Technical User Guide — SSB Retuning Automations

Tool overview

SSB Retuning Automations is an automation platform for SSB retuning projects that can run in GUI or CLI mode or through a Web Interface (using a server/client infrastructure) and orchestrates five functional modules:

• **Module 0**: Update Network Frequencies.

• **Module 1**: Configuration Audit & Logs Parser.

• **Module 2**: Consistency Check (manual Pre/Post).

• **Module 3**: Consistency Check Bulk (automatic Pre/Post detection by market).

• **Module 4**: Final Clean-Up.

The main execution lives in "src/SSB_RetuningAutomations.py", where CLI arguments, GUI, configuration persistence, input resolution (folders/ZIP), per-module execution, and artifact versioning are managed.

Repository technical architecture

Main files

Orchestration core

• "src/SSB_RetuningAutomations.py": entry point, CLI/GUI parsing, module routing, batch/bulk execution, and versioning.

Main modules files

• "src/modules/ConfigurationAudit/ConfigurationAudit.py": log parsing and audit workbook construction (Excel + PPT).

• "src/modules/ConfigurationAudit/ca_summary_excel.py": assembly of "SummaryAudit" and discrepancy dataframes.

• "src/modules/ConsistencyChecks/ConsistencyChecks.py": PRE/POST loading, relation comparison, discrepancies, and output export.

• "src/modules/ProfilesAudit/ProfilesAudit.py": profiles audit (integrated into module 1).

• "src/modules/CleanUp/FinalCleanUp.py": final clean-up (base implementation for extension).

Common layer and utilities

• "src/modules/Common/*.py": correction command logic and shared functions.

• "src/utils/*.py": IO, parsing, frequency handling, Excel, pivots, sorting, infrastructure, and timing.

Inputs, outputs, and content per module

Module 0 — Update Network Frequencies

Input

• Input folder (may contain subfolders/ZIPs already supported by the IO layer).

• Logs with an "NRFrequency" table and the "arfcnValueNRDl" column.

Process

1. Scan logs and detects "NRFrequency" blocks.

2. Extracts numeric values from "arfcnValueNRDl".

3. Removes duplicates and sorts frequencies.

4. Updates the persisted “Network frequencies” configuration for GUI/CLI.

Output

• Does not generate Excel/PPT.

• Updates the persisted network frequency value used for filtering and selection in later runs.

Module 1 — Configuration Audit & Logs Parser

Inputs

• Input folder with logs (".log", ".logs", ".txt") or ZIPs resolvable by utilities.

• Frequency parameters:

• "n77_ssb_pre"

• "n77_ssb_post"

• "n77b_ssb"

• allowed SSB/ARFCN lists pre/post.

• Flags:

• "profiles_audit"

• "frequency_audit"

• "export_correction_cmd"

• "fast_excel_export".

Process

1. Parses files and extracts MO tables by "SubNetwork" blocks.

2. Generates one sheet per detected table.

3. Builds "SummaryAudit" + pivots/auxiliary summaries.

4. Runs profiles audit if enabled.

5. Exports CA correction commands if requested.

6. Generates the summary PPT.

Outputs

• Folder "ConfigurationAudit_<timestamp>_v<version>/".

• Excel file "ConfigurationAudit_<timestamp>_v<version>.xlsx":

• Sheets for each parsed MO table.

• "SummaryAudit".

• NR/LTE parameter discrepancy sheets.

• Summary/pivot sheets by frequencies and relations.

• PPT file "ConfigurationAudit_<timestamp>_v<version>.pptx".

• Optional folder "Correction_Cmd_CA/" with AMOS commands.

Main semantic content

• **SummaryAudit** contains rows with:

• "Category", "SubCategory", "Metric", "Value", "ExtraInfo",

• and execution context fields (stage, module, etc. depending on the flow).

• "Value" usually represents a count of impacted nodes/cells/relations.

• "ExtraInfo" contains the NodeId list or a compact discrepancy detail.

Module 2 — Consistency Check (Pre/Post)

Inputs

• "input_pre" and "input_post" (or equivalent resolved structure).

• Frequencies "n77_ssb_pre" and "n77_ssb_post".

• Optional reference to PRE and POST "ConfigurationAudit" to enrich target classification.

• Optional list of frequency filters ("cc_freq_filters").

Process

1. Loads relation tables ("GUtranCellRelation", "NRCellRelation").

2. Normalizes columns/keys and selects the most recent snapshots by date.

3. Detect:

• new relations,

• missing relations,

• parameter discrepancies,

• frequency discrepancies,

• summary by PRE/POST frequency pair.

4. Classify destination targets as "SSB-Pre", "SSB-Post" or "Unknown".

5. Exports detailed Excel outputs and correction commands.

Outputs

• "CellRelation_<timestamp>_v<version>.xlsx" (end-to-end relations view).

• "ConsistencyChecks_CellRelation_<timestamp>_v<version>.xlsx" with:

• "Summary"

• "SummaryAuditComparisson" (if there is PRE/POST SummaryAudit)

• "Summary_CellRelation"

• GU blocks: "GU_relations", "GU_param_disc", "GU_freq_disc", "GU_freq_disc_unknown", "GU_missing", "GU_new"

• NR blocks: "NR_relations", "NR_param_disc", "NR_freq_disc", "NR_freq_disc_unknown", "NR_missing", "NR_new"

• optional "GU_all", "NR_all".

• "Correction_Cmd_CC/" with commands per type (new/missing/discrepancies).

Module 3 — Consistency Check Bulk

Inputs

• Root folder with subfolders like "yyyymmdd_hhmm_step0" (optionally nested by market).

Process

1. Detects PRE/POST candidates by the most appropriate date/time.

2. Excludes folders using a blacklist ("ignore", "old", "bad", "partial", "incomplete", "discard", etc.).

3. Runs Module 2 for each detected market.

Outputs

• Same output structure as module 2, per market.

• Traceability file "FoldersCompared.txt".

Module 4 — Final Clean-Up

Inputs

• Final retune working folder.

Process

• Executes final cleanup policies (structure prepared to expand rules).

Outputs

• Versioned cleanup directory according to the active implementation.

Configuration Audit module in detail

SummaryAudit checks philosophy

SummaryAudit sheet contains a high-level checks table by categories. The flow:

1. Excludes "UNSYNCHRONIZED" nodes based on "MeContext".

2. Evaluates NR, LTE, ENDC, Externals, TermPoints, cardinalities, and profiles.

3. Records each check as a row ("Category/SubCategory/Metric/Value/ExtraInfo").

Operational meaning of SummaryAudit rows

• **Category**: audited technical domain (NR/LTE/ENDC/MeContext/etc.).

• **SubCategory**: type of analysis (Audit/Inconsistencies/Profiles).

• **Metric**: specific rule evaluated.

• **Value**:

• Integer: number of affected nodes/relations/cells.

• "N/A": not evaluable due to missing columns.

• Text: captured status or error.

• **ExtraInfo**: list of nodes or bounded detail for troubleshooting.

SummaryAudit checks catalog

MeContext Audit

• total unique nodes and unsynchronized node exclusion.

• "UNSYNCHRONIZED" nodes (excluded from the rest of the audits).

NR Frequency Audit / NR Frequency Inconsistencies

Source tables: "NRCellDU", "NRFrequency", "NRFreqRelation", "NRSectorCarrier", "NRCellRelation", "ExternalNRCellCU", "TermPointToGNodeB", "TermPointToGNB".

Main checks:

• Detection of NR nodes with N77 SSB (band 646600–660000).

• Classification of NR nodes as LowMidBand / mmWave / mixed.

• Nodes whose N77 SSBs are fully within allowed PRE or POST lists.

• Nodes with N77 SSB outside allowed lists.

• Old/new SSB presence per node (only old, only new, both).

• Nodes with NRFreqRelationId in an unexpected format (auto-created outside convention).

• NR relations to old/new SSB.

• NR externals and termpoints pointing to old/new/unknown.

Typical triggering:

• Each check is enabled if the table and minimum required columns exist.

• If columns are missing, a "N/A" status row is added.

• If the table is empty or not found, an informative row "Table not found or empty" is added.

Interpretation:

• "Value > 0" in inconsistencies indicates a real deviation that requires investigation.

• "ExtraInfo" typically lists affected nodes for operational targeting.

LTE Frequency Audit / LTE Frequency Inconsistencies

Source tables: "GUtranSyncSignalFrequency", "GUtranFreqRelation", "GUtranCellRelation", "ExternalGUtranCell", "TermPointToENodeB".

Main checks:

• LTE nodes with old/new SSB.

• Nodes with both old/new or old without new.

• SSB outside the expected pre/post set.

• LTE relations to old/new and parameter discrepancies per cell relation.

• LTE externals OUT_OF_SERVICE for old/new.

ENDC Audit / ENDC Inconsistencies

Source tables: "EndcDistrProfile", "FreqPrioNR".

Main checks:

• "gUtranFreqRef" and "mandatoryGUtranFreqRef" with old/new + N77B combinations.

• Nodes that do not contain the expected frequency combination.

• In "FreqPrioNR": old without new, both present, and parameter mismatch per cell.

Cardinalities Audit / Inconsistencies

Cardinality checks per relation table (per node and/or per cell) to detect overprovisioning or gaps versus expected limits.

Profiles Audit (if enabled)

• Compares PRE/POST profiles by supported profile MO.

• Detects parameter discrepancies between old/new variants.

• Adds results to SummaryAudit and auxiliary detail sheets.

Consistency Check module in detail

Filtering by non-retuned nodes

If a POST SummaryAudit exists, the module obtains PRE/POST node lists and can exclude discrepancies whose target points to nodes that did not complete retune, reducing operational noise.

How it detects parameter discrepancies

1. Selects common PRE and POST relations by composite key:

• GU: typically "NodeId", "EUtranCellFDDId", "GUtranCellRelationId".

• NR: typically "NodeId", "NRCellCUId", "NRCellRelationId".

2. Excludes control columns (keys, frequency, Pre/Post, Date).

3. Compares value-by-value across shared columns.

4. Sets "ParamDiff=True" if at least one column differs.

5. In GU it ignores "timeOfCreation" and "mobilityStatusNR" to avoid false positives.

How it detects frequency discrepancies

1. Extracts base frequency from relation references ("extract_gu_freq_base" / "extract_nr_freq_base").

2. Discrepancy rule:

• if PRE had "freq_before" or "freq_after", and POST does **not** end up in "freq_after", it marks "FreqDiff=True".

3. Classifies the discrepancy as:

• "FreqDiff_SSBPost" (target identified as SSB-Post),

• "FreqDiff_Unknown" (cannot be associated to a known target).

How it detects neighbor discrepancies

They are split into three groups:

• **New relations**: keys present in POST and absent in PRE.

• **Missing relations**: keys present in PRE and absent in POST.

• **Discrepancies**: same key in PRE/POST but with parametric or frequency differences.

Content of each ConsistencyChecks output sheet

• **Summary**: KPIs per table (PRE/POST volume, discrepancies, new/missing, source files).

• **SummaryAuditComparisson**: diff of SummaryAudit PRE vs POST metrics (without "ExtraInfo" to keep the comparison clean).

• **Summary_CellRelation**: KPI per "Freq_Pre/Freq_Post" pair and per technology.

• **GU_relations / NR_relations**: relation universe enriched with target classification and command snippets.

• **GU_param_disc / NR_param_disc**: common relations with param differences.

• **GU_freq_disc / NR_freq_disc**: frequency discrepancies to SSB-Post targets.

• **GU_freq_disc_unknown / NR_freq_disc_unknown**: discrepancies with non-classifiable targets.

• **GU_missing / NR_missing**: relations removed versus PRE.

• **GU_new / NR_new**: relations added in POST.

• **GU_all / NR_all**: optional consolidated dump for extended analysis.

Input requirements and operational best practices

• Keep market log exports in a consistent structure (especially for bulk).

• Validate that PRE/POST have the same table granularity and consistent naming.

• Validate frequency inputs ("n77_ssb_pre", "n77_ssb_post", "n77b_ssb") before batch execution.

• Correctly configure allowed SSB/ARFCN lists to minimize false positives.

• Run **Configuration Audit** before Consistency Checks whenever possible.

• Use Bulk mode only with a controlled folder naming convention.

• Review "Summary" and "Summary_CellRelation" first, then then deep-dive into detail sheets (ConfigurationAudit) and discrepancy tabs (ConsistencyCheck).

• Consume "Correction_Cmd_CA" and "Correction_Cmd_CC" as a remediation proposal, not as blind execution.

Execution Modes and Versioning

• **GUI mode**: run without CLI arguments.

• **CLI mode**: run with explicit module and options.

• **Web Interfacee**: the tool can be run in a server/client infrastructure, accessing the server through a Web Interface where you can unpload your inputs, enqueue different tasks and export the results when finish..

All Generated artifacts include a versioned suffix: "<timestamp>_v<TOOL_VERSION>".

This guarantees traceability and avoids collisions between runs.

Known limitations and considerations

• The engine depends on log quality and structure: missing columns downgrade checks to "N/A".

• Some rules depend on naming conventions in references (NR/GU relation refs).

• The Final Clean-Up module is prepared to extend operation-specific policies.

Quick module reference

Module	Main input	Main output	Goal
0 Update Network Frequencies	Logs folder	Persisted config	Update network frequency list
1 Configuration Audit	Logs/ZIP folder	Excel + PPT + CA commands	Audit configuration and profiles

Module	Main input	Main output	Goal
2 Consistency Check	PRE and POST folders	2 Excel + CC commands	Compare pre/post relations
3 Consistency Check (Bulk)	Multi-market root folder	Module 2 outputs per market	Run bulk comparison
4 Final Clean-Up	Final folder	Clean-up folder	Operational final clean-up