holding page and a PDF report can't show different cost bases. The rules live in one place, pinned by a dedicated cost-base test suite.

`tests/test_acb_costbase_engine.py`

UNDER THE HOOD · SAME-DAY TRADES AND BACK-DATED LOTS (OCTOBER 2026)

s.47(1)(b) averages the cost at each purchase over the shares owned *immediately after* it, so a purchase made after a sale cannot enter that sale's cost. A broker reports a day's trades with no time of day, and nothing establishes the order it lists them in, so a Questrade sync settles a day's sales against the shares the account held at the start of that day (every row already on file, including a kept transfer leg, plus the import's earlier days). Until 16 October 2026 every same-day purchase was booked ahead of every sale, so selling 100 shares held at $10 and buying 100 back at $50 recorded a $2,000 gain instead of $4,000 and a $3,000 cost base instead of $5,000. A purchase goes ahead of the sale only as far as the sale needs shares bought that day. The known limit: a genuine buy-then-sell day trade on a position already held is booked sale-first, because the dates carry no time to tell the two apart. A holdings-template file that adds a lot dated before a sale already on file now restates that sale's gain, as the other importers do. `questrade_sync.py — _replay_order · api/import_data.py — _parse_template`

UNDER THE HOOD · THE LEDGER KEEPS ITS OLD VALUES, AND YOU CAN TAKE IT WITH YOU

Every edit or deletion you make to a transaction, and every change the system makes to an input your cost base is built from — an exchange rate, a price, a share count, a date, the currency — is kept with its value before and after. An edit re-books a trade under a new id, and the history links the two. Derived figures (realized gains, superficial-loss flags) are not logged: they are rebuilt from those inputs, so the history of the inputs explains every change in them. Two choke points make every such write, and a build-time test fails on any third. **Download my ledger** (web Settings, Android Settings and Delete Account) gives you all of it as one CSV: transactions, holdings with native and Canadian-dollar cost base, realized gains by year for non-registered accounts, closed profit and loss for every account, and the change history. The Income Tax Act (s.230) asks you to keep records for six years. `portfolio_db.rewrite_txn / delete_txn_rows · ledger_export.py · tests/invariants/test_txn_history_invariants.py`

Chapter 16

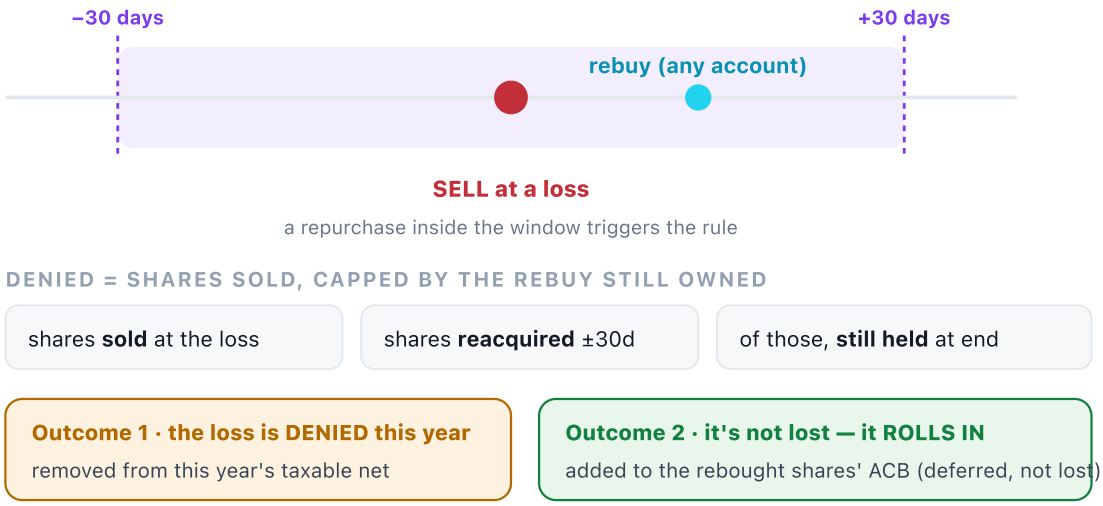
Superficial-loss rules — the regulatory showpiece

Canada's anti-"wash-sale" rule, implemented with a rigour most professional software never attempts.

If you sell at a loss to book the tax benefit and buy the same stock back within 30 days, the CRA disallows the loss — you can't claim a loss on a position you didn't really let go of. This is the **superficial-loss rule**, and it is deceptively intricate: it pools *all* your accounts, it treats a same-day buy in your RRSP as a repurchase, and the disallowed loss isn't gone — it's *added to the cost base of the shares you rebought*, as long as you still own them outside a registered plan when the window closes. CompoundWise implements every one of these wrinkles.

One further wrinkle decides where the window's edges fall. CRA's administrative position is that a disposition — and an acquisition — of an exchange-traded security happens on **settlement**, not on the day the trade is executed. CompoundWise therefore measures both the tax year a sale belongs to *and* the ±30-day window on settlement dates, and matches a repurchase on the date *it* settles. The distinction only bites near the edges of the window, but that is precisely where the rule is decided: a sale executed on 28 December 2023 settles on 2 January 2024, so a repurchase executed 32 calendar days later is 29 settlement days later, and the loss is denied rather than allowed. Settlement cycles are computed per exchange and per era — T+3 before September 2017, T+2 until May 2024, T+1 since — over a real holiday calendar, so an older trade is never re-dated on today's cycle.

FIGURE 16 · THE ±30-DAY SUPERFICIAL-LOSS TEST, AND WHAT HAPPENS TO THE LOSS



The denied amount is the shares you sold, capped by the shares you actually rebought **and kept**: for each holder — your non-registered accounts together, each TFSA or RRSP, and the same again for a declared spouse — the **least of** what it bought in the window and what it still holds at the end, added up. Shares one holder held throughout, such as a TFSA position bought years earlier, are not a rebuy and never stand in for another holder's; inside one holder identical shares are interchangeable, so they do cover that holder's own rebuy (your pool also holds a joint, corporate or in-trust account, as it does for your cost base). A rebuy withdrawn in kind from a TFSA or RRSP to your own account before the window closes is still kept when the move's two legs are linked; unlinked, the move reads as a custody transfer and the rebuy stops counting. An unlinked transfer into one of your individual accounts is never a purchase of its own: the shares it brings in join that holder like any others it holds. A rebuy in a *registered* account (TFSA/RRSP) denies the loss **permanently** with no roll-in anywhere; the timing of the roll-in lands at the *later* of the sell and the rebuy. `Source: portfolio_db.py — _superficial_window_qty, _superficial_cross_rollin, recalculate_realized_gains`

UNDER THE HOOD · TWO DIFFERENT QUANTITIES THE CRA NEEDS

- **Denial** (every account counts, each for itself): a rebuy in *any* of your accounts — including a permanent TFSA/RRSP rebuy — disallows the loss for the year, as far as that holder still owns what it rebought when the window closes. This is what's subtracted from your taxable net. Until 15 October 2026 the shares still held were added up across every account, so a TFSA position held for years could stand in for a rebuy you had already sold again; measured on the live book that day, no booked loss was affected.
- **Roll-in** (into the substituted shares, inside one pooled cost base): the disallowed loss is added to the ACB of the specific shares you rebought, so you recover the tax benefit when you eventually sell those. Since September 2026 the cost base itself is pooled across *all* your non-registered accounts, as s.47 requires, so the roll-in lands in that one pool rather than in a per-account average. A rebuy by a *spouse or common-law partner* you have declared denies the loss under s.251.1 with no roll-in to you — the deferral goes to *their* cost base under s.53(1)(f), and since 15 October 2026 the app books it there (before, that part was denied and added to no one's cost). When you both rebought, your own substitute takes the denied amount first, then your spouse's; the rest — a substitute inside a plan — is denied permanently. The Act names the buyer's cost but no order between two buyers; this is the order the order preview quotes.
- **Timing**: the deferral attaches at the *later* of the sale and the repurchase — so a disposition that happens *before* the rebuy correctly uses the un-inflated cost base; a spouse's substitute takes it at the same later date — at the end of the day of your sale, or on the day they bought, whichever is later — so a sale or transfer of theirs dated on or before that day is priced without it, and the T1135 peak takes it on that day; and the two of you are replayed together, so a loss of one that lands in the other's cost and is then part of a loss of theirs is sized correctly. A substitute bought in the same pool *before* the sale defers into the shares still held *after* it, whichever of your accounts it sits in.
- **Amount**: the roll-in is the loss as booked — measured against the pooled cost base as it stood at the sale, including any earlier deferral — so the amount added back always equals the amount denied, less any share whose substitute sits in a registered plan, which is denied permanently.
- **The substitute must still be owned** (since 13 October 2026): s.54 makes the substituted property what you (or a declared spouse) *own at the end of the window*, so the roll-in is capped by the non-registered shares still held then. A purchase

inside the window that was sold again before it ended takes no roll-in; if the shares owned at the end sit in a TFSA or RRSP, that part is denied permanently. The booking, the order preview and the independent oracle all apply the cap; until that date they rolled the loss into any non-registered purchase in the window, even one sold again.

- **When it is re-tested**: every change that can move the answer — a trade, or its edit or deletion, however long before the sale (what a holder still owns when the window closes counts every earlier trade); declaring or clearing a spouse, or leaving a household; force-deleting an account; a file import's Replace and a broker's Full rebuild, once the re-import is in. With a declared spouse, anything that re-books one of your cost bases — a trade, a return of capital, an exchange rate stamped or filled in on a sale — re-books the other's, since each of your losses can sit in the other's cost. Until 15 October 2026 only trades within about 40 days of a sale re-tested it, and the household, account and import paths re-tested nothing; measured that day, no taxable sale on the live book had a stored figure that differed from a fresh re-booking.
- **Your broker's figure** (since 16 October 2026): a broker's book cost leaves a denied loss out — Questrade's is "the cost of a security adjusted by any corporate actions, reinvested dividends, and return of capital" (its own help page, read that day) — so a Questrade or SnapTrade sync keeps the app's cost for a position that holds one, yours or a declared spouse's, instead of overwriting it. It is asked of the position as it stands, in both currencies: a loss in US dollars that was a gain in Canadian dollars denies nothing, but the dollar cost still carries it, and taking the broker's figure would rescale the correct Canadian-dollar cost — so that is kept too, and named for what it is. When such a position still has an exchange rate pending, it is kept and labelled as pending, since which of these it is cannot be said until the rate arrives. The Reconcile page checks a kept cost against the broker's: it matches when the broker's figure is the app's cost with or without the denied loss (and with or without a return of capital or reinvested distribution you entered by hand) to within $1 or 1%, whichever is larger, and flags it otherwise. The sync itself does not turn the account red for it, since some such gaps are legitimate (a transfer's book value). A position held across two taxable accounts is compared as it always was, since the broker's figure is one account's and the app's the pooled cost. Measured on the live book on 19 October 2026 with this rule, no linked account held a position it would keep.
- **What counts as a repurchase**: an *acquisition*. Moving shares you already own between two of your own *individual* accounts (cash, margin, USD) — or a broker "Transfer in" of shares arriving there at their carried cost — acquires nothing and

never denies a loss. A move *into* a TFSA or RRSP does count (the plan trust acquires them, and it is affiliated with you), and so does a move into a joint, corporate or in-trust-for account, each of which changes who owns the shares.

The net effect on a year's tax is a single clean identity: `allowed loss = realized loss + denied amount` (the denied amount is a non-negative add-back). `portfolio_core.py — _allowed_gain_cad`

HOW IT CONFORMS · ITA 40(2)(G)(I) & 53(1)(F)

One rule, one implementation — since 24 September 2026. The Tax Centre's harvest list used to answer this question with its own shortcut: "was there any purchase of this ticker in the last 30 days?" That is not the rule. It has no *quantity* test, so rebuying one share blocked harvesting a thousand; no *still-held* test, so a rebuy that had since been sold again still blocked a loss it cannot deny; no *affiliated-person* scope, so it said "harvest now" on a loss a spouse's purchase denies; it counted a $0 stock-split row as a repurchase; and it read the trade date rather than the settlement date. Both directions were wrong — it blocked harvests that were fine, and cleared harvests the CRA would deny. The harvest check now calls the same primitive the signal engine, the year-end summary and the cost-base replay use, and the ±30-day window is one named constant rather than six inline copies. Measured on the live book before the change: four taxable loss positions, zero where the two answers differed — so no published number moved. It was a second implementation of a tax rule, removed before it drifted. **And one scope, since 13 October 2026.** Consolidating the harvest list fixed the surface that was measured and left two that were not. The question "whose purchases count?" was still answered separately at four places in the code, and two of them answered "yours only": the amber *Superficial* flag in Transaction History, and the rebalance planner's tax preview. So a declared spouse's rebuy raised no flag, and a plan could quote a harvest the ledger would then deny — while the paragraph above this one, and the support guide, already told you a partner's purchases counted. The document was ahead of the code on two screens, which is the more embarrassing direction and the reason the scope is now a single named decider that every denial path reads. Two things were needed, not one: the planner caps a denial at the least of (shares sold, shares reacquired, shares still held when the window shuts) and it built that last figure from *your* holdings alone — so selling out of a position entirely drove it to zero and the cap swallowed the denial however much a partner had bought. Widening only "who reacquired" would have changed nothing, and a test now re-introduces exactly that half-fix to prove it. Measured on the live book before and after: nobody has declared a spouse, so no flag and no quoted tax moved — this closes a gap between what the preview promises and what the ledger books, rather than correcting a number anyone was shown. The Tax report and the AI Tax & Structure Audit quote harvest savings only on losses the rule would let a sale *today* keep; a blocked loss is listed as blocked, not priced. The Tax page's headline on web and Android does the same — *"Tax you could save by harvesting now"* — with the losses that have to wait out the 30-day window on a second line.

The 30-day, all-accounts, "still-held" test is the **superficial-loss rule of ITA 40(2)(g)(i)** (with the definition in s.54); the denied loss flowing into the substituted shares' cost base is **ITA 53(1)(f)**. Pooling identical property across an individual's accounts — including registered plans and a spouse's holdings in the general case — is exactly the CRA's stated position. Getting the permanent-denial-in-registered case and the roll-in *timing* right is the difference between an estimate and a compliant calculation.

WHY YOU CAN TRUST IT

This is the most heavily-tested area of the whole codebase. An independent "oracle" re-implements the rule from scratch and cross-checks the engine, and dedicated suites cover cross-account roll-in, pooling, and the timing edge cases. When a new buy is recorded, the app even re-evaluates nearby loss-sales across *all* accounts to update their denied amounts. `tests/invariants/superficial_oracle.py` · `tests/test_superficial_cross_account_rollin.py`

Chapter 17

In-kind & account transfers — deemed dispositions

Moving shares between accounts can be a taxable event or a tax-free carry-over; the app decides correctly from the account types.

Transferring shares "in kind" (as shares, not cash) has sharply different tax consequences depending on where they go. Moving them from one non-registered account to another just carries the cost base along, no tax. Moving them *into* a registered plan is a **deemed sale at market value** — a gain is taxable and, importantly, a *loss is denied*. The app books each leg from the account types alone, so a sign or a rate can't be fat-fingered.

Transfer	Tax treatment	Cost base at destination
Non-registered → non-registered	No tax — cost carried over	Same ACB (book value)
Non-registered → registered (contribution)	**Deemed disposition** at market; gain taxable, **loss denied**	Market value
Registered → non-registered (withdrawal)	Withdrawal may be taxable income; cost base leaves at book	Market value
Registered → same registered type	No tax	Carried
Two *different* registered plans	**Blocked** with a plain CRA explainer	—

HOW IT CONFORMS · DEEMED DISPOSITIONS & FOREIGN-CURRENCY GAINS

Contribution-in-kind as a deemed disposition at fair market value with a denied loss is standard CRA treatment (the same stop-loss logic behind the superficial rule). And every deemed gain/loss is decided **in Canadian dollars** — a position that lost money in U.S. dollars can still be a real CAD gain once the exchange rate moved, which is exactly what **ITA s.39(1.1)/s.261** require. Qualifying direct rollovers (RRSP→RRIF, LIRA↔LIF) are recognised as non-taxable.

UNDER THE HOOD · BROKER-IMPORTED TRANSFERS

A broker reports an in-kind transfer as two separate rows at book value. After every account has synced — on a sync and, since 16 October 2026, on a Full rebuild too — the two legs are paired by ticker, share count, book value and the account numbers in the broker's description, and the deemed disposition is booked. A transfer *out* of a taxable account whose other leg is not on file (its account is unmapped, switched off, or failed to sync) is listed for review rather than left silently at book value. A sync or rebuild that fails on one account keeps the others it already saved and names them. A Questrade *file* import notes an in-kind transfer in the way the sync does, so it counts toward net deposits. `questrade_sync.py — reconcile_transfers · api/questrade_api.py — questrade_rebuild, _failure_message`

Chapter 18

The FX chokepoint — one currency source of truth

Every USD↔CAD conversion in the app flows through one guarded gate, so a bad exchange-rate tick can't poison your cost base.

A Canadian investor holding U.S. stocks lives in two currencies, and a single wrong exchange rate can quietly corrupt a cost base or a gain. So all conversions pass through one module with a **plausibility band** — USD/CAD must sit between 0.85 and 2.0, which rejects the glitched feed ticks (a feed once returned 495) that would otherwise wreck a calculation.

UNDER THE HOOD

- **Live rate:** cached briefly; on an implausible tick it keeps the last good rate rather than propagating garbage. The quote is read through one shape-safe reader, because the feed returns different shapes on different days — a two-day window normally gives two rows, and gives one whenever it spans a single trading day (a Monday, the morning after a holiday, a partial feed). Until 24 September 2026 that one-row case raised an exception and the live quote was discarded for no reason. Nothing was ever invented by it: the fallback chain is a real rate — the last rate the process observed, else the Bank of Canada published rate — so the cost was a slightly *stale* rate rather than a wrong one (measured that day: a live 1.4144 against a published 1.4136, a divergence of 0.06%). That stale rate is acceptable for a display and not for a booking: a trade, transfer or rebalance *dated today* takes only a rate *observed today* (a live quote, or the Bank's quote for today) — never a rate cached on an earlier day, nor yesterday's Bank quote on a cold start (review F1, 13 October 2026).
- **Historical rate — strict, and one decider for booking and preview:** a past date takes that day's close or, for a weekend or holiday, the *closest preceding* day's (Folio S5-F4-C1 ¶1.4) — until 13 October 2026 it looked five days forward, so a Saturday took Monday's close. Today has no close yet (its bar is still trading), so it is never returned or cached as one. If a genuine rate isn't available it returns *nothing* and stores a null, rather than stamping *today's* rate onto a years-old purchase. The order preview asks the same decider: with no rate for the sale's date the tax estimate is *unknown* (it used to take the live rate, so a preview could show a gain the ledger then booked as a loss), and the booked sale's gain stays empty — listed on the Tax page and in the export as awaiting its rate — until the overnight fill, never replayed at 1.0.
- **Broker imports — strict:** so a feed outage during an import can never silently distort an imported cost base. A U.S.-dollar deposit or withdrawal whose date has no rate yet is not booked on that sync; since 16 October 2026 it is kept as *pending* with the broker's own record of it and booked by a later sync once the rate exists, without moving the account's sync watermark (before, a row older than the 7-day overlap was never seen again), and the activity log names it. A Full rebuild during such an outage keeps a deposit already booked rather than deleting it. When the overnight backfill later fills a missing booked rate, a holding whose cost base came from the broker keeps the broker's figure and takes only the rate — the same rule as the Bank of Canada pass; until 16 October 2026 that pass wrote the app's partial replay over

it. `questrade_sync.py — _retry_fx_pending, _keep_unpriceable_cash_rows · portfolio_core.py — _store_cad_acb_keeping_broker_figure`

- **The Bank of Canada rate for the tax cost base:** stamped overnight on every U.S.-dollar trade, dividend and return of capital. CRA's Folio S5-F4-C1 ¶1.4 (and ITA s.261(1)) names the rate the Bank quotes on the day, the closest preceding day when none is quoted, and the Bank's *noon* rate for any day before 1 March 2017 — so January and February 2017 use the noon series (they took the daily average until 7 October 2026; no stored row was dated in that window). The Bank publishes "once each business day by 16:30 ET", so a trade dated *today* waits for its own quote: until then its cost base uses the booked rate, and the stamp follows the next run. Until 7 October 2026 a lookup made before publication took yesterday's quote and kept it (measured on production that day: 0 rows affected).
- **Hand-entered transfers:** a backdated transfer is priced at the rate for *its* date or not at all — with no rate for that date, or a source purchase still waiting for its rate — or a denied loss rolled into the position whose own sale, or a purchase behind it, still is, yours or a declared spouse's (since 19 October 2026) — nothing is recorded and you are asked to try again. Until 7 October 2026 a failed lookup fell back to today's rate, or to 1.0 when every feed was down, and that substitute became the transfer's permanent deemed-disposition rate (measured that day: 0 of 4 manual U.S.-dollar transfers affected). A transfer's rate is never revised afterwards, so a U.S.-dollar transfer *dated today* is refused — by the preview and the booking alike — until the day has a close; record it the next day with today's date. The display-currency rate likewise answers "unavailable" rather than 1.0, and the screen keeps showing Canadian dollars and says why. A broker's in-kind transfer is held to the same rule (since 19 October 2026): while a purchase behind its source is still waiting for its rate, the transfer stays at its book value and is named for review rather than priced off an unknown cost — the overnight rate backfill never revisits a transfer's legs, so a parity-priced cost would have stayed in them. It is not re-priced automatically once the rate is in (nor is one left at book value for a missing market price or a rate-feed outage); that is an open item.
- **Daily history (performance, growth chart, currency split):** read by one reader that fails loudly. When the history cannot be fetched the performance panel says it could not calculate just now and keeps the last good figure, and the value chart says it could not load — until 7 October 2026 both drew the whole history at *today's* rate, which overstates past U.S.-dollar purchase cost (by 2.3%–7.7% per book on production, had the feed failed) and showed a currency effect of exactly 0.00. A book with no U.S.-dollar holdings never needs the feed. Each daily bar keeps its

exchange-local date (a London-midnight bar in summer used to be relabelled the previous day), and a truncated feed is carried forward at most five business days, then the day is unknown. **Known limit:** the feed's daily history starts on 17 September 2003, so a U.S.-dollar row dated earlier cannot be converted and that book's history answers "could not load" — measured on production 13 October 2026, the earliest U.S.-dollar row is 10 June 2016, so no client is affected.

- **"Today" is Toronto's.** The server's date — for the Bank's publication, for the latest date a trade, contribution or withdrawal may carry, and for contribution-room years — is the Eastern-time calendar date, not the server clock's UTC date; entry forms default to your own device's date.

The 0.85 floor is deliberately below parity so the app correctly admits the 2007–2013 era when the loonie was strong. `fx_rates.py — get_usd_cad, usd_cad_today, _historical_usd_cad, boc_historical_usd_cad, usd_cad_history, usd_cad_on_days, today_toronto · portfolio_core.py — usd_cad_for_date, _pending_fx_sales · api/transactions.py — _fx_for`

One money inside a valuation

The section above governs your *cost base*. A second, separate question governs a *fair value*: a company can be quoted in one currency and keep its books in another — sampled 2026-09-17, 53 of 140 TSX rows quote Canadian dollars and report U.S. ones. Every ratio that divides a market quantity by a financial one is then wrong by the exchange rate. Since 2026-09-23 the statement figures are restated into the currency the price is quoted in before any model runs, so the price, the per-share figures and the enterprise-value bridge are one money.

UNDER THE HOOD

- **The rate is stored, not inferred.** Until 2026-09-24 it was reconstructed as the ratio of two stored market-cap columns. That identity holds only while one writer moves both, and the intraday price refresh — which runs about every two minutes through the session — rescaled one of them and not the other. The implied rate therefore tracked the *share price*, every reporting-currency input scaled with the quote, and the fair value partly restated the price it was meant to judge. On a snapshot of 165 affected rows, a simulated 5% price move corrupted the rate on **165 of 165** before the change and **0 of 165** after.
- **No rate means no number, never a guess.** Only USD↔CAD has a rate here. Any other pair leaves the dependent inputs unset, the models that need them drop out with the currency named, and — since 2026-09-24 — the vendor fallback may not quietly refill them either, which had been turning an honest gap into a wrong-currency figure.
- **Both places that value a company use it.** The nightly scan and the live per-model panel on a stock page run the same engine on the same denomination. Before 2026-09-24 the live panel carried no currency at all and silently valued at par, so on 112 rows it disagreed with the headline printed directly above it.

`data/denomination.py — reporting_to_listing, normalize_to_listing, venue_for_ticker`

WHY YOU CAN TRUST IT

Because there is one gate, the plausibility rule is enforced everywhere at once — the performance curve, the ACB engine, and the tax calculations all draw from it. A dedicated "chokepoint" invariant test asserts that no code path bypasses it.

`tests/invariants/test_fx_chokepoint.py`

PART V

Tax & regulatory

Where the money-math meets the tax code directly: gains, dividends, and the machinery that keeps every CRA constant current and correct.

Chapter 19

Capital-gains tax

Only part of a gain is taxable, losses offset gains first, and the exchange-rate move is part of the gain — the app gets all three right.

In Canada only a fraction of a capital gain is taxable (the **inclusion rate**), and losses net against gains *before* that fraction is applied. CompoundWise applies inclusion to the **year's net**, uses the correct inclusion rate *for the year of each disposition*, and — crucially for a cross-border investor — treats the currency movement as part of the gain.

UNDER THE HOOD

```
taxable_gain = max( net gains − losses (this year) , 0 ) ×
inclusion_rate(year) est_tax     = taxable_gain × marginal_rate #
registered accounts (TFSA/RRSP/…) → zero capital-gains tax # proceeds
convert at the SALE-date FX; ACB stays CAD at the PURCHASE-date FX
```

Because proceeds are converted at the sale-date rate while cost stays anchored at each buy's own rate, the embedded currency move is preserved in the CAD gain — a taxable component under Canadian rules. Registered accounts short-circuit to zero tax, decided by the account registry, not guessed. `portfolio_core.py — _realized_gains_ledger, _est_sell_tax`

HOW IT CONFORMS · ITA S.38 & S.39

The taxable fraction of a capital gain is **ITA s.38**; netting losses against gains within the year is standard. Applying the inclusion rate *per disposition year* means a legislated rate change is handled by dating, not by a hard-coded constant. Preserving the FX component of the gain follows **ITA s.39/s.261**.

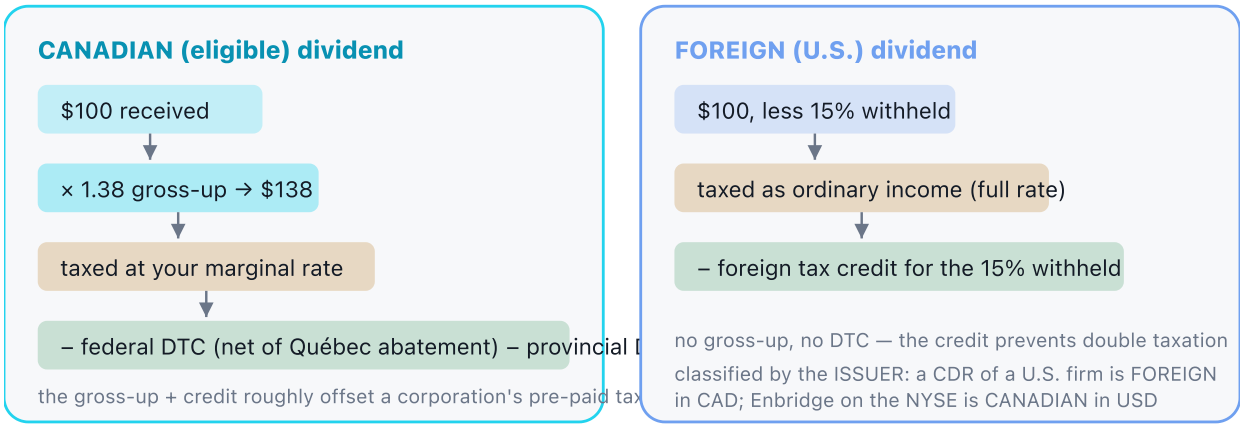
Chapter 20

Dividend tax — gross-up, the dividend tax credit & the foreign credit

Canadian and foreign dividends are taxed by two completely different machines; the app runs the right one based on where the company lives.

A dividend from a Canadian company and a dividend from a U.S. company are taxed in almost opposite ways, and — a subtle point most tools miss — what matters is the company's **home country, not the currency**. A Canadian dividend gets "grossed up" and then handed a credit; a foreign dividend is taxed as ordinary income but earns a credit for the tax already withheld abroad.

FIGURE 17 · TWO DIVIDENDS, TWO TAX MACHINES



Eligible-dividend gross-up is 38%; the federal credit is reduced by the Québec abatement for Québec residents; provincial credits vary by province. The foreign path grants a credit for the treaty withholding instead. `Source: portfolio_core.py — _dividend_income_tax · tax_config.py · tax_rules.json`

HOW IT CONFORMS · ITA S.82, S.121 & S.126

The dividend **gross-up** is **ITA s.82**; the federal **dividend tax credit** is **s.121**; the eligible vs non-eligible distinction is **s.89(1)**; the **foreign tax credit** for the 15% U.S. withholding (set by the Canada–U.S. tax treaty) is **s.126**. Classification is by the **issuer**, never the trading currency.

Who the issuer is — one decider (audit batch 1, 16 October 2026). Until then the engine said "foreign" when a holding traded in U.S. dollars (plus a hand list of ten CDRs) — so Enbridge or TD bought on the NYSE was taxed as a foreign dividend with a phantom 15% credit and no dividend tax credit, and every Canadian-listed fund, trust and CDR distribution was booked as an *eligible Canadian dividend*. `security_facts.tax_profile` now decides from the issuer, in this order: a checked-in override file (CDRs from CIBC's own CDR list; Canadian companies the market feed files under a U.S. head office — Restaurant Brands, Brookfield Asset Management, GFL, RB Global — from their SEC filings' state of incorporation); the curated fund list (a TSX-listed ETF there is a Canadian fund, a U.S.-listed one a U.S. fund); then the market feed's own issuer country and quote type, stored by each data scan. Each dividend then takes its issuer's treatment: a Canadian company's, on any listing, is grossed up and credited; a foreign company's or a CDR's is foreign income credited at the source country's treaty rate (U.S., Netherlands and Taiwan 15% — the treaty texts are retrieved and pinned; a country with no rate on file gets *no* credit, and the panel says so); a Canadian fund's or trust's distribution is listed as a *fund distribution* whose character is on your T3/T5 slip, and is never taxed here as an eligible dividend; a ticker nothing describes is *not assessed* — never guessed from its currency. Worked: US$1,000 from ENB on the NYSE at 1.40, Ontario 43.41% → C$1,400 eligible, 1,932 grossed up × (0.4341 – 0.150198 – 0.10) = **C$355.30**, no foreign credit (it was C$397.74 with a C$210 credit). C$1,000 from the AVGO.TO CDR → foreign income, C$150 credit, **C$284.10**. `security_facts.py — tax_profile · data/security_tax_overrides.csv · tests/invariants/test_issuer_domicile_one_decider.py`

The place of incorporation, not the head office (independent review, 19 October 2026). The market feed's country turned out to be a *head office*: it files Brookfield Renewable Corp (incorporated in British Columbia) under the United States — so a C$1,000 BEPC.TO dividend was taxed C$284.10 with an invented C$150 credit instead of **C$253.78** as an eligible dividend — and lululemon (incorporated in Delaware) under Canada, so it fell off the T1135. The country now comes from the SEC's EDGAR record of where the issuer is incorporated, which each data scan asks at most once a quarter per company and keeps; the market feed's country is used only where EDGAR has

none; and where both answer and disagree the issuer is *not assessed*. Every such disagreement found across the universe (28 listings) is settled in a short override file on two sources — EDGAR's record and the issuer's own annual-report cover — because either alone has been wrong (EDGAR's record says New Jersey for Bausch Health, Bermuda for Carnival). Canadian Depositary Receipts come from one list, CIBC's own directory of 133 receipts with each issuer's country. `security_facts.py — scan_country · sec_edgar.py — incorporation_code · data/security_tax_overrides.csv · data/cdr_receipts.csv · tests/invariants/test_b12_review_findings.py`

A foreign dividend is taxed on its gross. CRA's T5 guidance: report "the gross foreign income … Do not reduce the amount by any foreign income tax that was withheld." A broker row marked *NON-RES TAX WITHHELD* is booked net of the US withholding (checked on real rows: 86 AAPL × US$0.24 = US$20.64, booked US$17.55), so it is grossed back up (net ÷ 0.85) before tax and the foreign tax credit is taken on the gross. Rows without the marker are taxed as entered. Corrected 2026-09-29; before that the income and the credit were both 15% too small on netted rows.

The exchange rate a foreign dividend converts at is the one its cost base uses: the Bank of Canada rate for the day once it is published, else the rate booked on the row. Until 7 October 2026 the dividend read the booked (market-feed) rate while the cost base on the same day read the Bank's — a difference of about 0.14% (1,000 USD at 1.37262 vs 1.3707). This is consistency, not a compliance fix: CRA's Folio S5-F4-C1 ¶1.6 and ¶1.6.1 accept another widely available rate, or a period average for income, if used consistently. **A dividend with no rate at all is unknown, not parity**: it is left out, the panel says how many rows await an exchange rate, and foreign income, the foreign tax credit and the total read N/A until the rate arrives (overnight) — or, for a Canadian company's U.S.-dollar dividend (Enbridge on the NYSE), the Canadian figures and the total — before, the U.S.-dollar amount was counted as Canadian dollars.
`portfolio_core.py — _dividend_income_tax · portfolio_db._ACB_FX_SQL · tests/invariants/test_tax_fx_reads_decider.py`

THE HONEST LIMIT

The app's current dividend engine treats Canadian dividends on the *eligible* schedule. Non-eligible dividends (e.g. from some small-business corporations) are handled with their own dividend class, a 15% gross-up and the 9.0301% federal credit, alongside the eligible-dividend treatment. This was previously described here as in progress; it ships.

Not modelled: the character of a Canadian fund's or trust's distribution (interest, foreign income, capital gains, return of capital) — only the slip states it, so those amounts are named and left out of the total rather than estimated, and return of capital is not yet applied to the cost base from a distribution. The foreign credit is the treaty *cap*; CRA credits only the tax actually paid (Line 40500 / Form T2209), so claim what your slip shows. Where a broker books a foreign dividend net of a tax at a rate not on file, the dividend is taxed as booked and the figure *may be understated* — the panel says so. Taiwan withholds 21% on TSMC's ADR dividends unless treaty relief is claimed (TSMC's own 20-F); such a row is grossed up at 21% and credited at the 15% treaty cap, and the excess is named. A fund's or an unknown issuer's row still awaiting an exchange rate is listed as awaiting it and blanks no total. An issuer's country fills on the next data scan; until a holding has one, its dividend is "not assessed".

Chapter 20b

Asset location — and why a placement score is not an instruction to move

Every holding is graded on whether it sits in the right account type. The grade is cheap; acting on it can be expensive, so the advice is priced before it is given.

Two identical holdings can carry very different tax depending on which account holds them. A U.S. dividend payer in a TFSA loses 15% of every dividend to IRS withholding and cannot recover it; the same stock in an RRSP keeps all of it, because the Canada–U.S. treaty exempts RRSPs. `_location_efficiency_score` grades each holding 0–100 against the account-type registry, and `us_dividend_exposure` distinguishes a directly held U.S. payer from one held inside a Canadian-listed fund, where the withholding is lost *inside the wrapper* and no registered plan can reach it.

The issuer, not the currency (audit batch 1, October 2026). U.S. withholding and the RRSP treaty exemption (Canada–U.S. Convention Art. X and XXI(2)) apply to dividends paid by a *U.S.-resident* company. Until October 2026 every U.S.-dollar listing was treated as one: TD on the NYSE was told "Move to RRSP — 15% IRS withholding" (Toronto-Dominion is an Ontario bank; nothing is withheld), and ASML or TSM was told the treaty clears its withholding in an RRSP (the Dutch and Taiwanese tax is lost in any registered account). Placement, the withholding drag, the AI's RRSP list and the migrate-to-RRSP signal now read the issuer from `security_facts.tax_profile`: a U.S. issuer held directly gets the treaty rules; a non-U.S. issuer or a CDR loses its source tax in every registered account; a Canadian issuer on a U.S. listing has none; an issuer not on file is *not assessed*.

Where nothing can honestly be said, nothing is (review, 19 October 2026). A foreign company whose country has no treaty rate on file (the United Kingdom, for example, levies no dividend tax at all) is *not assessed* for placement, rather than told its tax is lost in an RRSP. A CDR in an RRSP reads "withholding in an RRSP not assessed — CIBC's materials conflict": CIBC's own pages say both that an RRSP receives a CDR's full dividend and that the drag is unavoidable unless U.S.-listed shares are held directly; in a TFSA the 15% stands. Units of a U.S. master limited partnership earn business income, which the treaty's RRSP exemption does not reach (Art. XXI(4)), so they are not assessed either. And an RESP, RDSP or FHSA is never called tax-free: an RESP's income is taxed to the student when paid out, an RDSP's investment income when paid out, and an FHSA's is tax-free only on a qualifying withdrawal for a first home.

What is lost is what is withheld, not the treaty cap (cross-model review, 19 October 2026). The treaty rate caps the foreign tax *credit*; it is not the rate a country withholds. Taiwan withholds 21% on TSMC's ADR dividends unless treaty relief is claimed (TSMC's 20-F), against the Canada–Taiwan 15% cap. The withholding drag now uses the withheld rate: a plan with no foreign tax credit (TFSA, RESP, RDSP, FHSA, and an RRSP for a non-U.S. issuer) loses all 21%, and a taxable account loses the 6% above the cap, which the T2209 cannot credit — it used to be skipped as creditable. C$10,000 of TSM at a 2% yield loses C$42 a year in a TFSA and C$12 in a taxable account. A country with no withheld rate on file is not assessed. `portfolio_core.py — withholding_block · security_facts.py — WITHHELD_ABSENT_TREATY_CLAIM, tax_profile`

The more heavily taxed account never ranks higher. In a taxable account a dividend is taxed at your marginal rate and the foreign tax credit only offsets the withholding against that tax, so it always costs at least what a TFSA or RRSP loses. The grade used to rank a U.S. payer higher in a taxable account than in a TFSA (55 against 32) — moving it out of the TFSA raised the grade while roughly tripling the tax (SCHD at 3% and 43.41%: TFSA 0.45%/yr, taxable 1.30%/yr). Every income character is now pinned to the order its hand-calculated tax cost gives (U.S. payer: RRSP 95 > TFSA 32 > taxable 28), and every holding's yield is read, so a Canadian payer earns its dividend-tax-credit note in a taxable account (80, below the TFSA's 85) and a Canadian fund's distribution, whose character only its T3 states, is not graded in a taxable account rather than given a default 68.

The grade answers "is this in the right place?". It does **not**, on its own, answer "should you move it?", and those are different questions with very different price tags. Directing *new* money to the better account costs nothing. Moving an *existing* holding in kind into a TFSA or RRSP is a **deemed disposition at fair market value**: the gain becomes taxable immediately, and a loss is not merely deferred but denied — *"a loss from the disposition of property to … a trust governed by a registered retirement savings plan … is nil"* (Income Tax Act 40(2)(g)(iv)). It is the same rule `portfolio_db._classify_transfer` books as `sell_basis='fmv'` with `deny_loss`.

Until September 2026 the scorer did not know any of that. It received the account, the currency, the dividend yield and the quality label — and never the embedded gain — so its taxable branch returned a flat red *"Move to TFSA/RRSP"* for any well-rated holding, whatever moving it would cost. Measured across the live book on 24 September 2026: **11 of 85 scored positions** carried that advice while sitting on a gain, **C$3,445,706** in total, which would have realised roughly **C$920,977** of immediate tax to shelter future growth — against **C$108,150** of registered contribution room across every account, so most of it was not executable even if someone had tried.

The scorer now receives the gain, and the advice states it. A position with an embedded gain is told what moving it would realise today and pointed at new money instead; a position sitting on a loss is told plainly *not* to move, and why; a flat position is told the move is free if the room exists. The **score is unchanged** in every case — a growth holding in a taxable account genuinely is less

tax-efficient, and that was never the error. Only the instruction moved, from an unpriced imperative to a priced choice. *Source:* `portfolio_core.py — _location_efficiency_score`, `location_scores`; *gated by* `tests/invariants/test_placement_advice_invariants.py`.

One number, one answer. The same scorer feeds the Tax Centre's placement panel and the Strategy report's misplaced-holdings block. Those two disagreed on **33 of 85** holdings until September 2026, because one caller passed the Deep Score verdict as the quality label and the other passed the *account's display name* — so on that path the branches testing for `BUY` and `AVOID` could never fire. Both now read from one row producer, `build_portfolio_rows`, which is also what carries the gain.

Foreign withholding by fund structure (Batch 5). Where foreign dividend tax is taken depends on how a fund is built. CompoundWise follows the typology of Bender & Bortolotti, *Foreign Withholding Taxes* (PWL Capital, 2016; pinned with Bender's 2020 restatement): **Level I** is the tax the stock's home country keeps; **Level II** is a second 15% the US takes when overseas dividends pass through a US-listed fund. Seven structures follow — a US-listed fund of US stocks (A) is treaty-exempt in an RRSP/RRIF; a Canadian-listed fund of US stocks (B) loses the 15% inside the fund in every registered account; a US-listed fund of overseas stocks (C, F) loses Level I everywhere and Level II outside an RRSP; a Canadian fund holding a US-listed overseas ETF (D, G) loses both layers in every registered account; a Canadian fund holding overseas stocks directly (E) loses Level I in a registered account and pays no US tax. A taxable account recovers what is reported on its slips. Each fund's structure is curated from its issuer's own document; dollars use the trailing distribution yield grossed back through the structure, with PWL's measured Level I rates (4.98% developed via the US, 8.04% developed held from Canada, 9.63% emerging) unless the fund's own annual report gives its rate. Funds mixing structures, and uncurated funds, are not assessed. `foreign_withholding.py — holding_drag, remedy · portfolio_core.py — us_dividend_exposure, withholding_block · data/etf_wht_sources.csv`

Chapter 21

The tax-rules config & the CRA auto-draft

Every year-varying tax constant lives in one validated file, and next year's numbers are proposed only when two independent reads agree.

Tax numbers change every year — brackets, credits, contribution limits, the OAS clawback threshold. Rather than scatter these through the code, CompoundWise keeps **every year-varying constant in one file** (`tax_rules.json`) behind a strict validator that refuses out-of-range values. And each January, an assistant *drafts* next year's numbers from official sources — but with a hard rule: it never changes anything on its own.

UNDER THE HOOD · THE CONFIG, AND THE ≥2-SOURCE DRAFT

`tax_rules.json` holds the inclusion rate, U.S. withholding, dividend gross-ups and credits, the CRA statutory bracket schedules (federal and all 13 provinces and territories) with the Ontario surtax, provincial dividend credits, T1135 thresholds, and every contribution limit — each read through a typed accessor whose validator enforces sane ranges. No marginal-rate matrix is stored: each income band's combined rate is computed from those schedules at the band's midpoint (the open top band takes the statutory top rate), and a user's rate is derived on every read from the province and income band on their profile. The income bands are themselves the active year's federal brackets: each band's edges are the federal thresholds and its label states them, so no federal rate change falls inside a band, and a new year's thresholds relabel the bands with nothing typed twice. A band saved under an earlier year's labels is read as the band in the same position — stored labels are never rewritten, so the previous release can still read every profile — and the validator refuses a year whose number of federal brackets differs, where that carry-over would not hold, and a year removed from the file without its labels retired. Every screen that quotes the rate states the income it is taken at, from one server sentence ("Rate at $219,961 (the band's midpoint); your rate may differ inside the band"). **Recorded limitation:** a cell is the statutory rate at its midpoint, not across its band. In 2026, 38 of the 52 non-top cells (13 jurisdictions × 4 bands) contain a provincial bracket threshold — Ontario at $230,000 is 49.53% against the 47.97% cell — and bands are not split at provincial thresholds. Each January the midpoints move with the indexed federal thresholds; where a provincial schedule is carried forward unchanged, a midpoint that crosses its threshold moves the cell with no statutory change (at +2% federal indexation with Ontario's brackets held, both Ontario middle cells rise 1.56 points), which is why the draft flags the provincial schedules for manual verification. The January draft process requires **at least two independent readings that agree exactly** on each figure; anything out of range is rejected, and any structural change (or a move beyond a set threshold) is flagged for manual code review. It *never* writes a file and *never* flips the active tax year — a human applies it deliberately.

`tax_config.py — _validate · tax_rules_fetcher.py — draft_next_year, _crosscheck`

HOW IT CONFORMS · T1135 & DISCIPLINED SOURCING

The config also drives the **T1135 Foreign Income Verification Statement** check (**ITA s.233.3**): a peak-during-the-year test on foreign-property cost above the C$100,000 threshold, excluding registered plans, replayed by the same pooled cost-base engine (and the same Bank of Canada rates and superficial-loss deferrals) that writes the cost base shown beside it. **Foreign property is the issuer's residence, not the trading currency** (s.233.3(1)(c), (d) and (i)): a share of a non-resident corporation, a unit of a non-resident trust, or a right to one such as a CDR counts; Canadian Natural Resources bought on the NYSE, or a Canadian ETF's U.S.-dollar units, does not. A holding whose issuer is not on file is neither counted nor dropped: it is named beside the status — today's holdings and any held during the year, sold ones included — and when those could carry the cost over C$100,000 the status reads **NOT ASSESSED**, never a green "not required" (review, 19 October 2026). Under NOT ASSESSED the year named is the one the unknown holdings could carry over the line — the last completed year first when its in-year peak could (its filing is the one due this spring, the order the known-only test uses), else the current year — and the figure beside it is labelled the *known* foreign cost of that year, never a "peaked at" line drawn from the known holdings alone (cross-model review, 19 October 2026). Until October 2026 every U.S.-dollar holding counted. The replay is dated on the settlement date and takes each day's trades as one step, so a same-day switch (sell one U.S. stock, buy another) is never counted as holding both — until 13 October 2026 the answer could depend on which of the two was entered first. The filing year turns over on Toronto's calendar. The app warns as you approach it, and the status names the tax year it belongs to ("T1135 FILING REQUIRED for the 2025 tax year"). The two-source-agreement rule for adopting new CRA numbers mirrors the professional discipline of never relying on a single unverified figure for a compliance-critical constant.

A joint account at the owner's share (23 October 2026). The user can enter the share of a joint account that is theirs — the part they paid in, which is how CRA has a joint holder report the income (line 12100). The test counts a joint account at that share and, with none entered, in full — the conservative reading, named beside the status. CRA's T1135 questions and answers have each party to a joint venture file on "the cost amount of the property owned by each of them"; applying that to a joint brokerage account is the app's reading, as no CRA text addresses one. The retirement plan reads the same share (half with none entered); the capital-gains cost pool does not yet.

The US estate return (23 October 2026). The executor of a Canadian resident who is not a U.S. citizen files Form 706-NA when the U.S.-situated assets exceed US$60,000

at death, and U.S.-company stock is U.S.-situated wherever it is held (IRS). The Canada–U.S. treaty's prorated credit and small-estate relief (Art. XXIX B paras 2 and 8) are claimed *on* that return, so it is owed even where the tax is not. The Strategy report counts U.S.-company stocks and funds in every account, registered ones included, at the user's share of a joint account (in full with none entered, 26 U.S.C. §2040), and prints a note when the line is crossed or the holdings it cannot count could cross it. A corporate account's holdings are left out: the corporation owns them, and stock of a corporation not organized under U.S. law is property located outside the United States (IRS). It is a disclosure that the return is owed, not an estimate of the tax.

WHY YOU CAN TRUST IT

Because the constants are centralised and validated, a wrong bracket can't hide in a corner of the code, and the whole file must pass validation before it's ever considered live. The auto-draft is advisory only, keeping a human in the loop for anything that touches tax correctness.

PART VI

Planning engines

Beyond picking and tracking: what level of retirement spending your pot supports, how often it lasts, your true net worth, and how concentrated you really are.

Chapter 22

Retirement income — decumulation

The Canadian "how much can I spend, and in what order should I draw down?" engine — every constant read from the one tax config, keyed by tax year.

Saving is only half the problem; spending down without running out is the other half, and in Canada it is tangled with CPP, OAS, RRIF minimums, and the OAS clawback. This engine computes a **sustainable annual spend** in today's dollars, and a **drawdown order** across account types that minimises lifetime tax — with every CRA constant read from the one tax config (`tax_rules.json`), keyed by tax year.

FIGURE 18 · THE DRAWDOWN ORDER, AND WHY

1 · Non-registered
Only the **gain** is taxed, so the tax bill is smallest per dollar spent.

→

2 · RRSP / RRIF
Fully taxable, but drawing early fills low brackets and shrinks the forced minimum at 71+.

→

3 · TFSA last
Tax-free, and withdrawals never count toward the OAS clawback — so it's the most valuable to keep.

OAS clawback watch
If the forced RRIF minimum at 71 would push income over the ~$95k threshold, the plan flags "melt the RRSP down earlier."

Everything is in **today's dollars**. CPP/OAS timing uses the official actuarial adjustment factors on *your own* Service-Canada estimates, not stale "maximum benefit" constants. `Source: decumulation.py — sustainable_spend, drawdown_order, oas_clawback · tax_rules.json (Reg 7308 table)`

UNDER THE HOOD

```
# The pot is projected to the retirement age first — the plan is not
written for today P = balances_today · (1 + r)^years to retirement # REAL
return, so P stays in today's dollars # Sustainable spend = a level,
inflation-adjusted annuity-due on THAT pot PMT = P · r / ( (1 − (1+r)^−n)
· (1+r) ) # P/n when r ≈ 0 # n = horizon − retirement age; horizon =
the FP Canada 25%-survival age; r = real return NET of fees # floored
to the dollar — the MIDDLE case, not a promise: it lands at exactly
zero at the horizon only if # every year earns r; when returns vary it
can run out early (it lasts in roughly half of simulated paths) CPP:
−0.6%/mo if early, +0.7%/mo if deferred OAS: +0.6%/mo if deferred
clawback: 15¢ per $1 of income over ~$95,323 RRIF minimum: the exact
CRA age-factor table
```

The plan apportions taxable income across pots (RRSP fully taxable; non-registered ~half via the inclusion rate; TFSA not income), assumes a modest investment yield so it doesn't report a false $0 clawback for a taxable-heavy retiree, and checks the forced-RRIF-at-71 case explicitly. `decumulation.py — build_plan`

The years before retirement are part of the plan. Until 25 September 2026 every figure was derived from the balances as they stand today, over the drawdown window alone — right for someone already retired, and wrong for everyone else. On production that was 2 of the 2 people with a plan: a 43-year-old retiring at 65 had her sustainable spend understated by 92%, a 45-year-old by 81%, and one worked example moved from “C$5,334 a year short of your target” to “C$26,610 a year ahead of it.” The pot now compounds at the plan’s own real return over those years, and the two figures that are statements about the retirement years move with it — the assumed non-registered investment income behind the OAS recovery tax, and the forced RRIF minimum at 71. What deliberately does *not* move is anything you can tie to a broker statement: the investable total and the account split remain observations of today, printed beside the projection so the two reconcile. **Future contributions are counted** from the annual figure on the client’s own profile — the same one eight other surfaces already read, which this engine simply did not. They compound as an ordinary annuity (level, paid at each year end: $FV = C\cdot((1+r)^n-1)/r$), matching the convention the accumulation simulator already documents so the two engines cannot contradict each other, and they land in the same proportions as the current accounts because nothing says otherwise. Measured on production, one of the two live plans carries C$60,000 a year, which over

twenty years compounds to more than the existing pot grows to — ignoring it was not a conservative rounding, it was most of the answer. What is still *not* modelled: any change to that contribution rate, and fees — the 2026 FP Canada Projection Assumption Guidelines these returns follow state that management fees “must be subtracted to obtain the net return”, and typical fees of 0.5–2.5% would lower every figure on the panel.

UNDER THE HOOD · THE HORIZON AND THE FEES (2026-10-06)

How long the money must last is the client's, not a house 95. The plan runs to the age at which the probability of survival falls to 25%, read from the Probability of Survival table in FP Canada's 2026 Projection Assumption Guidelines, section 4(e) (CPM2014 with improvement scale CPM-B; stored in `tax_rules.json` with its edition and checked row by row against the pinned source). A man or a woman reads their own column; with a partner's birth year on file (or a linked spouse's) the couple column — the age at least one of the two has that chance of reaching — anchored at the younger partner's age. When sex is not on file the longest applicable column is used; between the table's 5-year rows the later age is used. The source's own worked example (a 70-year-old: man 94, woman 96, couple 98) is a test. The sustainable spend is also shown for a horizon five years shorter and longer, as the guidelines recommend.

Projections are net of fees when the fees are known. The real return the client types is the return before fees. One decider (`retirement_sim.return_basis`) deducts the book-weighted fund MER — from the one fund-cost reader, shares and cash at 0% — plus any advisory fee the client declares; the guidelines state fees "must be subtracted to obtain the net return". If any fund's MER is not on file, nothing is deducted and every figure is labelled gross with the reason. The same deduction applies to the return a goal is measured against. A CI gate fails once the guidelines' edition is more than 13 months old.

HOW IT CONFORMS · ITA PART I.2 S.180.2, REG 7308, CPP/OAS FACTORS

RRIF minimum withdrawals use the exact factors of **Income Tax Regulation 7308**; the OAS recovery tax ("clawback") follows **ITA s.180.2** at the current threshold and 15% rate; the CPP early/late and OAS deferral adjustments are the official statutory actuarial factors. Every one of these constants lives in the one tax config, keyed by tax year, so it can be audited against the CRA's published tables; source notes in the config cover some sections, not yet every number.

Chapter 23

Will it last? — the Monte-Carlo simulation

Run your retirement a thousand times against different market lifetimes and count how often the money survives.

A single average-return projection hides the real danger: a bad run of early returns. So the planner simulates **1,000 market lifetimes**, starting from the pot projected at retirement — the same figure the sustainable spend is computed on — and reports the odds of success at your spending target (alongside the sustainable spend's own odds and the spend that lasts in 90% of simulations, solved on the same seeded simulations solved to within C$250 and rounded down, so it errs on the safe side), the tough / median / kind ending balances, the typical age money runs out on failing paths, and — most usefully — the **"bridge years"** warning for when you retire before your benefits start and the portfolio carries everything alone.

UNDER THE HOOD

Volatility is blended from *your* actual asset mix with disclosed assumptions, and the simulation uses a **fixed seed** so the same inputs always give the same answer — reproducibility, not a slot machine. What is checked across the two engines is narrower: at the "sustainable spend" from Chapter 22, the deterministic path — every year earning the assumed real return — lands the pot at zero at 95, and the odds and the spend run on the same projected pot. Under the simulation, where returns vary, that same spend lasts in only roughly half of paths — it is the middle case — which is why the spend that lasts in 90% of simulations is shown beside it. `retirement_sim.py · tests/test_retirement_sim.py`

HOW IT CONFORMS · SEQUENCE-OF-RETURNS DISCIPLINE

Modelling sequence-of-returns risk via **Monte-Carlo** simulation is standard financial-planning practice, and it is presented with its assumptions disclosed and a fixed seed for reproducibility — the honest way to show an uncertain future.

Chapter 24

True net worth & asset mix

The full picture beyond your brokerage — with "strict-kind" math so an unknown category can never silently distort the total.

The headline stays honestly labelled **invested assets** (holdings + cash). Add what lives outside CompoundWise — home, mortgage, held-away savings, pension, loans — and it computes real net worth: **invested + other assets – liabilities**. Values are entered positive; whether an item adds or subtracts is decided by its *kind*, not a sign you could fat-finger, and an unknown kind contributes *nothing* — so a future category can never silently distort today's number. Each outside figure carries how it was valued and the date it was true; both net-worth figures on the web state the share of the outside value (assets and debts together) that is undated or over a year old, decided from the amounts in each state rather than a rounded percent.

A layer above stock-picking sets your **investment policy**: how much of your investable money should be equity vs fixed income vs cash, with a drift band. A breach is *strictly* outside target ± band, so sitting on the edge is inside policy and doesn't nag you. `networth.py · assetmix.py`

The policy is compared with the client's own risk profile. The risk questionnaire scores risk tolerance (the crash answer, experience and purpose) and risk capacity (horizon and income) separately, and the profile is the *lower* of the two (Companion Policy 31-103CP s.13.2; Joint CSA/CIRO Staff Notice 31-368 — reference norms, not a compliance claim). A saved policy holding more than 5 points more equity than that profile, or than the risk tolerance the client declared, is stated as a recorded departure on every surface that carries the policy; the policy is never changed for the client. The band edges, the caps and the 5-point threshold are house decisions recorded in the model register. A plan is a review draft until the client explicitly confirms the six minimum profile fields are current; saving a form does not confirm them. `riskprofile.py · advisor_engine._ips_drift · portfolio_db.confirm_suitability_review`

When a breach has already been shown, the report asks about the policy. CFA Standard III(C) says an investment policy is reviewed at least annually, and a client holding an off-policy book is discussed with: the trade is made or the policy updated. The report asks that question when the book was outside the same policy, in the same class and direction, in the report before (and every report back to the first such one), and the policy has not been saved since; or when the policy was last saved more than 365 days ago. Re-saving the policy restarts the clock. A report inside the

band, or one whose lens failed, ends the chain. An unknown save date triggers nothing. The question never proposes moving the policy to match the book.

A declared cash reserve is outside the policy. Liquidity needs are the first constraint a policy is written under (CFA Standard III(C)). The client declares a reserve in dollars; the mix removes `min(reserve, cash)` from the cash class and from the base every percentage is a share of, so drift, the policy trade and the contribution plan are computed on deployable money only. A reserve larger than the cash on file is reported as a shortfall and is never made up from a GIC or a holding; new contributions rebuild it first. With no reserve declared the value is unknown, not zero: no number moves, and where a cash-over breach draws on savings the client entered, the report states how much of the cash that is and asks. The reserve is never inferred from an item's name or from months of expenses.

> **HOW IT CONFORMS · INVESTMENT POLICY STATEMENT DISCIPLINE**
>
> A target asset mix with an explicit drift band is the core of an **Investment Policy Statement** — the CFA-endorsed anchor for disciplined investing. Enforcing "breach = strictly outside the band" turns a good intention into a testable rule. Manual net-worth entries are private and never touch the investing analytics, keeping the two domains cleanly separated.

Chapter 25

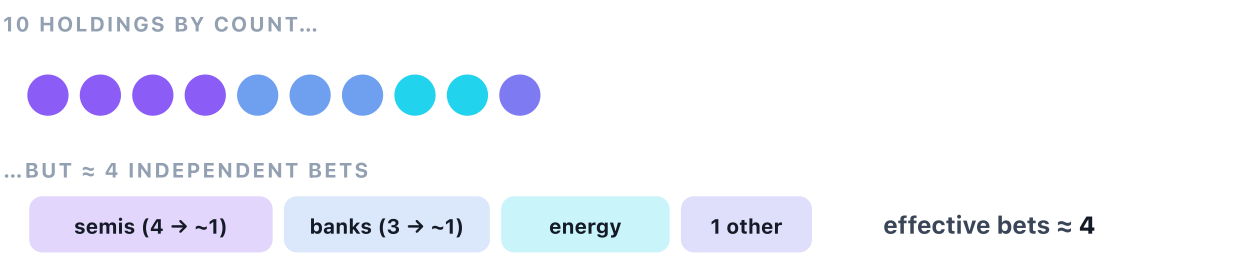
Risk X-Ray — concentration & effective bets

"You own 30 stocks" can still be one bet in disguise; this measures how many independent bets you really have.

Diversification is about independence, not headcount. Ten semiconductor stocks that rise and fall together are closer to one bet than ten. The Risk X-Ray measures your **effective number of bets** — both by weight and by correlation — so concentration you can't see in a holdings list becomes a single honest number.

Concentration limits the client declares. The house limits are 45% of holdings in one sector (ADDs stop at 40%) and 25% in one name. A limit the client declares replaces the house one, tighter or looser, through one decider (`signal_engine.concentration_limit`). A looser limit is a departure from the house view: it is honoured, and while the book sits above the house level the Strategy report states it for each sector or name concerned: the weight, the house limit and the declared one. CFA Standard III(C) treats a request that departs from policy as something to discuss and record, neither refused nor silently accepted. Every size judgement uses the whole position — the same ticker summed across all accounts — and ADDs stop at the name's ADD level, the declared single-name limit × 40/45 (about 22.2% at the house 25%, the same headroom as the sector pair), while Today's concentration card fires at the declared limit.

FIGURE 19 · EFFECTIVE BETS — HEADCOUNT VS REALITY



Highly-correlated names collapse into one effective bet. The chart is not just a read-out — a what-if simulator lets you re-weight and watch diversification change. `Source: risk_concentration.py — effective_names_by_weight, effective_bets`

UNDER THE HOOD

```
effective_names = 1 / Σ w_i^2 # the inverse-Herfindahl of your weights
                              # 10 equal names → 10; one 90% name → ~1.2
effective_bets  = the same idea applied through the correlation matrix
```

The **inverse-Herfindahl** answers "how many equal-sized holdings would be this concentrated?" Feeding the correlation matrix through the same idea turns "how many names" into "how many *independent* bets." A US-dollar holding is converted to Canadian dollars at the same day's USD→CAD close *before* any return is taken, because a Canadian investor's loss includes the currency move; one with no FX history is excluded and named ("no FX history"), never priced as if it were Canadian.

`risk_concentration.py`

HOW IT CONFORMS · PORTFOLIO-THEORY CONCENTRATION MEASURES

The **Herfindahl index** and its inverse are standard concentration measures, and "effective number of bets" via correlation is an accepted portfolio-risk diagnostic. Nothing here is a black box — it's classical portfolio theory made visible for an individual investor.

SINGLE NAMES AGAINST A BUDGET — STATED, NOT ENFORCED

Alongside the effective-bets view, the Strategy report states one blunter number: **what share of the portfolio is held as individual securities rather than funds**, measured by value rather than by count. Value is the point — thirty holdings at 3% each show no large position anywhere and are nonetheless entirely undiversified, which a list of the top five names cannot reveal. The figure appears whenever it exceeds the **20% default budget**, which the client may set for themselves.

The evidence behind the line is one-sided. Bessembinder (*Journal of Financial Economics*, 2018) finds that only **42.6%** of roughly 26,000 US stocks since 1926 beat one-month Treasury bills over their own lifetimes, and that the best **4%** of firms account for the entire net wealth creation of the market — so a concentrated single-name book is, arithmetically, a bet on landing in that 4%. Statman (*JFQA*, 1987) and Domian, Louton and Racine (*Financial Review*, 2007) put the count needed to remove 90% of diversifiable risk at **40–50 names**, not the 15–20 of the older rule of thumb.

The 20% itself is a house decision, not a figure any of those papers publishes, and it is recorded that way in the model register. It is **disclosed, not enforced**: no trade is blocked, no recommendation resized and no plan rejected for exceeding it. Measured on 24 September 2026, all six live books exceed it, from 62.9% to 100.0% — which is precisely why it informs rather than rejects. A limit that every portfolio violates is not a limit.

Currency as a named risk (Batch 5). For a Canadian investor the exchange rate is a source of risk in its own right, so "where your risk comes from" carries a **US dollar (USD/CAD)** line. Its weight is the book's USD exposure: every US-listed holding in full, plus the unhedged US share of each Canadian-listed fund taken from the issuer's own documents (an unhedged S&P 500 fund counts in full, a CAD-hedged one not at all; a fund whose share is not on file is named, never assumed). Each holding is shown in its own currency — $(1 + r_{CAD}) / (1 + r_{USD/CAD})^{share} - 1$ — and the Euler contributions of holdings plus currency still sum to the whole. Each crisis replay also reports the loss in local currencies over the *same* peak-to-trough dates as the CAD loss, and the currency effect (1 + CAD) / (1 + local) − 1 — the same decomposition the performance screen uses. `risk_data.py — usd_share, usd_exposure, local_closes · risk_allocation.py — _with_currency_line · stress_replay.py — _currency_split_of_loss · performance.py — currency_split`

One rule for which crisis windows a stock was in, and a crisis record on every pick (15 October 2026). The book's crisis replay, the stock page's replay and the Strategy Advisor's picks now ask *one* rule (`stress_replay.window_coverage`): a stock is in a window when it has at least

ten stored closes inside it and no stretch of the window longer than 31 days without one — from its start to the first close, between two closes, and from the last close to its end — so a fall is never measured over part of a window and called the window's. The book's replay asks the same of the dates on which every holding it covers has a close, so two holdings' gaps that do not overlap cannot hide the trough between them. A stored close of zero is no price, not a 100% fall, and a holding with no USD/CAD rate through a window is named as that, never as a stock with no price on record. When it is not, the reason depends on what the table can know. Only a stock whose stored history came from a full-history fetch can be said to have **no price on record** before its first recorded close; for any other the window is **price history not stored** — unknown. Neither says "not listed": a data feed's first close can come after the listing. Every single-name diversifier the advisor names, and every name on the report's watch-list, carries its **crisis record**: per window, the largest peak-to-trough fall of its own total-return closes in its trading currency, the dates, and whether and when it regained the peak inside the window — or the reason it was not measured. It is never an estimate and never zero; a beta-times-market figure was considered and declined, because it would present a model's guess beside measured falls. The advisor's model is given the same record and forbidden to state, estimate or imply a fall for a window it was not given. It reads stored closes only; after every stock scan a refresh keeps the latest ranked scan's BUY-or-better names, every held name and USD/CAD stored over their whole history on one adjustment basis, and names every name it could not refresh. A past fall is not a forecast.

A crisis's fall is the crisis's (16 October 2026). Each window runs well past its crisis (the 2022 window to June 2024) so the climb back to the peak can be dated, and the fall used to be searched over all of it: of 715 picks measured in the 2022 window on 15 October 2026, 87 showed a fall whose peak and trough both came after 2022, a slide in 2023 or 2024 under the 2022 label. Each crisis now names its **market low**, the S&P 500's closing low (9 March 2009, 23 March 2020, 12 October 2022), and the fall's peak and trough are searched only from the window's start to that date; the recovery is still searched to the window's end. The same cut applies to the stock page's replay (which draws the low as a dashed line), the book's replay and the picks' crisis records; Full History has no market low and measures every fall. Which windows a stock is in is unchanged. The low is one market-wide date per crisis, so a stock that kept falling after the market turned has its fall measured to the market's low, not its own.

PART VII

The advice layer

Everything so far becomes a short list of decisions. The rule that governs this whole layer: the math decides *what* to flag; the AI only ever explains *why*.

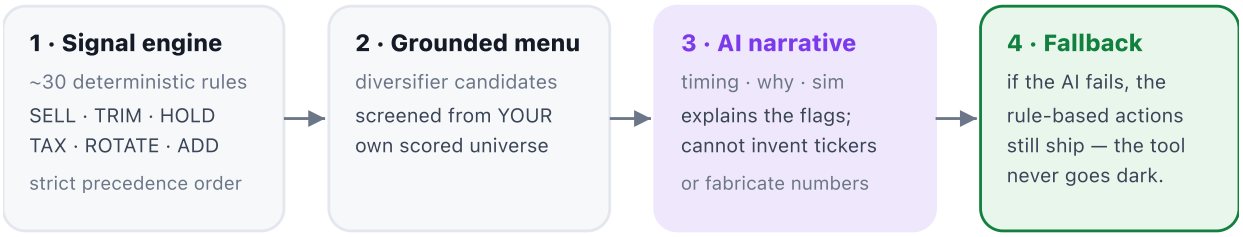
Chapter 26

The Advisor — deterministic signals + AI narrative

A rules engine decides the actions; the AI writes the explanation and can never invent a fact or a ticker.

The Advisor is a **hybrid by design**. A deterministic *signal engine* — about thirty named rules — decides every actionable flag (sell, trim, hold, harvest a loss, migrate to an RRSP, rotate, add). Only then does an AI layer add narrative, timing, and simulation, working from a *grounded* menu of candidates screened from your own universe. The AI cannot conjure a ticker or a number that the deterministic layer didn't hand it.

FIGURE 20 · THE ADVISOR PIPELINE — DETERMINISTIC CORE, AI NARRATION, HARD FIREWALL



Cardinal rule: NEVER sell on valuation alone, and never on rank alone · every SELL/TRIM states its capital-gains conseq

The precedence logic prevents contradictory advice: an exit rule suppresses a trim on the same name; a "hold" only fires when nothing stronger did; an "add" only when there's no sell, trim or hold. Sub-share trims are dropped as no-ops. `Source: signal_engine.py — run_signal_engine · advisor_engine.py — generate_advisor_report`

HOW IT CONFORMS · SUITABILITY, TAX-AWARENESS & AI GOVERNANCE

Two safeguards matter here. First, **suitability**: the rulebook forbids a valuation-only sell, forbids a sale triggered by the Deep Score alone — that score is a standing among whoever was scanned that night, so on its own it would sell a company because OTHER companies were scanned, and a gain or a loss against your own cost is a fact about the purchase, not about the business; something absolute must corroborate it (leverage above a named ceiling, returns or revenue below a named floor, a deep multi-year drawdown), and the reason names which one — and requires every sell/trim to state its tax consequence (quoting only that holding's own gain, scaled by the fraction sold) — advice that ignores the tax bill isn't advice, it's a trap. Every surface that tells the client to trade passes one suitability verdict; an input it needs that is not on file (a stock's industry against declared exclusions, a beta or dividend under an Income objective) makes the action a review, never a pass, and an input that could not be *read* (the profile, the cash reserve, the exclusions, a stored questionnaire) is stated as unreadable, never as blank or zero. Second, **AI governance**: the model is grounded (it can only choose from candidates the deterministic layer screened), metered for cost, and backstopped by a deterministic fallback — a controlled, auditable use of an LLM, not an oracle.

THE HONEST LIMIT

The Advisor is still *not* personalised financial advice you must follow — it's a research co-pilot. The separate "Intelligence" chat is a document-grounded assistant, distinct from this deterministic advisor, and neither takes custody or places trades.

Chapter 27

Strategy & Rebalance

Turn target weights into an exact, tax-aware trade list — and know when you've actually done it.

Give the Rebalance planner your target weights and it produces the precise buys and sells to get there, reusing the *same* money functions as the rest of the app so a rebalance can't drift from your real cost bases. It is careful in three ways that matter: it won't let a target forge fake proceeds, it only counts genuine new contributions against your registered-account room, and it nets the tax with full superficial-loss awareness.

UNDER THE HOOD

- **Sane targets.** Each target is clamped to 0–100% (a negative could forge sell proceeds that bypass the room gate), and the plan is rejected if targets sum past ~100% — cash absorbs the remainder.
- **Dust threshold.** Tiny trades are skipped, with a threshold that scales with book size (a quarter-percent of the portfolio, bounded), so you're not told to trade $6.
- **Room-gated on net-new only.** Only genuinely *new* contributions (a buy beyond what in-account cash and sell proceeds cover) are checked against contribution room; over-room produces a warning or a blocked leg, never a silent over-contribution.
- **Superficial-aware tax netting.** Losses net against gains, but the part of a loss whose ticker is rebought *in this plan or in the real ±30-day window* is denied — the same rule as Chapter 16, asked of the same decider as the order preview (so a declared spouse's rebuy counts). When the plan sells one ticker from several accounts at a loss, the decider is asked once with all of them and the denial is shared in proportion — two sales of 50 against 40 rebought deny 40, not 40 each. A ticker whose check could not run is listed under the plan, and its loss does not offset gains; a sale whose tax cannot be estimated (no exchange rate for today, an unreadable account) shows "tax unknown" and is left out of the total, which says so.
- **All-or-nothing execution.** Every leg is validated, then all legs run inside one transaction — a mid-plan failure rolls the whole thing back.

`api/rebalance.py — rebalance_plan, rebalance_execute`

UNDER THE HOOD · WHAT AN ALLOCATION-POLICY BREACH PRODUCES

Separately from the Rebalance planner, a client who has declared an **investment policy** — target weights for equity, fixed income and cash, plus a drift band — has that policy checked first in every Strategy report, ahead of any single-name idea. Until September 2026 a breach produced a sentence in percentage points and nothing else. It now produces a trade.

- **Every breached class, not just the largest.** The over-weighted class and the under-weighted one are *one trade*. Naming only the bigger side leaves half an instruction: a book 24pp over in cash and 20pp under in fixed income was told "reduce cash toward 5%", and the destination was the second breach.
- **In dollars, and the legs balance.** Every class moves to the target the client declared, so what is added equals what is reduced by construction. A class *inside* its band still appears — it is where the money comes from or goes, and a rebalance naming only breached classes does not add up.
- **The funding source, because it decides the cost.** Money moved out of *cash* is not a disposition and realises nothing; selling equity or bonds realises a gain or loss in a non-registered account and no taxable gain inside a registered plan, whose trust is not taxable (ITA s.146(4) RRSP, s.146.2(6) TFSA, s.146.3(3) RRIF, s.146.6(3) FHSA). The sale is split by account type — registered, non-registered, and any account whose type is not on file, which is named as unknown — and registered dollars are drawn first. New contributions are offered first, since they shift the mix without a sale. What the classes over their targets do not hold is named as new contributions or a sale of holdings outside the policy, not booked to an account of unknown type.
- **The destination is named (22 October 2026).** Each fixed-income and cash leg that takes money names the top screened fund in each listing currency (the fund score's top tier, the client's exclusions applied), or says that none reached it — never a fund from memory. A sale's prose may not send its proceeds into the class that is already over, and a fund being sold for a tax loss is dropped from the screen (ITA s.54). Measured on production before the fix: of 23 sales in reports with a class over policy, 16 named a redeploy ticker and 15 of those named none in the under class.
- **The tax itself is not estimated.** Which lots, in which accounts, at which cost base decides it, and this path is handed none of them — so it reports how much requires a disposition and stops, rather than publishing a plausible figure.
- **Unclassified money is named, never absorbed.** Holdings the fund look-through cannot split into an asset class are excluded from the trade and disclosed, so the

class figures are not quietly inflated.

`advisor_engine.py — _policy_rebalance_trade, _enforce_policy_first · tests/invariants/test_policy_rebalance_invariants.py, test_policy_breach_names_its_destination.py`

UNDER THE HOOD · STAGING A GAIN ACROSS TAX YEARS

A trim sized for risk is not yet timed for tax. A client may state, on their investor profile, the **capital gain they are willing to realize in one tax year**; when a recommended sale would take the year past it, the advice says so and proposes splitting the sale. This is meaningful because a capital gain is reported in the calendar year of the disposition (CRA, *T4037 Capital Gains*, 2025 edition), so the same trim in December and in January lands in different tax years.

- **Absence is unknown, not a number.** No budget on file means the engine stages nothing — it never assumes zero (which would fire on every trim) or unlimited (which would silence every warning). A stated *0* is different: it is a real instruction, “realize nothing this year”, and is honoured.
- **The year-to-date figure comes from the one ledger.** Realized gains are read from the same settlement-dated source the Tax Centre and the year-end summary use, so the advisor and the tax page cannot disagree about the same year. If that total cannot be read, the advice says whether the trim alone already breaks the budget, or says the answer is not knowable — it never checks against a partial picture in silence.
- **A loss never stages.** Realizing a loss reduces the year’s net gain, so it is never deferred for budget reasons.
- **Every surface gets the same answer.** The budget travels with the signals to the Strategy report, the web and mobile lists and the AI chat alike, enforced across the call graph rather than at the call sites that happened to be wrong.

`advisor_engine.py — _annual_gain_budget_cad, _gain_staging_note · tests/invariants/test_advisor_gain_budget_invariants.py`

A PRACTICE CONSIDERED AND DECLINED

Vanguard's threshold-rebalancing rule — trigger at 200 basis points of drift and rebalance *to* 175bp rather than all the way to target — was evaluated for this engine and **not adopted**. It is derived on target-date funds rebalanced at institutional scale, where the live question is when to trim a portfolio that is already *near* its targets and each trade carries a real cost. That precondition does not hold on these books: declared bands are around 5 percentage points and the measured breaches run 20–25 points, four to five times outside them. A smarter trigger would change nothing, because nothing here is near a trigger. What was actually missing was an executable trade, which is what was built instead. Every leg therefore targets the policy the client declared, not a band edge.

Stated plainly because the reverse — adopting a respected institution's rule whose precondition fails here — is how a product accumulates machinery that looks rigorous and decides nothing.

WHY YOU CAN TRUST IT · "DONE" DETECTS ITSELF

A recommendation quietly checks itself off once a matching trade is detected on or after the report date (or, for an exit, once the position is simply gone). This is *derived on read* — it costs nothing and never mutates the stored report, so a paid AI regeneration is never triggered just to notice you acted. The same self-checking logic drives the Tax page and the daily brief. `api/portfolio.py — _action_completed · tests/test_advisor_action_completion.py`

Chapter 28

Action Center, Market Pulse & macro

The front door: a single ranked list of the few decisions that actually need you — each card naming where it came from.

The "Today" page fuses every engine into one ranked list of decisions: signals, the Tax audit's to-dos, off-track goals, asset-mix drift, and concentration checks — de-duplicated so a portfolio-level rule firing on five holdings becomes *one* card naming all five. Buy, sell, trim and add cards come *only* from the rule engine; the AI Tax & Structure Audit contributes structural moves only — MIGRATE, CONSOLIDATE, HARVEST — labelled as such, and every card names its source ("Rule engine", "AI Tax & Structure Audit", "Your goal", "Your concentration limit", "Your target mix"). It costs nothing to build, with a breathing health ring whose count is exactly the number of cards below it. Dismiss is a 30-day snooze, not a mute — a dismissed sell that still fires a month later comes back.

UNDER THE HOOD · MARKET PULSE & MACRO, WITHOUT SPENDING

Market Pulse is a personalised "weather report" built with zero AI and zero new data fetches: an index/FX/VIX/rate tape, then up to three "for your portfolio" lines that cross the macro with *your* actual exposures ("76% of your money is in USD and the loonie is firming…"), driven by transparent, tested thresholds. Every dollar figure comes from your own summary; an unknown simply no-ops rather than printing the word "unknown." A **macro repatch** refreshes the free data hourly and, only on a genuinely volatile tape, triggers at most one rate-limited paid narrative refresh — so it "can never move a cost" when nothing's happening. `market_pulse.py — build_pulse · alert_engine.py — run_macro_repatch`

HOW IT CONFORMS · TRANSPARENCY & COST DISCIPLINE

Every figure on this surface is traceable to the user's own numbers or a named public source, with no hidden model in the decision path — a transparency property, and a cost-governance one (the expensive AI path is rate-limited and event-gated so it can't run up a bill). The daily brief and its dismiss-as-snooze behaviour are locked by tests.

`tests/test_market_pulse.py · tests/test_macro_repatch.py`

PART VIII

Trust & the fine print

The architecture that keeps every number honest — and a plain statement of everything this tool deliberately does not claim.

Chapter 29

How we know it's right

Correctness here is a property of the architecture and a wall of automated tests, not a promise.

Everything in this document rests on two structural commitments. First, **one source of truth per number**: total value, ACB, superficial loss, contribution room, and the FX rate each live in exactly one function that every surface reads — so a figure cannot disagree with itself across the app. Second, **independent verification**: the most sensitive money-math is checked by hand-written "oracles" that re-derive the answer a different way and assert the engine matches.

Guardrail	What it protects
Independent oracles `tests/invariants/`	ACB, superficial loss, contribution room, and the FX chokepoint are each re-implemented from scratch and cross-checked against the production engine.
Golden tests	Performance (TWR/MWR/twin), T1135, withholding, decumulation, net worth, and the retirement simulator are pinned to known-correct outputs that fail loudly on any drift.
Registry parity tests	Deep Score, Winner and Inflection each derive their factor list/weights/labels from one file, and a test forbids a factor being counted twice or a surface counting to a stale total.
Full backend suite	~5,300 tests run green before shipping; a matching CI job runs on every push.
Scan integrity guardrail `scan_integrity.py`	After every scan: freshness, coverage against the universe files, and row-quality *rates* that name the defect classes verification sweeps have actually found — a score on a company with no history, an all-zero equity history, a spike in zero debt histories, unknown sectors. A companion architecture test forbids any per-year history in the metrics builder from spelling a missing cell as 0.0.
Adversarial audits	Recurring "refute-by-default" financial-integrity audits (17 domains) sweep the whole codebase; findings are fixed root-cause-first, substrate before instance.

WHY YOU CAN TRUST IT

The strongest evidence isn't a marketing claim — it's that this document could be written at all. Every formula above cites the one function that owns it, and that function is guarded by a test that would fail the moment the number changed by accident. Legitimacy here is *checkable*.

Chapter 30

Disclosures & deliberate limitations

The most important page in the document: everything CompoundWise chooses not to claim.

A tool earns trust by being honest about its edges. These limits are not caveats bolted on at the end — most are enforced in the code itself.

- **Rank, not probability.** Winner Odds ships a rank and a tier, never a win-percentage, because the data can't yet certify one. A flag in code guarantees no probability is emitted.
- **Assessment, not forecast.** Deep Score rates a business's standing today; it does not predict its price. A point-in-time population test (6 October 2026) found its fundamentals core ranked later peer-relative returns modestly and passed the certification bar narrowly; its valuation inputs, label cut-offs, Canadian listings and the sizing bands were not tested, so it is never called "back-tested" or predictive.
- **Discrimination, not alpha — and not yet validated.** The +64-point tercile spread was measured in-sample on a hand-picked, survivorship-biased basket; it is not an out-of-sample test of the factor ordering and not evidence of live returns. The forward prediction log is accruing the honest evidence.
- **A range, not a target.** Fair value is a blend of models with a disclosed confidence, and it is suppressed entirely when the engine cannot stand behind it — never a price target. There are four reasons it withdraws, and since 2026-09-23 the product names the one that applied rather than blaming disagreement for all of them: no model had the inputs it needs; every model was rejected for sitting implausibly far from the traded price; too few independent model families corroborated; or the families that survived disagreed too widely. A withdrawn estimate publishes no value and no range, and nothing downstream acts on it.
- **No recent trade, no rating.** A stock that has gone more than 5 exchange sessions without a trade on record, counting its scan day, gets no Deep Score, no fair value (no verdict, no range, no single model's value, and no purchase range in Compare), no Inflection rank, no Winner Odds tier and no advisor action, and it is left out of every peer group and sector median so it cannot move anyone else's rank or benchmark; the stock page says why ("no trade on record since 2026-08-14"). A price nobody has paid for weeks is not a price to rate on, and a close printed on a day when no shares changed hands is not a trade. The 5 sessions are our threshold: the SEC's fair-value rule (Rule 2a-5) and the GIPS standards name the condition — a market quotation that is not readily available, a value that is not current — and give no number of days. Added 2026-10-16, after names were found rated on prices weeks old.

- **Holdings-only performance.** The "am I beating the index?" panel covers your holdings with dividends treated identically on both sides and an over-sold twin capped — its scope is disclosed on the panel.
- **A disclosed tax gap.** Non-eligible Canadian dividends carry their own class, gross-up and credit — no longer an approximation, and no longer a gap.
- **Not advice, not custody.** CompoundWise is a research and bookkeeping aid. It does not hold your money, place trades, or issue advice you must follow. Nothing here is legal, tax, or investment advice; confirm tax positions with a qualified professional.

General disclosure. This document describes the *methodology and conformance intent* of the CompoundWise platform. Tax treatments reference Canadian federal rules (the Income Tax Act and CRA guidance) as implemented in the software's dated, sourced constants; provincial rules and individual circumstances vary. Section numbers are provided to locate the governing rule and are not a substitute for professional advice. Every calculation is only as accurate as the data and the assumptions behind it.

APPENDIX A

Glossary — every term, defined

Alphabetical. Each entry gives the plain meaning, why it matters, and a formula where one applies. Every dotted term in the document appears here.

ACB ADJUSTED COST BASE

The CRA's running **average** cost of all your identical shares of a stock — adjusted for commissions, returns of capital and reinvested distributions. Your taxable gain is proceeds – ACB. *Ch 15.*

ACB replay

Recomputing the ACB by processing every transaction in chronological order through one engine, so the number is always correct and consistent everywhere.

Accruals (Sloan)

The gap between reported profit and actual cash generated. Low accruals = earnings backed by real cash = higher quality. A Winner Odds factor.

Alpha

Return above a benchmark. In the performance panel, `alpha = your MWR – the twin's MWR` — the gap to an index, a difference rather than a measure of skill.

Altman Z″-score

Altman's bankruptcy-distress model for non-manufacturers, stored without its constant so the distress line reads 1.10 (Altman's 4.35). Winner Odds uses it as a hard **gate** (a distressed name is capped "Unlikely"), never a scored factor, and does not judge banks, insurers, REITs, regulated utilities or funds with it.

Annuity-due

A stream of equal payments made at the **start** of each period. The "sustainable spend" is solved as a real (inflation-adjusted) annuity-due on the pot projected at retirement.

Benchmark twin

A simulated portfolio that makes the **same** deposits and withdrawals as you, on the **same** dates, into one index fund (XEQT or SPY). The dollar gap is your holdings plus your dividends counted as if reinvested in the same index, minus the twin — a gap, not a measure of skill. *Ch 13.*

Beta

How much a stock moves relative to the whole market. 1.0 = moves with it; >1 swings harder; <1 is calmer. Lower beta earns points in Deep Score's Moat pillar.

CAGR COMPOUND ANNUAL GROWTH RATE

The smooth yearly growth rate that connects a start value to an end value over several years — the standard way to express multi-year growth.

Capital gain / inclusion rate

Profit on selling an asset. In Canada only a fraction — the **inclusion rate** (historically 50%) — is taxable, applied to the year's net gains after losses. *Ch 19.*

CDR CANADIAN DEPOSITARY RECEIPT

A Canadian-listed, currency-hedged receipt for a foreign company's shares (most are U.S. companies; ASML's is Dutch). Its dividends are **foreign income** and it is specified foreign property, though it trades in Canadian dollars. A **phantom-CDR** is a data-feed error that attaches the U.S. company's huge market cap to the tiny receipt price; the app filters these out. *Ch 2, 14.*

Cohort

The peer group a stock is scored against — its sector, and where possible its size tier (large- vs small-cap). Fairness comes from comparing like with like. *Ch 4.*

Cost of equity / ERP

The return shareholders require. `cost of equity = risk-free rate + beta × ERP`, where **ERP** (Equity Risk Premium) is the extra return stocks pay over bonds. Feeds the DCF discount rate.

Crisis record

What a single-name pick actually did through each named crisis window (2008 GFC, 2020 COVID, 2022 rate shock): the largest peak-to-trough fall of its own stored total-return closes from the window's start to that crisis's market low (the S&P 500's closing low), the dates, and whether it regained the peak by the window's end. A window it cannot measure is named — no price on record before its first recorded close, price history not stored, or could not be read — never estimated. A past fall is not a forecast. *Ch 25.*

Cross-sectional

Comparing many companies **at one point in time** (across the "section" of the market), rather than one company over time. All three scoring engines are cross-sectional.

CV COEFFICIENT OF VARIATION

Spread ÷ average. In the fair-value ensemble, a low CV means the models agree (HIGH confidence); a high CV means they don't, so no number is shown.

DCF DISCOUNTED CASH FLOW

Value a business as the sum of all the cash it will generate, each future year discounted back to today's dollars (because a dollar later is worth less than a dollar now). *Ch 10.*

Deemed disposition

The CRA treats you as having sold at market value even though you didn't — e.g. contributing shares in-kind to an RRSP — triggering a taxable gain (a loss is denied). *Ch 17.*

Deflated Sharpe Ratio

A Sharpe ratio corrected for the number of trials and short samples, so a strategy can't look good by luck. Used as the certification bar Winner Odds must clear to show odds. *Ch 7.*

Decumulation

The retirement phase of **spending down** savings (the opposite of accumulation). The planner computes a sustainable spend and a tax-smart drawdown order. *Ch 22.*

Dividend gross-up

Canadian dividends are inflated (×1.38 for eligible) before tax, to reflect the corporate tax already paid — then a credit gives it back. *Ch 20.*

DTC DIVIDEND TAX CREDIT

A federal + provincial credit that offsets the tax on the grossed-up Canadian dividend, roughly cancelling the double tax between company and shareholder.

Drawdown

A peak-to-trough decline. In performance, the worst fall from a high; in retirement, "drawdown order" means which accounts you spend from first.

EDGAR / XBRL

The U.S. SEC's public filings database (EDGAR) and the machine-readable tagging inside filings (XBRL). The source of a decade of official financials used to test the models. *Ch 2.*

Effective bets / inverse-Herfindahl

`1 / Σ(weight_i²)` — "how many equal-sized holdings would be this concentrated?" Applied through correlations, it counts your **independent** bets, not just your holdings. *Ch 25.*

Eligible vs non-eligible dividend

Eligible dividends (from generally-taxed corporate income) carry a bigger gross-up and credit; non-eligible (e.g. small-business) dividends carry a smaller one.

Ensemble

Combining many independent models into one estimate. The fair-value ensemble blends 14 models across 4 families and only reports a number when they agree.

EPV EARNINGS POWER VALUE

The value of a company's current sustainable earnings assuming **zero** growth — a deliberately conservative valuation floor.

Federal abatement (Québec)

A reduction of federal tax for Québec residents (who pay a distinct provincial system); it lowers the federal dividend credit in the calculation.

FCFF FREE CASH FLOW TO THE FIRM

The cash a business generates for **all** its capital providers (debt + equity), before financing. The DCF models discount true FCFF, then subtract net debt to reach equity value.

FCF yield

Free cash flow ÷ market value — like an interest rate the business "pays" you at today's price. A Deep Score valuation factor.

Foreign tax credit (s.126)

A Canadian credit for tax already withheld abroad (e.g. 15% on U.S. dividends), so the same income isn't taxed twice. *Ch 20.*

FX

Foreign exchange — here, the USD↔CAD rate. Every conversion passes through one guarded gate with a 0.85–2.0 plausibility band. *Ch 18.*

GIPS GLOBAL INVESTMENT PERFORMANCE STANDARDS

The industry rulebook for calculating and presenting returns fairly. TWR is its prescribed method for benchmark comparison. *Ch 13.*

Gordon Growth / DDM

Dividend Discount Model: value = next dividend ÷ (required return – dividend growth). Meaningful only for steady dividend payers; one of the ensemble's models.

Graham Number

A classic Benjamin-Graham valuation floor from earnings and book value — one of the ensemble's asset/earnings-power models.

Herfindahl index (HHI)

The sum of squared weights — a concentration measure. Its inverse gives the "effective number" of holdings or bets. *Ch 25*.

IC INFORMATION COEFFICIENT

The rank correlation between a model's scores and what actually happened next. A modest positive IC (~0.05) is genuinely good; a huge one usually signals overfitting. *Ch 7*.

In-kind transfer

Moving shares (not cash) between accounts. Tax-free between non-registered accounts, but a deemed disposition when contributed to a registered plan. *Ch 17*.

Income Tax Act (ITA)

Canada's federal tax statute. Section references (s.38, 40(2)(g)...) point to the exact rule a calculation implements.

Last trade on record

The most recent day, on or before a scan, on which a stock actually traded — a daily bar with shares changing hands, not just a printed close. A stock that has gone more than 5 exchange sessions since its last trade, counting its scan day, is not rated anywhere, and says so. *Ch 30*.

Look-ahead bias

Accidentally using information that wasn't known yet when testing a model — which flatters results. The backtests slice data to each point in time to avoid it. *Ch 7–8*.

LLM LARGE LANGUAGE MODEL

The AI that writes narratives and drafts. In this app it is always grounded, cost-metered, and firewalled — it explains, it never decides the numbers.

MAD MEDIAN ABSOLUTE DEVIATION

The median of how far each value sits from the median — a spread measure that a few outliers cannot inflate. ×1.4826 makes it comparable to a standard deviation. *Ch 3*.

Margin durability

Whether a company's profit margins are holding up versus their own history — a sign of a real competitive moat.

Margin of safety

`(fair value – price) / fair value` — how much cheaper than its estimated worth a stock trades. A cushion against being wrong. Drives a sizing bump. *Ch 10–11*.

Monte-Carlo simulation

Running a plan across many random market lifetimes to see how often it succeeds — revealing risks that a single average hides. *Ch 23.*

MWR / XIRR MONEY-WEIGHTED RETURN

The single return rate that makes your dated cash flows net to zero — the return your dollars actually earned, timing included. *Ch 13.*

OAS clawback

The Old Age Security recovery tax: 15¢ of OAS is clawed back per $1 of income above a threshold (~$95k). TFSA withdrawals don't count toward it — which shapes the drawdown order. *Ch 22.*

P/E · P/B · P/S · PEG · EV/EBITDA

Valuation multiples: price vs earnings, book value, sales; PEG divides P/E by growth; EV/EBITDA compares enterprise value to operating cash earnings. Lower usually = cheaper.

Percentile rank

The share of the peer field a value beats — a self-calibrating 0–100 score. Deep Score ranks each pillar this way, and then ranks the composite the same way across the whole universe, so the full scale is always used and a label cliff means the fraction its spacing names. A consequence worth knowing: the headline score is your standing across the universe, so it is not the sum of the pillar bars, which show your standing inside your own sector-and-size peer group. *Ch 3–4.*

Piotroski F-score

A 0–9 checklist of nine financial-health tests (profitability, leverage, efficiency). More passes = healthier fundamentals. A Winner Odds factor.

Purged / name-grouped cross-validation

A backtest that tests on companies held out entirely (grouped by name) so the model can't "peek" — the honest way to estimate out-of-sample skill. *Ch 7.*

Residual income

Profit earned above the cost of the capital used to earn it — a valuation model in the ensemble's asset family.

Robust-z

A z-score built from the median and MAD instead of mean and standard deviation, so outliers can't distort it, and clamped to ±3. *Ch 3.*

ROC RETURN OF CAPITAL

A distribution that is a return of your own money, not income. It lowers your ACB (and triggers a gain if ACB hits zero). *Ch 15.*

ROE · ROIC

Return on Equity / Invested Capital — profit per dollar of owners' money (ROE) or of all capital (ROIC). The signatures of a quality business; heavy in Deep Score.

Rule of 40

For growth companies: revenue-growth% + profit-margin% should exceed 40 — balancing "fast" against "profitable." A Winner Odds factor.

Sequence-of-returns risk

The danger that a market crash **early** in retirement (while withdrawing) does far more damage than the same crash later. Monte-Carlo reveals it. *Ch 23*.

SSOT SINGLE SOURCE OF TRUTH

The core design rule: every number is computed in exactly one place, and every screen reads that place — so figures can never disagree across the app.

Superficial loss

Canada's anti-wash-sale rule: sell at a loss and rebuy the same security within ±30 days (any account) and the loss is denied — but added to the new shares' ACB if they are still held, outside a registered plan, when the window closes. *Ch 16*.

Survivorship bias

Testing only on companies that survived, which flatters results by ignoring failures. The forward logs deliberately include delistings to avoid it. *Ch 7*.

T1135

The CRA's Foreign Income Verification Statement, required when foreign-property cost exceeds $100,000 during the year. Foreign property means a **non-resident issuer** (not a U.S.-dollar listing): a Canadian company bought on the NYSE does not count. The app runs a peak-during-year test and warns early. *Ch 21*.

Fund distribution (T3)

What a Canadian ETF, mutual fund trust or REIT pays out. Its character — interest, foreign income, capital gains, return of capital, eligible dividends — is set on your T3/T5 slip, so the app names it rather than taxing it as an eligible dividend. *Ch 20*.

Issuer domicile

The country a company or fund is resident in — which decides its dividend's tax treatment, U.S. withholding and the T1135. Decided from the issuer — overrides, the curated fund list, then SEC EDGAR's record of where the company is incorporated, the market feed's country only where EDGAR has none — never from the currency it trades in; not assessed when the two disagree or neither has one. *Ch 20, 20b*.

Tercile / decile spread

Split the ranked names into thirds (terciles) or tenths (deciles); the spread is the forward-return gap between the top and bottom group — a discrimination test. *Ch 7*.

Terminal growth

The perpetual growth rate assumed after a DCF's forecast period (capped at 2.5% — no firm outgrows the economy forever).

TFSA · RRSP · RRIF · FHSA · RESP · RDSP · LIRA · LIF

Canadian registered accounts with different tax treatment and contribution rules. The app knows each one's rules (tax, room, withdrawal) from its account registry.

TWR TIME-WEIGHTED RETURN

Return with deposit/withdrawal timing stripped out — what your holdings did against the index (a difference, not a measure of skill), and a GIPS-consistent time-weighted methodology for comparing against an index. *Ch 13*.

WACC WEIGHTED AVERAGE COST OF CAPITAL

The blended required return on a firm's debt and equity — the discount rate in the DCF. Floored at 7.0% (terminal growth + 4.5 points) and nudged by Deep Score quality. *Ch 10*.

Winsorize

Pulling extreme values in to a set percentile (here the 5th/95th) before analysis, so freak data points can't dominate a ranking. *Ch 3*.

Z-score

How many "typical spreads" a value sits above or below the middle of its group. Positive = above average, negative = below. The building block of the scoring engines.

APPENDIX B

Formula index & source map

Every headline formula in the document, with the single source-of-truth file that owns it — so any number can be traced straight to the code.

What	Formula (essence)	Source (SSOT)
Robust-z	`clamp((clip(x)−median)/(MAD·1.4826), ±3)·dir`	`cross_sectional_stats.py`
Percentile rank	`count(peers ≤ x) / n`	`cross_sectional_stats.py`
Deep Score pillar	`PILLAR_MAX × percentile(Σ w·z / Σ w)`	`deep_score_engine.py`
Deep Score composite	`100 × percentile(Σ pillar_score)`, ranked across the whole day's universe	`deep_score_engine.py`
Value×Quality quadrant	`quality = class_quality_z>0.5 ∧ score≥55 ; cheap = mean(class valuation z)>0`	`deep_score_engine.py`
Winner composite	`Σ_core(w·z) / Σ_core w` · abstain <6 factors	`winner_engine.py`
Winner pillar (radar)	`clip(50 + 25·mean(z), 0, 100)`	`winner_engine.py`
Rank tier	`≥.75 Strong candidate · ≥.50 Watch · else Unlikely` (the .90 cut removed 2026-09-28)	`winner_model.py`
Inflection score	`round(100 · Σ_scored earned / Σ_scored max)`	`inflection.py`
Growth acceleration	`recent YoY − earlier-half median (pp)`	`inflection.py`
Fair-value blend	price-sanity → outlier-trim → family means → mean; CV→tier	`valuation/aggregate.py`
DCF discount rate	`max(risk_free + β·ERP + quality_adj, 7.0%)`	`valuation/models/dcf.py`
Margin of safety	`(fair_value − price) / fair_value`	`valuation/aggregate.py`
Position size	Deep-Score tier → band; +1 tier if MoS ≥25%; cap 25%	`sizing.py`
Total equity	`Σ value_cad + cash_cad`	`advisor_engine.py`
Since-inception return	`(equity − net_deposits) / net_deposits`	`advisor_engine.py`

What	Formula (essence)	Source (SSOT)
MWR / XIRR	`0 = Σ flow / (1+r)^(days/365.25)`	`performance.py`
TWR (chained)	`level ×= max(0, ($V_t$−$F_t$)/$V_{t-1}$)`	`performance.py`
ACB — buy/sell/ROC/reinvest	commission→cost; per-share ACB invariant to sells; ROC↓; reinvest↑	`portfolio_db.py`
Superficial denied qty	`min(sold, Σ per holder least(reacquired ±30d, still-held at window end))`	`portfolio_db.py`
Allowed loss	`realized_gain + superficial_denied`	`portfolio_core.py`
Capital-gains tax	`max(net gain,0) · inclusion(year) · marginal`	`portfolio_core.py`
Canadian dividend tax	`grossed·marginal − grossed·(fed_DTC·(1−abatement)+prov_DTC)`	`portfolio_core.py`
Foreign dividend tax	`max(foreign·marginal − foreign·withholding, 0)`	`portfolio_core.py`
Sustainable spend	`P·r / ((1−(1+r)$^{-n}$)·(1+r))` — floored	`decumulation.py`
OAS clawback	`15% × (income − ~$95,323)`	`decumulation.py`
Effective bets	`1 / Σ $w_i^2$` (weight & correlation forms)	`risk_concentration.py`
FX plausibility band	`0.85 ≤ USD/CAD ≤ 2.0` ; strict-historical defers	`fx_rates.py`

APPENDIX C

Regulatory & Standards Conformance Matrix

The credibility case at a glance: every engine and money calculation, the rule or professional standard it answers to, how the code enforces it, and the test that locks it. CRA / ITA = Canadian tax law · CFA / GIPS = professional / methodological standard.

20
engines & calculations mapped

10
CRA / Income Tax Act citations

11
CFA / GIPS standards answered

100%
rows locked by a named test

Engine / calculation	Conforms to	How the code enforces it	Locked by
Deep Score	CFA / methodology factor investing; robust statistics	Peer-relative (sector+size) robust-z; one factor registry; percentile self-calibration; confidence disclosed; no forecast	`registry-p`
Winner Odds	CFA / methodology multi-factor models; López-de-Prado overfitting controls	Rank-only until certified; `CALIBRATED=False` nulls odds; in-sample rank statistics on point-in-time statements with filed-date entry, a real Deflated Sharpe over a recorded trial log, and a t ≥ 3 bar; no external base rate has been established	`backtest_v`
Emerging / Inflection	CFA / methodology earnings-momentum & revision anomalies; accruals quality	Scored-only normalization; cyclicality guard; upward-only revisions; walk-forward reads discrimination not alpha	`inflection time popul the 2026-1 Sharpe 0.4`
Fair-value ensemble	CFA valuation DCF, multiples, residual income, DDM, EPV	14 models / 4 families; price-sanity + outlier trim; suppress on high dispersion; ≥2 families required	`valuation/`
Position sizing	Prudent diversification risk budgeting	Conviction→band from shared cliffs; hard 25% single-name cap; explicit rationale; no cheapness-rescue	`tests (siz`

Engine / calculation	Conforms to	How the code enforces it	Locked by
Total value & return	GIPS spirit broker reconciliation	One equity SSOT; since-inception vs net deposits; basis relabelled when unknown; parity contract	`test_netwo`
TWR / MWR / twin	GIPS / CFA performance standards	Time-weighted for benchmark; money-weighted for the investor; same-cashflow twin; scope disclosed	`test_perfo`
ACB	ITA s.47 average-cost, identical properties	Chronological replay; commission→cost; ROC↓; reinvest↑; one display SSOT	`test_acb_c`
Superficial loss	ITA 40(2)(g)(i) · 53(1)(f)	Sold capped by each holder's least(reacquired, still-held); every account counts; roll-in to substituted shares; permanent-in-registered; later-of timing	`superficia`
In-kind transfers	Deemed disposition · s.39(1.1)/261	Booking table from account types; loss denied into registered; gain always decided in CAD	`tests (tra`
Capital-gains tax	ITA s.38 · s.39	Inclusion on year's net; per-year inclusion rate; FX component preserved; zero in registered	`test (real`
Dividend tax	ITA s.82 · s.121 · s.126 · s.89(1)	Issuer classification (security_facts.tax_profile, never the currency); eligible gross-up + fed/prov DTC less Québec	`test_t1135`

Engine / calculation	Conforms to	How the code enforces it	Locked by
		abatement; foreign credit at the source country's treaty rate, none when unknown; fund distributions named, not taxed as eligible	
T1135	ITA s.233.3	Peak-during-year foreign-cost test; registered excluded; early warning	`test_t1135`
Tax-rules config	CRA constants disciplined sourcing	One validated file; ≥2 independent agreeing reads to draft; never auto-applies; structural moves → manual review	`tax_config`
Contribution room	TFSA/RRSP/FHSA rules	Room from actual contributions not market value; per-key pooling; FHSA carry-forward capped; the FHSA's close year (ITA s.146.6(1): December 31 of the earliest of the 15th anniversary of opening the first FHSA, age 71 and the year after the first qualifying withdrawal, a planned purchase standing in for the withdrawal; "at the latest" when a missing input could make it earlier); RRSP $2k grace; RESP limit per child across family and individual plans (unassigned family money → undetermined); first-60-day RRSP	`tests (cor` `test_schem`

Engine / calculation	Conforms to	How the code enforces it	Locked by
		contributions counted once (the prior-year notice has them; without its unused line the check is incomplete, not zero); a later TFSA withdrawal reduces an excess, and the tax already owed — 1% of the highest excess each month (ITA 207.02) — is shown. Baselines as CRA states them: TFSA room as of Jan 1 + its year (a figure with no year is Jan 1 of Toronto's room year); RRSP deduction limit + the notice year, refused without one and never counted without one; FHSA from its opening year. The bare room figures older mobile apps saved (read by nothing) are refused, and schema v19 carried those on file: TFSA → Jan 1 room of the Toronto year it was typed, RRSP → the limit with its year left for the client, FHSA dropped; an existing baseline is never overwritten	
Decumulation	Reg 7308 · OAS s.180.2 · statutory CPP/OAS factors	Exact RRIF factor table; clawback at threshold; official actuarial adjustments; sustainable spend floors	`test_decum`

Engine / calculation	Conforms to	How the code enforces it	Locked by
Monte-Carlo sim	Planning practice sequence-of-returns	1,000 paths; fixed seed; vol from actual mix; starts from the projected pot the spend is computed on; deterministic path at the sustainable spend lands at zero at the client's FP Canada 25%-survival horizon; 90%-success spend shown beside it (panel and Strategy report); odds at the horizon ±5 years and the return ±1 point	`test_retir`
Risk X-Ray	Portfolio theory concentration measures	Inverse-Herfindahl by weight and correlation; transparent, no black box	`tests (ris`
Advisor	Suitability · AI governance	Deterministic signals decide actions; no valuation-only sell; tax note per sell; grounded, metered, fallback LLM	`test_signa`
Rebalance	Room + superficial · IPS	Targets clamped; room-gated on net-new; superficial-aware netting; single-transaction rollback	`test_rebal`

CompoundWise — The Methodology. Every figure, formula, threshold and regulatory citation in this document traces to a single source-of-truth function in the codebase, guarded by an automated test. It documents the platform's methodology and conformance intent; it is not legal, tax, or investment advice.

`Primary sources: deep_score_engine.py · winner_engine.py · inflection.py · valuation/ · performance.py · portfolio_db.py · portfolio_core.py · decumulation.py · tax_config.py ·`

tax_rules.json · fx_rates.py · signal_engine.py · advisor_engine.py · api/rebalance.py · risk_concentration.py · tests/invariants/