Newborough Warren

Interactive Web Tools

Technical Note

Model Equations · Data Architecture · Rendering

Hollingham (2026)
May 2026

Part A

Groundwater Flooding Forecaster

A1. Overview

The Groundwater Flooding Forecaster (forecaster.html) is a single-page application built from a Jinja-style template (forecaster_template.html, 1187 lines) and a JSON data bundle injected by Script 11b (11b_spatial_thresholds.py, 1933 lines). From the May 2026 simplification onwards, the forecaster presents only the report's cluster-block equations (Tables 6, 7, and 10) with inline value substitution. The per-well SSM iteration path (ssmIterate, horizonMonths, FORECAST_SOURCE tabs, and per-well SSM/P_flood coefficient exports) has been removed.

A2. Data Bundle Structure

Key	Contents
cluster_coeffs	Per-cluster: label, peak_month, trough_month, P_flood slope A and intercept B, P_clim_total_mm, horizon_months, monthly_clim (12-month depth climatology). Five clusters: C1–C5.
block_tf	Seasonal transfer-function coefficients (b1, b2, c, R^2) for each geographic block, split into winter and summer sub-models. Each block lists its member clusters.
P_clim / PET_clim	Monthly climatological precipitation and PET (mm), keyed 1–12, from RAF Valley 2005–2026 means.
winter_climatology_mm	Mean Oct–Mar cumulative rainfall (~521 mm), the denominator for λ.
wells	Array of well objects with: name, display_name, E/N (OSGB), ground_elev, cluster, nearest_cluster_only, default_h_prev, default_h_max, and (from the simplification) monthly_clim (per-well 12-month depth climatology), trough_month, and peak_month.
base_layer	Map extent (OSGB: E 240100–243900, N 362100–365900 — a 3800 m × 3800 m square), Base64 hillshade PNG, and KML feature polylines with styling.

A3. Model Equations

The forecaster now presents only the three report-table equations. Per-well SSM iteration has been removed.

A3.1 Block transfer functions (Tables 6 and 7)

Forecast 1 (winter peak, Table 6):

$$h\_peak = \beta_1 \cdot P\_winter + \beta_2 \cdot h\_min(summer) + intercept$$

Forecast 2 (summer minimum, Table 7):

$$h\_min = \beta_1 \cdot P\_summer + \beta_2 \cdot h\_max(winter) + intercept$$

R^2 values range from 0.40 (Lake Edge winter) to 0.91 (Forest summer). When $R^2 < 0.50$ the forecast card displays a low-explanatory-power caveat. P_winter is the climatological Oct–peak_month total scaled by λ; P_summer is the climatological Apr–Sep total (always at $\lambda = 1.00$).

A3.2 P_flood linear form (Table 10)

Forecast 3 (cumulative rainfall to slack floor):

$$P\_flood\ (mm) = A \cdot d + B$$

where d is depth below ground (m, positive), A is slope (mm/m), and B is intercept (mm). The result is normalised as $\lambda = P\_flood / P\_clim\_total$.

A4. Cluster and Block Mapping

Cluster	Label	Block TF	Peak	Horizon
C1	Lake Edge	Lake_Edge	Feb	5 mo
C2	Dune	Eastern_Block	Feb	5 mo
C3	Western Residual	Western_Block	Mar	6 mo
C4	Main Forest	Forest	Mar	6 mo
C5	Coastal Forest	Coastal_Forest	Mar	6 mo

A5. Per-Well Monthly Climatology

Script 11b reads 01_wells_clean.csv and computes a 12-month depth climatology per well (mean depth below ground for each calendar month). These are injected as monthly_clim, trough_month, and peak_month fields on each well object. The cluster_coeffs also receive their own monthly_clim (cluster-average depths), trough_month, and peak_month. The forecaster's renderWellMeta function renders two stacked tables — per-well first, then cluster — each labelled by the entity's own trough and peak months (which may differ).

A6. Two-Input Architecture

The sidebar contains two separate depth inputs, each driving a distinct forecast. This split resolves an ambiguity in earlier versions where a single "observed depth" field was silently used as both the summer minimum (for Forecast 1) and the current depth (for Forecast 3), which confused users whose today's reading was not their summer minimum.

- current-depth-input: feeds Forecast 3's d. Default: coeff.monthly_clim[currentMonth] (cluster's long-term depth for the current calendar month), falling back to default_h_prev if no monthly climatology.
- summer-min-input: feeds Forecast 1's h_min. Default: default_h_prev (cluster's long-term summer minimum).

Both inputs fire a shared onDepthInput handler that calls renderForecasts. selectWell() populates both from cluster defaults when the user picks a well. Forecast 2 continues to use default_h_max (cluster-mean winter peak) with no user input.

A7. Timing Note Logic

Forecast 1 includes an amber timing note when the current calendar month falls before the cluster's trough_month (the month in which the summer minimum is typically reached). The inWinterSeason() function walks the calendar from trough_month towards March; if the current month is reached before March, the minimum has been observed. If not, the note reminds the user when this year's minimum is typically reached and that the summer-min input currently holds the long-term default.

A8. Live Met Office Integration

On initialisation, loadLiveData() fetches RAF Valley station data. The parser handles estimated values (asterisk-suffixed) and missing data ("---"). buildLiveBanner() computes Oct–Mar cumulative totals for current and prior hydrological winters, deriving λ = observed / climatological. The display adapts by calendar month (Oct–Dec emphasises the previous winter; Jan–Apr the current; May–Sep the most recently completed).

If the fetch fails, a manual fallback modal accepts pasted valleydata.txt and parses it identically.

A9. Map Rendering

The map uses a square SVG viewBox (1000 × 1000) matching the square base-layer extent (3800 m × 3800 m, OSGB E 240100–243900, N 362100–365900). The project() function linearly maps OSGB coordinates to SVG space with 30 px padding. preserveAspectRatio is set to xMidYMin meet, top-aligning the content. The legend overlays the top-left corner; the rainfall slider overlays the top-right.

Rendering layers: (1) Base64 hillshade PNG at 85% opacity, (2) KML polylines/polygons clipped to the plot area, (3) well dots (r=10, r=14 when selected) coloured by P_flood category, re-rendered on every slider change.

Note for maintainers: MAP_W and MAP_H control only the SVG viewBox, not the geographic extent. The geographic extent comes from DATA.base_layer.extent. If one changes without the other, project() independently stretches each axis, distorting the hillshade. Keep viewBox aspect = base-layer aspect. Currently both are square (1:1).

A10. Resizable Panels

The main layout uses CSS flex with 6 px draggable gutters between the sidebar, map, and forecast panel. setupResizer() attaches mouse and touch handlers; dragging sets the adjacent panel's flex-basis within a 200–900 px range. Widths persist via localStorage (keys nb-panel-width-sidebar and nb-panel-width-panel). On screens narrower than 960 px, panels stack vertically and gutters are hidden.

A11. Build Provenance

Script 11b assembles block transfer-function coefficients (Tables 6/7/10), cluster metadata, per-well monthly climatology (from 01_wells_clean.csv), hillshade, KML features, and RAF Valley climatology into a single .html file from the forecaster_template.html template. Google Fonts (Libre Baskerville, Source Sans 3, JetBrains Mono) are the only external resource; the tool degrades gracefully without them.

Citation: Hollingham, M. (2026). Hydrogeological Dynamics, Behavioural Clustering and Management Intervention Analysis at Newborough Warren Coastal Sand Dune Aquifer, Wales.

Part B

Hydrological Scenario Viewer

B1. Overview

The Scenario Viewer (scenario_viewer.html) is generated by Script 19 (19_spatial_groundwater.py, v2.6.0). All computation is client-side JavaScript; no server requests are made after the initial page load.

B2. Physical Constants

Parameter	Value and source
Hydraulic conductivity K	6 m/day (Betson et al. 2002)
Forest interception (Corsican pine)	24% (Freeman 2008)
Broadleaf interception (deciduous)	15% annual mean (Komatsu et al. 2011)
Sy floor (C1 / C2–C5)	6% / 12%
Ridge mask threshold	1.0 m
Excluded wells	ceh12 (bedrock), ceh15 (forest slack edge)

B3. Δh Model Equation

For each well, baseline equilibrium net recharge is:

$$\mathrm{net}_0 = b_1 \cdot \mathrm{P\_eff}_0 - b_2 \cdot \mathrm{PET}_0 - b_3 \cdot |h|$$

Under a scenario:

$$\mathrm{net\_sc} = b_1 \cdot \mathrm{P\_eff\_sc} - b_{2}\mathrm{\_sc} \cdot \mathrm{PET\_sc} - b_3 \cdot |h|$$

where P_eff_sc = P × sP × (1 − I_scenario) for forest wells, PET_sc = PET × sPET, and b_2_sc = b_2 × sB_2 (where sB_2 is now season-specific: sB2_w for winter, sB2_s for summer). The per-well Δh = net_sc − net_0. For the annual season, Δh is the unweighted mean of winter and summer values (each computed with their respective seasonal sB_2). The b_3 drainage term cancels in the difference.

B4. Scenario Parameter Sets

From v2.6.0, the β_2 scaling is seasonally resolved: separate sB2_w (winter) and sB2_s (summer) multipliers replace the former per-cluster sB2_c4/sB2_c5 controls. Both are applied identically to C4 and C5.

Scenario	sP_w	sP_s	sPET_w	sPET_s	I_c4/c5	sB2_w	sB2_s
Baseline	1.00	1.00	1.00	1.00	0.24	1.00	1.00
UKCP18 2050s	1.10	0.85	1.05	1.20	0.24	1.00	1.00
UKCP18 2080s	1.20	0.70	1.10	1.35	0.24	1.00	1.00
Clearfell	1.00	1.00	1.00	1.00	0.00	1.108*	1.108*
Broadleaf	1.00	1.00	1.00	1.00	0.15	0.87†	1.09†
Thinning	1.00	1.00	1.00	1.00	0.12	1.054*	1.054*

** Clearfell and Thinning sB2 values are computed dynamically from the BACI-corrected Edge-tier β_2 ratio (Script 10e, loaded via load_clearfell_b2_multiplier() in clearfell_common.py). The clearfell multiplier is the full Edge-tier ratio; thinning is half the perturbation above unity. C5 receives the same multiplier as C4 by extrapolation. Exact values vary with each pipeline run.*

† Broadleaf sB2 values derive from the Script 21 deciduous phenology profile: winter 0.87× reflects reduced transpiration during the leaf-off period, summer 1.09× reflects elevated transpiration during the growing season. This seasonal split is new in v2.6.0.

B5. Spatial Interpolation

The map surface uses inverse-distance-weighted (IDW) interpolation with power 1 and k = 8 nearest neighbours. A minimum distance floor of 10 m prevents singularities; points within 5 m of a well return the well’s exact value. Interpolation is constrained to the site boundary polygon.

In depth mode, a DEM grid (160 × 110 cells) provides the ground surface. Depth = DEM elevation minus IDW head. Ridge masking suppresses rendering where DEM exceeds the IDW-interpolated DEM surface by > 1.0 m.

B6. Data Bundle

- WELLS: well objects with name, cluster, coordinates, seasonal heads, Sy, SSM coefficients, DEM ground elevation.
- POLYS: KML polygons (site boundary, forest, clearfell, broadleaf, lake).
- CLIMATE: seasonal baselines (P/PET), cluster heads, cluster betas, monthly climatology.
- DEM_GRID: downsampled elevation grid for ridge masking and depth mode.
- HILLSHADE: Base64 RGBA PNG (1100 × 750 px), site-masked, 32 grey levels.

B7. Viewer Extent

Fixed at E 240200–243700, N 362400–364800 (OSGB36), covering all 69 wells. Canvas default 640 × 440 px, resizable. A parallel CSV (19_scenario_summary.csv) mirrors the viewer’s calculations for the manuscript.

B8. Build Provenance

Script 19 assembles data from Scripts 00, 01, 03, 17, 18, and site GIS layers.

Citation: Hollingham, M. (2026). Hydrogeological Dynamics, Behavioural Clustering and Management Intervention Analysis at Newborough Warren Coastal Sand Dune Aquifer, Wales.

Part C

Seasonal Extremes Scatter

C1. Overview

The seasonal extremes scatter (14_seasonal_extremes_scatter.html) is generated by Script 14 (14_climate_projections.py, v1.1.0). It reads per-well summary statistics from the well network table (00_well_network_table.csv) and cluster assignments from 02_cluster_stats.csv.

C2. Data Sources

- Mean_Summer_Min_m: mean of annual summer (Apr–Sep) minimum water-table depths, 2005–2026, per well. Values in metres relative to pipe top (negative = below).
- Mean_Winter_Max_m: mean of annual winter (Oct–Mar) maximum depths, computed identically.
- Cluster: from 02_cluster_stats.csv. Unmatched wells labelled “UNKNOWN”.

C3. Threshold Definitions

Ecological thresholds from Curreli et al. (2013), imported from utils/config.py:

Constant	Value (m)	Meaning
SD15b (summer)	0.61	Summer minimum viability limit for wet dune slack.
SD16 (summer)	0.98	Summer minimum viability limit for dry dune slack.
SD15b_WINTER	0.10	Winter maximum flooding threshold for wet slack.
SD16_WINTER	0.25	Winter maximum flooding threshold for dry slack.

C4. Chart Implementation

Uses Chart.js 4.4.1 from cdnjs. Well data serialised as JSON point objects. A custom plugin (thresholdPlugin) draws the four threshold lines after the scatter data. The search mechanism rebuilds datasets with modified colours and radii: matched well in orange (#ff6600) at radius 13, others at 33% opacity.

C5. Cluster Styling

Colours, labels, and markers defined in utils/config.py (CLUSTER_COLOURS, CLUSTER_LABELS, CLUSTER_MARKERS) as the single source of truth. Five clusters under the k = 5 partition: C1 Lake Edge, C2 Dune, C3 Western Residual, C4 Main Forest, C5 Coastal Forest.

C6. Build Provenance

Script 14 also produces static PNG figures (summer trajectory, winter flooding, stacked two-panel), a summer trend CSV, annual extremes CSV, and winter exceedance summary. The scatter HTML is an additional interactive output complementing these.

Citation: Hollingham, M. (2026). Hydrogeological Dynamics, Behavioural Clustering and Management Intervention Analysis at Newborough Warren Coastal Sand Dune Aquifer, Wales.