

Technical Reference

[Русский](#) · [User Guide](#) · [ОКТМО](#)

Contents

- [Address Parsing](#)
- [Slots](#)
- [Matching Passes](#)
- [Settings](#)
- [Sources](#)
- [Run Artifacts](#)
- [Result Formatting](#)
- [Project Layout](#)

How address parsing works, the slots, the matching passes, the configuration files, and the run artifacts.

Address Parsing

A raw address string is broken into components. The order matters: first the trailing building number is cut off, then the type and name of the street or territory are determined, then the settlement and the administrative unit above it.

What Is Recognised

Street types. улица, переулок, проспект, бульвар, тракт, шоссе, тупик, проезд, аллея, линия, набережная, площадь, квартал, микрорайон, километр, автодорога — with all the usual abbreviations ([ул.](#), [пер.](#), [пр-т](#), [б-р](#), [наб.](#), [мкр.](#)).

Territory types. тер., территория, СНТ, ДНТ, ТСН.

Settlement types. город, посёлок, деревня, село, слобода, станица, станция, хутор, улус, местечко, кишлак, аул, аал, арбан, починок, выселок, заимка, кордон, маяк, погост, слободка, усадьба, лесоучасток, метеостанция, разъезд. Railway forms are parsed separately: `п. ж/д ст.`, `ж/д остановочный пункт`, `ж/д блокпост`, `ж/д будка`, `ж/д ветка`, `ж/д казарма`, `ж/д платформа`, `ж/д площадка`, `ж/д путевой пост`, `рзд.`. Compact forms `гп`, `рп`, `кп`, `дп`, `пгт`, `нп` are understood with and without dots.

Building number. The trailing part is split into a base and modifiers: `10`, `10а`, `10/2`, `10к2`, `10 корп. 2`, `10 стр. 1`, `10 литера А`. The base is the number; the modifiers are everything else, brought to one form.

The Key Rule About Building Numbers

Building modifiers can never be collapsed into a bare number. `10к2` does not take `10`, and `10` does not take `10к2` — neither on a strict pass nor on a relaxed one. The looseness of a pass concerns the spelling of the street, not blocks and letters.

Slots

A slot is one named component of an address. The address line is assembled from the enabled slots; in decomposed form each slot becomes a column.

Address Line Slots

Slot	Column	Short	What it is
<code>postal_index</code>	<code>Postal_Index</code>	<code>Index</code>	Postal index
<code>federal_district</code>	<code>Federal_District</code>	<code>Fed_District</code>	Federal district
<code>parent_subject</code>	<code>Parent_Subject</code>	<code>Parent</code>	Parent subject
<code>subject</code>	<code>Subject</code>	<code>Subject</code>	Federal subject
<code>autonomous_okrug</code>	<code>Autonomous_Okrug</code>	<code>AO</code>	Autonomous okrug
<code>municipality</code>	<code>Municipality</code>	<code>Municipality</code>	Municipality
<code>locality</code>	<code>Locality</code>	<code>Locality</code>	Settlement
<code>territory</code>	<code>Territory</code>	<code>Territory</code>	Territory, dacha association, and the like

Slot	Column	Short	What it is
<code>microdistrict</code>	<code>Microdistrict</code>	<code>Mkr_Qtr</code>	Microdistrict, quarter
<code>street</code>	<code>Street</code>	<code>Street</code>	Street
<code>house</code>	<code>House</code>	<code>House</code>	The building as a whole
<code>premise</code>	<code>Premise</code>	<code>Premise</code>	Premise, flat

Technical Slots

Slot	Column	Short	What it is
<code>oktmo_code</code>	<code>OKTMO_Code</code>	<code>OKTMO</code>	OKTMO code
<code>oktmo_name</code>	<code>OKTMO_Name</code>	<code>OKTMO_Name</code>	Official OKTMO name
<code>house_base</code>	<code>House_Base</code>	<code>House_No</code>	Building number without modifiers
<code>house_mods</code>	<code>House_Mods</code>	<code>House_Mods</code>	Modifiers in normalized form
<code>house_modifiers</code>	<code>House_Modifiers</code>	<code>House_Parts</code>	Modifiers as a list
<code>street_numbers</code>	<code>Street_Numbers</code>	<code>Street_Nums</code>	Numbers inside the street name
<code>territory_key</code>	<code>Territory_Key</code>	<code>Territory_Key</code>	Territory key
<code>match_key</code>	<code>Match_Key</code>	<code>Match_Key</code>	Matching key

Defaults

The default assembly order of the address line is the order of the address line slots in the table above, top to bottom.

Every slot is enabled by default except `microdistrict`: in Russian addresses a microdistrict or quarter more often duplicates the street than complements it. The microdistrict column still remains — only the value is excluded from the line.

Slot column headers are written in full or short form, chosen on the reference processing screen.

Matching Passes

The matching engine is `Ultimate_GT_Aligner`. For each address column a candidate pool is built, and the reference rows take their pairs out of it.

A matched pair is removed from the pool immediately. One candidate cannot be handed out twice, and a pass that ran earlier is never overridden by a later one.

The passes run in strictly increasing looseness.

Pass	Name	Match condition
0	Source row	Collection-before-match mode only: the candidate comes from the same source row and its components agree
1	Exact string	The normalized address texts are equal
2	Component hash	Territory, street words, street numbers, building number, and modifiers all match
2R	Relaxed hash	The same without modifiers in the key — but the building check still requires matching modifiers
2C	Core hash	Collection mode only: street and building without territory. Requires a strict building check and compatible territory codes
3	Fuzzy, inside a territory	Street name similarity within one territory cluster, cutoff 85
4	Fuzzy, across territories	The same across the whole pool, cutoff 85, with weights
5	Relaxed	Cutoff 80; building modifiers are still protected

Precise passes are 0, 1, 2, 2C, 3, and 4. **Relaxed** are 2R and 5. The log summary separates them into their own lines.

Weights on the Fuzzy Passes

At equal street name similarity a candidate receives bonuses:

- matching building number — plus 50;
- matching territory — plus 30;
- matching building modifiers (pass 5) — plus 20;

- matching OKTMO territorial context — a bonus from the search scope profile.

A pair with incompatible territory codes is discarded before weights are counted.

Rows Without a Street

When a reference row has a building number but no street, comparing names approximately is meaningless. Such rows look for a candidate with the same building number, modifiers, and street numbers, and pick the best one by territorial weights.

Settings

config/project.yaml

```
address_aligner:
  city: "ТЮМЕНЬ"           # city removed from keys during normalization
  ground_truth_column: "B"  # reference column; "" = auto-detect
  target_columns: "C,D"     # address columns; "" = auto
  companion_mode: "auto"    # auto | none | left | right | between
  max_auto_align_columns: 5 # how many columns to take on the right in auto mode

safety:
  cleanup_managed_workspace_only: true  # clean only project folders
  never_delete_original_inputs: true    # never delete original sources
```

Values from the window take priority over the file: an empty field in the window means "take from settings or detect automatically", a filled one overrides.

config/gui_settings.yaml

```
gui:
  language: "en"           # ru | en
  theme: "code_dark"
  emoji: false
  allow_runtime_switching: true
  advanced_open: false     # open the Advanced block right away
  source_path: ''
  destination_path: ''
```

Other Configuration Files

File	Contents
<code>config/tool_manifest.yaml</code>	Every command, field, hint, and service function
<code>config/version.json</code>	Version and the language of the root README
<code>config/path_history.json</code>	History and pins of source and target paths
<code>config/ui_colors.yaml</code>	Interface colours
<code>config/oktmo_current_keys.txt</code>	OKTMO search keys
<code>config/oktmo_*_pins.json</code>	Pinned regions and municipalities

`tool_manifest.yaml` is what the window is built from. Each command is described by a title, a description, a set of fields, and a reference to a service-layer function. Both languages live in the same file as `label` / `label_ru` pairs.

Sources

Supported: `.xlsx`, `.docx`, `.odt`, `.txt`, `.csv`, `.md`, `.markdown`, and `.pdf` with a text layer. A PDF without a text layer is not recognised — that is not this program's job.

Legacy `.doc` and `.xls` are handled only after conversion by a separate command.

Office temporary files (`~$*`) are skipped and removed during input validation.

Two-Level Headers

A row that carries no data values and sits under a horizontally merged parent header counts as a sub-header row. Its values are joined to the parent headers, and the row itself does not reach the data.

A column headed "Проектная мощность, мест" over "по корпусам" becomes "Проектная мощность, мест по корпусам" instead of an unnamed `column_5`.

Run Artifacts

Results

Prefix	What it is
AddressCollection_*.xlsx	Collected address workbook
AddressReference_*.xlsx	Reference directory

Names carry a YYYYMMDD_HHMMSS timestamp.

Reports in report/

File	Run
address_collection_summary.json	Address collection
address_collection_benchmark.json	Collection scored against a control column
address_alignment_summary.json	Address alignment
reference_normalization_summary.json	Reference normalization
reference_processing_summary.json	Reference processing
table_comparison_summary.json	Comparison of two tables
safe_table_join_summary.json	Moving columns by address
legacy_office_conversion_report.json	DOC/XLS conversion
rosstat_oktmo_update_summary.json	OKTMO registry update

Some operations additionally write <result name>.summary.json next to the workbook.

Audit Sheets

Moving columns by address adds an AUDIT_ADDRESS_JOIN sheet to the result. Address collection writes a rejected sheet when that option is on.

Result Formatting

Result workbooks go through post-processing: column widths are fitted to the content, long text wraps, row heights grow, and status columns get colour fill.

This is not cosmetics: without it every export has to be tidied by hand, and a row with a long address cannot be read in a narrow column.

Project Layout

Layer	Where	Responsibility
Window	<code>system_core/ui_nicegui/</code>	Interface, workspace folders, log, artifacts
Services	<code>system_core/services/</code>	Binding the window to the engines, reports, parameters
Address engine	<code>system_core/address_engine/</code>	Parsing, slots, OKTMO, reference, collection
Matching	<code>system_core/Ultimate_GT_Aligner.py</code>	Matching passes and linked column transfer
Join by address	<code>system_core/safe_table_join.py</code>	Table comparison and column transfer through Excel
Core	<code>system_core/core/</code>	Jobs, paths, encodings, manifest

A detailed file map is in tools/PROJECT_FILES_GUIDE_EN.md.