DAI / RBC – Update Bias (Robust EWMA) v1.5

Detailed Design Document

Version 1.5 | Status: Live – Under Protected Architecture Mode v5.9

Maintainer: Simon Angell | Advisory Role: ChatGPT-5

Storage Path: C:\Users\simon\OneDrive\ChatGPT\Projects\DAI\Project Specific Governance Docs\Detailed Designs

1 Purpose and Scope

Defines the complete functional design of the RBC – Update Bias (Robust EWMA) blueprint script. The script implements a robust exponential weighted moving average (EWMA) for bias correction between actual and estimated energy totals. It ensures deterministic, idempotent, and bounded bias updates within the DAI + RBC learning framework. This design is the single architectural authority for all bias-producer variants.

2 Functional Overview

Applies an EWMA filter to the daily error (Actual – Estimate), clamps excessive errors using a ratio of the estimate and a fixed floor (0.5 kWh). Caps overall bias magnitude, rounds to 2 d.p., and writes the result to a bias helper. Records the processed day key in an input text helper to guarantee one update per day. Provides selectable logging levels for audit transparency.

3 Interface Definition

Parameter	Type / Domain	Default	Description
actual_entity	entity	—	Source of actual total (kWh)
estimate_entity	entity	—	Forecast / estimated total (kWh)
bias_helper	input_number	—	Output bias store
last_period_text	input_text	—	Stores last processed date
half_life_days	number 1–30	4	EWMA half-life
clip_ratio	number 0.1–2.0	0.5	Error-clip multiplier × estimate
max_abs_bias	number 5–200	60	Absolute bias limit

			(kWh)
require_nonzero	boolean	true	Skip if both actual = estimate = 0
log_level	select {quiet, normal, verbose}	normal	Controls logging granularity

4 Algorithm Logic

Input Validation → Robust Clipping → EWMA Update → Cap and Round → Idempotence Guard.

Canonical order: Clip → EWMA → Cap → Round(2).

5 Operational Flow

1) Check already processed → skip.
2) Validate inputs → skip if unavailable.
3) Apply maths.
4) Write bias and period.
5) Log result.

6 Concurrency and Safety

Mode: single ensures serial execution per script entity. Residual race conditions occur if multiple instances share helpers. Mitigation: Early lock (H1) and unique helpers (H2). Combined approach ensures no cross-thread drift.

7 Mathematical Specification

Symbol	Meaning	Formula / Range
y	Actual total	float(kWh)
ŷ	Estimated total	float(kWh)
e	Error	y − ŷ
c	Clip bound	max(0.5, ŷ × clip_ratio)
e*	Clipped error	clamp(e, −c, +c)
λ	Smoothing factor	1 − 0.5^(1/half_life_days)
b_prev	Prior bias	current helper state
b_raw	EWMA output	λ·e* + (1−λ)·b_prev
b_capped	Capped bias	clamp(b_raw,

		±max_abs_bias)
b_new	Final bias	round(b_capped, 2)

8 Logging and Diagnostics

quiet → minimal logs; normal → final summary; verbose → full trace. Log template includes y, ŷ, e, e*, λ, prior.

9 Acceptance Test Matrix

Covers daily update, already processed, unavailable inputs, both-zero, large errors, cap enforcement, concurrency race, and day boundary cases.

10 Compliance Map

All Protected Architecture Mode rules satisfied; writes limited to helpers, Visual-Editor safe syntax, and full logging parity.

11 Dependencies

Home Assistant Core ≥2025.10.2; Input helpers; Script blueprint in mode: single.

12 Change History

Version	Date	Summary
1.0	2025-08-15	Initial prototype.
1.3	2025-09-10	Added clip ratio and caps.
1.4	2025-10-10	Introduced idempotence via period key.
1.5	2025-11-01	Live baseline; robust EWMA maths, concurrency notes.