

Collar Voltage Report

Technical Guide

Per-subject historic collar/battery voltage monitoring

Generated August 31, 2026

Workflow id: **source_voltage**

1. Overview

The **source_voltage** workflow fetches GPS relocations for a named EarthRanger subject group over a user-defined time range, extracts the collar/battery voltage embedded in each observation's details, and compares it against a matching previous period to build a historic voltage band per subject. Results are rendered as a chart, assembled into a dashboard widget, and merged into a Word doc report.

For each subject the workflow delivers:

- A historic voltage chart — current-period voltage plotted against a min/mean/max band derived from the previous period (falls back to the current period's own values when no previous-period data is available)
- A PNG screenshot of that chart, used in the dashboard widget and the Word report
- A section in the merged Word doc report (**overall_report.docx**)

Output summary

Output type	Count	Description
Historic voltage charts	1 per subject	Current voltage vs. previous-period min/mean/max band
GeoParquet exports	2 total	Current-period and previous-period relocations, each with a voltage column
Dashboard widget	1 total	Merged 'Collar Voltage' map/plot widget across all subjects
Word doc report	1 total	Cover page + one historic voltage chart per subject

2. Dependencies

2.1 Python packages

Package	Version	Channel
ecoscope-platform	>=2.15.0, <2.16.0	ecoscope-workflows
ecoscope-workflows-ext-custom	0.1.0rc14.*	ecoscope-workflows-custom
ecoscope-workflows-ext-ste	0.0.0rc1.*	ecoscope-workflows-custom
ecoscope-workflows-ext-mep	1.0.1.*	ecoscope-workflows-custom
pydeck	0.9.2	conda-forge
opentelemetry-sdk	>=1.20.0, <2.0.0	conda-forge

2.2 Connections

Connection	Task	Purpose
EarthRanger	set_er_connection	Fetch subject group observations for both the current and previous periods

2.3 Grouper

The workflow groups all data by **name** (the renamed subject-name column). The **\$defs** block in the spec restricts the grouper UI's **ValueGrouper.index_name** field to a single option, "Subject Name" — users cannot change the grouping dimension.

2.4 Subject group

A user-provided string parameter (**Subject Group Name**, default: *Elephants*) is passed to both the current-period and previous-period observation fetches to scope the analysis to a single EarthRanger subject group. It is also threaded through unused for a future Word doc integration (comment: "pass this to word doc").

3. Relocation Pipeline

The current and previous periods run through an identical pipeline, in parallel, starting from their own `get_subjectgroup_observations` call.

3.1 Fetch and clean

Step	Task	Detail
1	<code>get_subjectgroup_observations</code>	filter: clean, raise_on_empty: false, include_details: true, include_subjectsource_details: true
2	<code>get_timezone_from_time_range</code> → <code>convert_values_to_timezone</code>	Convert the fixtime column to the analysis timezone (computed once from the current time range, reused for both periods)
3	<code>process_relocations</code>	Retain 12 columns: <code>groupby_col</code> , <code>fixtime</code> , <code>junk_status</code> , <code>geometry</code> , <code>extra__subject_name</code> , <code>extra__subject_hex</code> , <code>extra__subject_sex</code> , <code>extra__created_at</code> , <code>extra__subject__subject_subtype</code> , <code>extra__subjectsource__id</code> , <code>extra__subjectsource__assigned_range</code> , <code>extra__observation_details</code> . Filter 3 invalid coordinate pairs: (180,90), (0,0), (1,1)
4	<code>sort_values</code>	Sort by <code>fixtime</code> , ascending, nulls last
5	<code>persist_df</code>	Persist as GeoParquet to <code>\$ECOSCOPE_WORKFLOWS_RESULTS</code> (<code>relocations.parquet</code> / <code>previous_period_relocations.parquet</code>)

3.2 Previous Period selection

The comparison period is computed by `ecoscope_workflows_ext_ste.tasks.filter.flexible_previous_period`, exposed to the user as a "Previous Period" task-group. It always ends on the current time range's Start Date, so the two periods never overlap:

Mode	Description
Custom	User-supplied Years / Months / Weeks / Days offset (default: 1 month back)
Preset	A common lookback: 1, 3, or 6 months, or 1 year back
Calendar	An exact Start Date

4. Voltage Extraction

Both relocation sets go through `extract_value_from_json_column` against their `observation_details` column, producing a float `voltage` column:

Parameter	Value
column_name	observation_details
field_name_options	battery, mainVoltage, batt, power (checked in this order)
output_type	float
output_column_name	voltage

Note: This generalized field-name lookup replaces an earlier, collar-model-specific voltage extraction task, so the workflow tolerates different EarthRanger device integrations reporting battery/voltage under different JSON keys.

5. Historic Voltage Chart

5.1 Per-subject fan-out

Step	Task	Detail
1	split_groups	Split current and previous relocations by the configured grouper (name)
2	column_first_unique_value	Get each subject's display name from the current-period slice
3	safe_string (ext_ste)	Sanitize the subject name for use in a filename
4	prefix_string_var	Build the chart filename: <code>_historic_voltage.html</code>
5	groupbykey	Pair each subject's current-period slice with its previous-period slice (<code>zip_current_prev_name</code>)

5.2 Plotting

plot_historic_voltage (column: voltage) draws the subject's current voltage series against a band built from the previous period's 2.5th/97.5th percentile and mean. If the previous-period slice is missing or empty, the current period's own values are used for the band instead, so a chart still renders on a subject's first run. If the computed band collapses to a single value, it is widened by $\pm 2.5\%$ so the shaded region remains visible.

5.3 Persisting and rendering

Step	Task	Detail
1	groupbykey	Pair each chart's filename with its rendered HTML (<code>historic_voltage_text</code>); skipped if any dependency was skipped or any keyed pair is a skip
2	persist_text	Write the chart HTML to <code>\$ECOSCOPE_WORKFLOWS_RESULTS</code> (filename_suffix: none — the filename built in step 5.1 is used verbatim)
3	create_map_widget_single_view	Title: "Collar Voltage"; skipif: never, so a widget is always created even if the underlying chart data is empty
4	merge_widget_views	Merge every subject's widget into a single dashboard widget
5	html_to_png	<code>device_scale_factor</code> : 2.0, <code>wait_for_timeout</code> : 10 ms, <code>max_concurrent_pages</code> : 1, <code>full_page</code> : false

6. Word Doc Report

The report template (`source_voltage_report.docx`) is downloaded once via `fetch_and_persist_file` (`ecoscope_workflows_ext_ste`) from a fixed Dropbox URL, with up to 3 retries and `overwrite_existing: false`.

6.1 Report Logo

The "Generate Word Doc Report" task-group exposes a single field, **Report Logo**, backed by `get_file_path`. It accepts an uploaded PNG/JPG saved under `$ECOSCOPE_WORKFLOWS_RESULTS`. This field is optional — the report still generates without a logo.

6.2 Generation

`generate_source_voltage_report` (`ecoscope_workflows_ext_mep`) walks `$ECOSCOPE_WORKFLOWS_RESULTS` for image files, classifies `_historic_voltage`-suffixed images by subject (stripping the suffix) and any other image by its full filename stem, renders them all into the template's voltage-chart list, and adds the cover-page fields below:

Field	Value
<code>org_logo_path</code>	The uploaded Report Logo, or none (cover page omits it)
<code>report_period</code>	The current time range (Since — Until)
<code>prepared_by</code>	"Ecoscope"
<code>filename</code>	<code>overall_report.docx</code>

7. Dashboard

gather_dashboard assembles the final interactive dashboard from the workflow details, the current time range, the configured grouper, and the single merged Collar Voltage widget produced in Section 5.3.

8. Output Files

All outputs are written to **\$ECOSCOPE_WORKFLOWS_RESULTS**. Files marked with *<subject>* are produced once per subject in the group.

File	Description
_historic_voltage.html / .png	Historic voltage chart — current voltage vs. previous-period min/mean/max band
relocations.parquet	All subjects' current-period relocations, including the extracted voltage column
previous_period_relocations.parquet	All subjects' previous-period relocations, including the extracted voltage column
source_voltage_report.docx	Downloaded report template (intermediate asset, not a final deliverable)
overall_report.docx	Final Word doc report (cover page + every subject's historic voltage chart)

9. Workflow Execution Logic

9.1 Global skip conditions

The top-level **task-instance-defaults** block applies two skip conditions to every task unless overridden:

- **any_is_empty_df** — skips the task if any upstream DataFrame dependency is empty
- **any_dependency_skipped** — skips the task if any upstream task was skipped

Two tasks override this default:

- **collared_voltage_widget** uses **skipif: conditions: [never]**, so a dashboard widget is always created even if the underlying chart data is empty
- **historic_voltage_text** uses **any_dependency_skipped** plus **any_keyed_iterables_are_skips** (`unpack_depth: 1`), so an individual subject's filename/chart pair is skipped without failing the whole zip

9.2 mapvalues fan-out

Per-subject processing runs in parallel via **mapvalues**. The fan-out starts at **split_groups** (current and previous relocations, independently) and continues through:

- `split_current_relocs / split_previous_relocs` → `column_first_unique_value` → `safe_string` → `prefix_string_var` (build each subject's chart filename)
- `groupbykey (zip_current_prev_name)` → `plot_historic_voltage` → `groupbykey (historic_voltage_text)` → `persist_text` → `create_map_widget_single_view` → `html_to_png`

9.3 Screenshot timing

The historic voltage chart screenshot uses: `wait_for_timeout: 10 ms`, `device_scale_factor: 2.0`, `max_concurrent_pages: 1`, `full_page: false` — the same configuration used across other Ecoscope chart-to-PNG steps.

10. Software Versions

Package	Version pinned
ecoscope-platform	>=2.15.0, <2.16.0
ecoscope-workflows-ext-custom	0.1.0rc14.*
ecoscope-workflows-ext-ste	0.0.0rc1.*
ecoscope-workflows-ext-mep	1.0.1.*
pydeck	0.9.2
opentelemetry-sdk	>=1.20.0, <2.0.0

Note: All packages are resolved from the prefix.dev Ecoscope conda channels. The wildcard patch-version pin (.*) allows bug-fix releases to be picked up automatically while keeping minor and major versions locked. ecoscope-workflows-ext-ste is pinned to a pre-release (0.0.0rc1.*).