

Power BI endpoints - list of available data sources

The voxDeveloper CRM Power BI API provides data in JSON format. You can use it in Power BI, in your own scripts and dashboards, and in tools built with the help of AI agents.

This article contains a catalog of data sources and extended documentation for four reports: [investment status](#), [contract register](#), [contacts and unit handling stages](#), and [payment schedule](#). This scope is enough to start building a dashboard. The available sources and investments depend on the permissions of the API key.

Access (Dostęp): The API key must be obtained from the support team. Power BI configuration is described in the article [Integration with Power BI](#). Do not place the key in public HTML, in a URL, or in a conversation with an AI agent.

☐☐ **Scope of verification** The parameter names, response shapes and filter behavior described in this article were confirmed **on a live demo instance (panel)** (September 2026) — every call was executed and its response checked. The record counts given below come from that instance and serve only to show the scale of the phenomenon; on your instance they will differ. The JSON examples remain illustrative: they contain fictitious values and selected fields, and are not a dump from a client instance. Translated field labels and the scope of shared data depend on the instance and its settings — confirm them against your own response when you first connect.

Getting started

1. Determine the instance (panel) name and the access scope of the key.
2. Retrieve the list of sources available to this key via endpoints.
3. Choose a report, read the record definition and set the correct date range.
4. Retrieve the data, check the response and the required fields, and only then calculate metrics.

First HTTP request:

```
GET https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/endpoints
Api-key: TWOJ_KLUCZ_API
Accept: application/json
```

NAZWA_PANELU and TWOJ_KLUCZ_API are placeholders to be replaced with your own values. The endpoints endpoint returns a list with the fields endpoint and description. The list in this article is a catalog of sources; the API response determines the access of a specific key.

General retrieval rules

- A report address has the form /webservice/power-bi/source/NAZWA_ENDPOINTU. Authorization is done via the Api-key header.
- URL-encode parameter values, e.g. a space as %20. The investment name must be written exactly as in the system. Do not treat an unknown parameter as an effective filter just because the response has HTTP status 200.
- No filters does not mean full history. An endpoint may have a default range of dates, statuses and object types. The available investments always depend on the key.
- Record the retrieval date, the filters applied and the record count separately for each report. Several consecutive retrievals do not constitute a single database transaction.

How to pass parameters

The rules in this section apply to **all** data sources, not only the four described in detail below. Filters can be supplied in two ways, and both work identically — they return byte-for-byte the same

Page 1 / 53

response:

- **segment pairs in the URL path** — https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/clients-report/contact-created-date-from/2026-01-01/contact-created-date-to/2026-03-31
- **classic query string** — https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/clients-report?contact-created-date-from=2026-01-01&contact-created-date-to=2026-03-31

The source documentation uses the path notation — stick to it so that the examples remain comparable. URL-encode values, e.g. a space as %20, and Polish characters in investment names as UTF-8 sequences (Jaworowe%20Wzg%C3%B3rze).

Copy parameter names exactly. date-from and date_from are two different names and are not interchangeable, and the API has no single convention — whether a given report uses a hyphen or an underscore is determined solely by that report's description. For example, clients-report and leads-report require hyphens (contact-created-date-from, lead-created-date-from), while agreements, sales-forecast and activities-list require underscores (date_from, activity_date_from). The same report can mix both conventions: agreements-registry uses both agreement_date_from with underscores and first-agreement-date-from with hyphens.

⚠ Important: An unrecognized parameter does not cause an error — it is **silently ignored**, and the response has status HTTP 200. Only that one parameter is ignored: the remaining filters and default ranges still apply. If it was the only filter, you get the full default set — lead-comparision/nieistniejacy-parametr/abc returns the same 24 rows as a call without parameters. This is the most common cause of silent data discrepancies: the report refreshes correctly, but on a different range than requested.

The endpoint name in the path is also written with hyphens. An underscore variant does not exist — lead_comparision returns HTTP 404 with an empty body. The correct list of names available to your key is provided by the endpoints source.

HTTP response codes

The API returns JSON only for successful responses and for 400 and 500 errors. Access denial and an unknown endpoint name return an HTML body or an empty body — a JSON parser will then report a syntax error rather than an empty table. **Before reading the body, check the HTTP code and the Content-Type header.**

Code	When it occurs	Response body
200	Valid request	JSON, Content-Type: application/json; charset=utf-8
400	Missing required parameter, invalid date format or the report's maximum date range exceeded	JSON {"error": "..."} — read the message, do not treat the response as empty
403	Missing Api-key header or invalid key	HTML of the CRM login page (approx. 7 kB, with Login - vCRM in the title), Content-Type: text/html
403	The key has no access to this	empty body (0 bytes),

Code	When it occurs	Response body
	source, has no investments assigned, or an unrecognized name was given in the investment filter	Content-Type: text/html
404	Unknown endpoint name in the path (e.g. lead_comparison instead of lead-comparison)	empty body (0 bytes), Content-Type: text/html
500	System-side error	JSON {"error": "An unexpected error has occurred. Please contact the system administrator."}
429 / 503	Firewall block under intensive polling; the response does not come from the API and is not JSON	—

The codes 401 and 429 are not generated by the API itself — the controller returns only 200, 400, 403, 404 and 500. The codes 429 and 503 may come from the firewall in front of the instance when requests are sent too frequently. In that case, reduce the polling frequency and do not retry many requests in parallel; no application layer guarantees a Retry-After header.

⚠ Important: The API **does not return a 401 code** — every authorization problem is reported as 403. Do not treat an HTML response or an empty body as a network failure or as a zero business result: it is an access denial. You can tell the two different 403 cases apart by body size — approx. 7 kB of HTML means a problem with the key itself, 0 bytes means a valid key without permission to the given source or to the specified investment.

An HTTP error, invalid JSON and an empty array are three different situations and must not be handled by a single code branch.

Response shapes

Do not assume a single format. A typical instance has **six** response shapes. A loader should recognize all of them before it starts reading rows.

No.	Shape	Where the rows are	Example reports
1	Bare array [{...}, {...}]	the entire response	agreements-registry, schedule-payment-list, transaction-list, transaction-list-anonymized, sales-forecast, leads-list, clients-list, meeting, returns, available-for-sale, sold-associated-units, mobile-call-list, lead-comparison, gclid-conversions,

No.	Shape	Where the rows are	Example reports
			realestate-changes-list, reservations-list, warehouse-contracts-by-source, endpoints
2	{ "data": [...] }	the data field	agreements, invoices, payments, lead-list-agreements
3	{ "data": [...], "meta": { ... } }	the data field; meta is range metadata	investment-status, contact-transactions
4	{ "success": true, "timestamp": ..., "total_count": ..., "data": [...] }	the data field, after checking success	agreements-and-reservations, clients-report, leads-report, activities-list (here the counter is called total_records, not total_count)
5	{ "NAZWA_WŁASNA": [...] }	a field with a name other than data	agreements-list → transactions, branch-users → users
6	Object without a row array	the nested structure must be expanded	realestate-list (investment name → unit UUID → {"name": "1.1"}), sold-and-not-sold (Sprzedane / Niesprzedane / Podsumowanie → unit type → array), investments-sales-targets (formally a named collection {"investments": [...]}), but its only element is an investment → year → month tree, so it has to be expanded anyway), incoming-call (a single flat object, not a list)

Practical notes:

- The success field occurs only in shape 4 — its absence does not indicate an error, and do not make loading data in other reports conditional on it.
- An empty response can take the form [] (shape 1) or {"users": []} (shape 5). Both mean no data for the given filters, not an error.
- Shapes 5 and 6 have no data field at all — code that looks for it unconditionally will crash on these reports.
- Confirm the shape on the first retrieval of each report and check it again after a system update.

The default report population is usually a subset

No filter does **not** mean complete data. Contract and payment reports omit some records by default — without the appropriate parameter the result is understated, and the system issues no warning. Figures measured on the demo instance (September 2026) show the scale of the phenomenon:

Report	Without filter	With filter	What is dropped by default
transaction-list	629	1505 (all-agreements/1)	cancelled contracts — more than half of all
agreements-registry	610	1472 (include_canceled/1)	862 cancelled contracts
agreements-list	629	157 (status/signed)	nothing — this filter narrows to signed contracts
schedule-payment-list	703	1138 (date-from/2010-01-01)	tranches with a due date older than one year back
sold-and-not-sold	457 sold apartments (the entire "Sprzedane" group is 842 units of all types)	100 (agreement-status/Podpisana)	nothing — this filter narrows to units with a signed contract

⚠ **Important:** Before you report a number of contracts or an amount of arrears, check whether the report omits by default what you are asking about. transaction-list without all-agreements/1 will never show the status "Anulowana" (Cancelled), and schedule-payment-list without an explicit date-from starts due dates exactly one year back from the retrieval date.

Status filter values: sometimes in English, sometimes in the instance language

Status names given in filters follow no single convention, and an unrecognized value is ignored without notice — the response then gives you the full default set or an empty array, never an error message.

- agreements-list — English only: inprogress, signed, canceled. The query status/signed returns 157 records, while status/Podpisane returns 629, i.e. everything.
- reservations-list — English only: active, canceled, expired. Polish names return an empty array.
- sold-and-not-sold — the opposite: values are **translated into the instance language**, so on a Polish installation you enter Podpisana. Letter case beyond the first letter matters, and the status "Anulowana" (Cancelled) cannot be selected with this filter.

For each status filter, check in the description of the specific report which convention it expects, and compare the record count with the result without the filter.

Investment filter — a typo gives 403, not an empty list

The value of the investment filter is matched against the names of the investments assigned to the API key. If none of the given names matches, the system denies access in the same way as for an invalid key: **HTTP 403 with an empty response body**. You do not get an empty list or a message

about an incorrect name.

Before you start looking for a problem with the API key, check whether the investment name is written exactly as it appears in the system — including letter case, spaces and diacritics. The list of names the key has access to is provided by the `realestate-list` endpoint: investment names are the top-level keys of its response.

For several investments, names are separated by commas, but **a single correct name is enough for the whole request to pass** — the others, if misspelled, are silently ignored and there is no information about them. After setting the filter to several investments, always compare the number of investments present in the response with the expected number.

Example: https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/available-for-sale/investment/Jaworowe%20Wzg%C3%B3rze,Magiczny%20Zak%C4%85tek

The investment filter does not remove records that have no investment assigned — in some reports (including `clients-report`, `leads-report`, `mobile-call-list`) they are always returned, with an empty investment name. As a result, totals calculated separately for each investment will not add up to the result without the filter: the same “no investment” records will be repeated in every call.

No pagination and response size

The API does not provide limit, offset or page parameters. Each request returns all rows matching the filters — the only way to limit the response is to narrow the range of dates, investments or statuses. The response is built entirely in server memory before it is sent, so a wide range on a large instance means a long wait and a risk of connection timeout.

For orientation — orders of magnitude measured on a small demo instance (approx. 610 contracts in the register, approx. 1,600 units in the records, of which 733 are available for sale):

Report	Number of rows	Response size
sales-forecast	468	approx. 1.5 MB
schedule-payment-list	703	approx. 730 kB
agreements-list	629	approx. 720 kB
transaction-list	629	approx. 580 kB
agreements-registry	610	approx. 570 kB

With a portfolio of tens of thousands of units, the same reports reach tens of megabytes. Split large ranges into periods (month, quarter) or by investment and combine the results in Power BI. On the first retrieval, start with a narrow, explicitly specified range — widen it only after checking the response. Keep in mind that some reports impose a maximum date window width themselves (e.g. 3 or 6 months), and exceeding it results in a 400 code with a message, not a shortened list.

Number and amount format

The same quantities may be returned in two forms, depending on the report:

- **JSON number**, with a decimal point — e.g. `"Wartość BRUTTO": 243332.64`, `"amount": 63994.02`, `"realestate_value": 12196251`;
- **text in Polish format**, with a decimal comma and a space as the thousands separator — e.g. `"current_gross_price": "231 166,01"`, `"Łączna sprzedaż (wartość netto) PLN": "182 116"`

524,86".

The thousands separator in the second form is a regular space (U+0020), not a non-breaking space — when cleaning values it is enough to remove spaces and replace the comma with a period. In Power Query, set the type to Decimal Number with the locale "Polish (Poland)". Do not apply this conversion wholesale to all text columns.

⚠ Important: A comma in a value is not always a decimal separator. Aggregated identifier fields return a list of IDs joined by commas — "ID Klient": "918,1" in the contract register means two clients with IDs 918 and 1, not the number 918.1. In addition, these fields have a mixed type: in the same report some rows are numbers and some are text. Load all fields whose names begin with ID as text and do not convert them to numbers.

Percentages may also be text with a comma ("Łączna ilość (%)": "78,52") and are not fractions — they are values expressed in percent.

null, empty text, a missing field and zero are not interchangeable. Calculate amounts with decimal precision.

JSON key names — Polish and English

The API has no single naming convention, and it cannot be predicted from the endpoint name.

- Reports derived from CRM exports return keys **in Polish**, with Polish characters, spaces and punctuation: Typ umowy, Wartość BRUTTO, Końcowa NETTO, Źródło pochodzenia, Łączna ilość (%), L.P., Powierzchnia lokalu (total).
- Newer and integration endpoints return **English** keys in a technical format: investment_name, sales_progress, realestate_count, transaction_id, client_crm_id, payment_date.

The same information may therefore have different names in two reports. Do not join tables based on similar column names — before joining, confirm that they refer to the same object and the same identifier space. Polish key names come from interface translations, so they may change after a system update.

Common parameter: format-numbers/1

You can add the segment /format-numbers/1 to the address of any report. Column names are then rewritten to ASCII: Polish characters lose their diacritics, periods and closing parentheses are removed, and opening parentheses are replaced with a space. **Field values remain unchanged** — the parameter modifies only key names.

Original name	After applying format-numbers/1
L.P.	LP
Wartość BRUTTO	Wartosc BRUTTO
Końcowa NETTO	Koncowa NETTO
Usuniętych leadów	Usunietych leadow
Powierzchnia lokalu (total)	Powierzchnia lokalu total (two spaces — the opening parenthesis turns into a space)

Example: https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/transaction-list-anonymized/format-numbers/1

⚠ **Important:** The parameter works correctly only in reports whose response is a bare array of rows (shape 1 in the response shapes table). In reports that return an object — for example agreements-and-reservations (fields success and data) or investment-status (data and meta) — processing operates on the top-level keys: column names **are not changed**, and all scalar top-level fields, including success and timestamp, **disappear from the response**. If your flow checks the success field, do not add this parameter. Also apply it consistently in all refreshes — renamed columns will break an existing Power BI model.

How to check whether a filter worked

An HTTP 200 status does not confirm that the filter was applied. An unrecognized parameter, an unrecognized dictionary value and a parameter written in the wrong convention (hyphen instead of underscore) are silently ignored. Therefore, check every new filter with one simple test.

- Run the same query **twice**: once without the filter and once with it, keeping the other parameters unchanged.
- Compare the record counts: in shape 1 — the array length, in shapes 2-4 — the length of the data array or the total_count / total_records field, in shapes 5 and 6 — the number of elements in the relevant collection.
- **An identical record count means that the parameter or its value was not recognized**, not that all records meet the condition. Interpret an identical response size in bytes the same way.
- If the filter did work, also check the range of values in the response — for a date filter, the oldest and newest date in the result. A date older than the given boundary means the report filters on a different field than you assume.

Example of a filter that works: .clients-report returns 110 records, while .clients-report/contact-created-date-from/2026-01-01/contact-created-date-to/2026-03-31 returns 46 — both total_count and the length of the data array change.

Example of an ignored parameter: .lead-comparision returns 24 rows and .lead-comparision/nieistniejacy-parametr/abc returns the same 24 rows; the response is byte-for-byte identical.

When there are several filters, a typo in one name disables only that one filter — the others still work, so the result will be narrowed, but wider than requested. A comparison with a call without filters will not detect this; check each filter separately.

💡 **Tip** When narrowing to an investment, the test looks different. An incorrect name will not give the full set, but HTTP 403 with an empty body — so if you get 403 instead of data, first check the spelling of the investment name, and only then the key's permissions.

What records and identifiers mean

Concept	Meaning for analysis
Lead and contact	These are different objects. A lead ID is not a contact ID. Lead and client counts may refer to different populations.
Contract and unit	One unit can appear in successive contracts. The number of contracts is not automatically the number of units sold.

Concept	Meaning for analysis
Contact-subject link	A contact can have many rows in a report. Counting rows does not give the number of unique clients.
Tranche and payment	A tranche has a due date and a balance. The amount settled in a tranche does not determine the date the money was received.
ID and UUID	Store them as identifiers, preferably as text. Keep the instance name and the object type. The contract UUID, the UUID lokal (unit UUID) and the tranche UUID identify different objects.
Investment or unit name	This is a label. It is not a guaranteed persistent key for joining reports.
Source and contact method	Contact source, lead source and contact method are separate pieces of information. They should not be merged based on similar names.
Contract status vs. the fact of signing	The Status column (W toku / Podpisana / Anulowana) describes the workflow state of the transaction in the CRM, not the fact that the document was physically signed. The date does not settle this either: in agreements-registry the Data and Podpisana columns contain the same value — the contract date — and are sometimes filled in for contracts that are still being prepared. On the demo instance, within a 12-month window, all 276 sales contracts (reservation, developer, preliminary, final) had a date filled in the Podpisana column, but only 58 of them had Status = Podpisana. A metric calculated by status and a metric calculated by contract date will therefore give results differing several-fold — choose one and record it in the definition.
Source “Inne” (Other)	This is the default entry in the source dictionary (id 1), inserted when no one indicated a channel — on import without a source column, for automatically created contacts and for leads without a matched value. It is not an acquisition channel. In a channel ranking, show it as a separate item “no source indicated” and do not include it in campaign effectiveness comparisons. On the demo instance, in the warehouse-contracts-by-source report, “Inne” was the most frequent value of the source_of_origin field.
“Pozostałe” (Remaining) in investment status	In the investment-status report, the Pozostałe category means “no matching signed

Concept	Meaning for analysis
	contract”, not “available in the offer”. The report does not check the status of the unit record, so it also includes units that are withdrawn, on hold, not yet released for sale, and units with a contract in W toku (In progress) status. For the question “how many units can be bought today”, use available-for-sale.

Data joining rule: matching ID values do not prove a shared meaning. Before joining, confirm the object, the identifier space and the one-to-one or one-to-many relationship. Do not automatically substitute a missing key with a name, e-mail address or phone number. Also remember that ID value coverage — i.e. what share of identifiers from one report has a counterpart in the other — is a data quality diagnostic, not proof that both columns describe the same object; high coverage can just as well confirm a shared relationship as a coincidental overlap in numbering.

Data source catalog

Below is a catalog of 41 data sources. Items 11, 16, 19 and 27 are described in detail. The remaining descriptions serve as orientation among the available reports; they are not a complete response specification.

This list is a catalog of sources existing in the system — not a list of your key's permissions. Actual access is determined by the response of the endpoints endpoint (item 37), not by this table: a key may see a narrower set. On the demo instance, the key sees all 41 sources.

No.	Category	Endpoint name	Short description
1	Leads and clients	lead-comparision	Monthly summary of lead and client counts (without a breakdown by investment)
2	Leads and clients	leads-list	List of leads; client classification and status columns are conditional
3	Leads and clients	clients-list	List of clients with data, consents and preferences
4	Leads and clients	clients-report	Report by client with activity history
5	Leads and clients	leads-report	Report by lead with processing history
6	Leads and clients	lead-list-agreements	Leads linked to contracts
7	Leads and clients	gclid-conversions	Google Ads offline conversions (GCLID)
8	Leads and clients	google-ads-conversions	Conversions to Google Ads with hashed contact data

No.	Category	Endpoint name	Short description
9	Leads and clients	leads-activities-history-rp	Activity history of leads from the rynekpierwotny.pl portal
10	Leads and clients	company	Company records (corporate clients and counterparties)
11	Contracts and reservations	agreements-registry	Contract register with unit details
12	Contracts and reservations	transaction-list	Contract list export
13	Contracts and reservations	agreements-list	List of contracts with a status filter
14	Contracts and reservations	agreements	Units with the current contract stage (without identifiers)
15	Contracts and reservations	reservations-list	Units with reservations
16	Contracts and reservations	agreements-and-reservations	Contacts and recorded subject handling stages
17	Contracts and reservations	returns	Cancelled contracts / returns
18	Contracts and reservations	investments-sales-targets	Sales targets broken down by investment, year and month
19	Payments and finance	schedule-payment-list	Payment schedule (list of tranches)
20	Payments and finance	payments	List of actual payments recorded on contracts
21	Payments and finance	invoices	List of invoices
22	Payments and finance	sales-forecast	Register of active contracts per unit (prices, discounts, schedule)
23	Units and properties	realestate-list	Dictionary of unit identifiers and names
24	Units and properties	available-for-sale	Units available for sale
25	Units and properties	sold-and-not-sold	Sold and unsold report
26	Units and properties	sold-associated-units	Sold adjacent units (appurtenances) (storage units, boxes, parking spaces)

No.	Category	Endpoint name	Short description
27	Units and properties	investment-status	Current sales status of the investment
28	Activities and contact	meeting	Report of meetings with clients
29	Activities and contact	activities-list	List of client activities
30	Activities and contact	contact-transactions	Client transaction history
31	Activities and contact	incoming-call	Incoming calls (Cludo)
32	Activities and contact	mobile-call-list	Calls from the mobile app
33	Activities and contact	branch-users	List of users by branch
34	Tenant changes	realestate-changes-list	List of tenant changes
35	Technical and data warehouse	endpoints	List of endpoints available to the API key
36	Technical and data warehouse	transaction-list-anonymized	Contract register without personal data
37	Technical and data warehouse	warehouse-contracts-by-source	Summary of contracts by acquisition source
38	Technical and data warehouse	warehouse-contracts-by-source-stg	The same summary in a non-anonymized variant — with the real developer and investment names
39	Technical and data warehouse	daily-statistics-usage	Daily system usage statistics (requires a date range)

Leads and clients

1. Lead report — comparison (lead-comparison)

A summary of the number of leads and clients broken down by month. By default it covers the last 24 months: the current month and the 23 preceding ones.

Response: an array of rows, one row per month. Columns: Data, Data new, Leady, Usuniętych leadów, Powiązanych leadów, Klientów z leadów, Klientów, Klienci bezpośredni, Pierwszych spotkań, Umów rezerwacyjnych.

Available filters:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — limits the result to selected investments among those assigned to the key
- date-from/MM-RRRR — first month of the range, in **month-year** format, e.g. 01-2025
- date-to/MM-RRRR — last month of the range, e.g. 12-2025

⚠ **Important:** this is the only report that accepts dates in the MM-RRRR format (month-year) rather than RRRR-MM-DD. Passing 2026-01-01 does not raise an error — the API responds with HTTP 200 and returns a single row “Styczeń 1970” (January 1970) containing only zeros. If you see the year 1970 in the result, the date format is wrong; it does not mean there is no data.

⚠ **Important:** the response does not contain an investment column. The investment filter only narrows the input set — breaking the result down by individual investment requires a separate call for each name. In addition, using this filter removes clients with no assigned investment from the summary, so calls with and without the filter count different populations.

⚠ **Important:** the columns are calculated on different cohorts — leads by lead creation date, clients by client creation date, first meetings by meeting date, reservation contracts by contract creation date. This is not a funnel: the columns should not be subtracted from or added to each other. In particular, Klientów z leadów and Klienci bezpośredni do not add up to Klientów — a conversion is counted in the month in which it occurred, not in the month the lead was acquired. In the demo instance (panel) response for January 2025, Klientów is 16, Klientów z leadów is 0 and Klienci bezpośredni is 8. Do not calculate percentage shares from these columns.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/lead-comparision/investment/Osiedle%20Zielony%20Staw/date-from/01-2024/date-to/12-2024

2. Lead list (leads-list)

Detailed information about leads. By default it retrieves data from the last 100 days — the default range applies only when neither date-from nor date-to is provided. Providing only the start date removes the default upper limit, and providing only the end date removes the lower limit.

Response: an array of rows. Base set of columns, always present: Id Lead, Osoba, Telefon, E-mail, Źródło pochodzenia, Opis, Status, Inwestycja, Utworzono, Przyp., Skonwertowany, Ostatnia zmiana. The Status column is the **lead** status (values found in the demo instance: “Nowy”, “Skonwertowany”, “Powiązany”, “Usunięto”), not the client status.

Four columns are conditional — they appear only after the corresponding instance setting is enabled and are absent by default:

Column	Instance setting that enables it (technical name)
Lead classification (Klasyfikacja leada)	lead_show_lead_classification_field
Client classification (Klasyfikacja klienta)	power_bi_lead_list_show_client_classifications
Client status (Status klienta)	power_bi_lead_list_show_contact_status
Client status change date (Data zmiany statusu klienta)	power_bi_lead_list_show_contact_status_change_date

⚠ **Important:** conditional columns are inserted **before** the Ostatnia zmiana column, so the column number within a row is not the same across all instances. In Power BI, always refer to columns by name, never by position, and before building your model check the actual set of columns in your instance's response. If you want these columns enabled, give the technical setting names to the support team.

Available filters:

- date-from/DATA_OD — lead creation date from (YYYY-MM-DD)
- date-to/DATA_DO — lead creation date to (YYYY-MM-DD)
- lead-classifications/NAZWY_PO_PRZECINKU — lead classifications; the value Puste (empty) selects leads with no classification
- contact-classifications/NAZWY_PO_PRZECINKU — classifications of the linked client; the value Puste selects clients with no classification
- contact-classification-created-from/DATA_OD and contact-classification-created-to/DATA_DO — date the classification was assigned to the client (YYYY-MM-DD)
- contact-status/NAZWY_PO_PRZECINKU — client status; works only in instances with the “Client status” (Status klienta) column enabled
- date-modification-contact-status-from/DATA_OD and date-modification-contact-status-to/DATA_DO — client status change date; works only in instances with the “Client status change date” (Data zmiany statusu klienta) column enabled

⚠ Important: the last three items are read only when the corresponding column is enabled in the instance. When the setting is disabled, the parameter does not return an error — it is silently ignored and the response covers the full range. Rule of thumb: if the response has no “Client status” column, the contact-status filter does not work either.

⚠ Important: this data source has no investment filter to narrow the result. Enter classification and status names in the instance language, exactly as in the dictionary — the comparison is literal, with the first letter automatically capitalised. A name that is not in the dictionary does not return an error, just an empty list.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/leads-list/date-from/2024-01-01/date-to/2024-12-31

3. Client list (clients-list)

Detailed information about clients: personal data, status, classification, financing method, source, advisor, preferences, marketing consents, linked leads and UTM data. By default it shows clients from the last 3 months.

💡 **Tip** The maximum search period is 3 months. If no start date is provided, the system automatically sets it to 3 months before today. If you need to retrieve clients from a longer period, make several calls with ranges of up to 3 months each and combine the results in Power BI.

Available filters:

- date_from/DATA_OD — client creation date from (YYYY-MM-DD)
- date_to/DATA_DO — client creation date to (YYYY-MM-DD)
- contact_id/ID_KLIENTA — limits the result to a single client

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/clients-list/date_from/2024-01-01/date_to/2024-03-31

⚠ **Important:** this is the only data source in this category whose date parameter names use **underscores** — date_from and date_to. The date-from and date-to variants are not read here: the request ends with HTTP 200 and returns the default 3-month window. The name of the data source itself still uses a hyphen: clients-list.

⚠ Important: exceeding 3 months or providing a start date later than the end date returns HTTP 400 with a message in the error field: {"error": "The period between date_from and date_to cannot exceed 3 months"}.

4. Client report (clients-report)

Detailed information about clients, including the dates of their latest sales activities (meeting, e-mail, phone call), the number of days without activity, the source of first and last contact, and GDPR (RODO) consents.

Tip The maximum search period is 6 months. If no start date is provided, the system automatically sets it to 6 months before today.

Response: an object {success, timestamp, total_count, data}, where data is an array of rows with the fields: contact_id, contact_name, contact_phone, contact_email, contact_status, investment_name, user_id, user_name, contact_created_date, contact_last_meeting_date, contact_last_email_date, contact_last_call_date, days_without_activities, contact_first_source, contact_last_source, contact_rod_email_consent, contact_rod_phone_consent, contact_last_call_comment.

⚠ Important: this report does not return the client's investment preferences — they are available in the clients-list data source, in the “Preferences” (Preferencje) column. The report provides only the **dates** of the latest activities; the full list of activities is in the activities-list data source.

⚠ Important: the investment_name field may contain several investment names separated by commas, or be empty — one row is one client, not a client-investment pair. Contacts in anonymisation status are skipped by default; they are included in the result only when the include-anonymized filter is used, and then they have null in the name, e-mail, phone, comment and both consent fields.

Available filters (names always with hyphens):

- contact-created-date-from/DATA_OD — client creation date from (YYYY-MM-DD)
- contact-created-date-to/DATA_DO — client creation date to (YYYY-MM-DD)
- user-id/ID_LUB_ID_PO_PRZECINKU — ID of the advisor assigned to the client; several comma-separated IDs can be given, e.g. user-id/332,388
- contact-status/NAZWY_PO_PRZECINKU — client status names in the instance language, not IDs, e.g. contact-status/Przeterminowany; copy the name exactly from your instance's status dictionary
- investment/NAZWY_PO_PRZECINKU — limits the result to selected investments among those assigned to the key
- include-anonymized/1 — includes contacts in anonymisation status, with personal data nulled and an additional is_anonymized field; works only in instances with the power_bi_clients_report_include_anonymized setting enabled

Filters can be passed either as segment pairs in the URL path or as query string parameters — the result is identical.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/clients-report/contact-created-date-from/2026-03-01/contact-created-date-to/2026-08-31/user-id/332

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/clients-report?contact-created-date-from=2026-03-01&contact-created-date-to=2026-08-31

⚠ **Important:** names with underscores — `contact_created_date_from`, `contact_created_date_to`, `user_id`, `contact_status` — are not read by the API. A request using them ends with HTTP 200 and returns the full default 6-month window, so the lack of filtering is easy to miss. The demo instance shows this in the numbers: a call without filters returned 110 records, a call with hyphenated filters — 26 records, and the same call with underscores — 110 again. The name of the data source itself also uses a hyphen: `clients-report` works, `clients_report` returns HTTP 404.

⚠ Important: providing the name of an investment that is not assigned to the key, or a non-existent name, ends with HTTP 403 and an empty body — before the report is even run. The investment filter, however, does not remove clients with no assigned investment from the result: they are always returned, with an empty `investment_name`. Therefore totals calculated separately for each investment will not add up to the unfiltered result — the same “no investment” records will be repeated in every call.

5. Lead report (leads-report)

Information about leads, including the status change history of the linked client and the dates of the latest sales activities.

💡 **Tip** The maximum search period is 6 months. If no start date is provided, the system automatically sets it to 6 months before today.

Response: an object `{success, timestamp, total_count, data}`, where `data` is an array of rows with the fields: `lead_id`, `lead_status`, `lead_removal_reason`, `contact_id`, `contact_name`, `contact_status`, `contact_status_history`, `lead_phone`, `lead_email`, `investment_name`, `user_id`, `user_name`, `contact_created_date`, `lead_created_date`, `lead_conversion_date`, `contact_last_meeting_date`, `contact_last_email_date`, `contact_last_call_date`, `lead_source`, `utm_data`, `lead_rodo_email_consent`, `lead_rodo_phone_consent`, `contact_last_call_comment`.

The `contact_status_history` field is a nested array of `{contact_status_date, contact_status}` objects, sorted in ascending order by date — it lets you reconstruct the client's status path. In Power BI the column must be expanded (Expand), otherwise it will appear as List; for a lead with no linked client it is an empty array. The `utm_data` field is text in query string format (`utm_campaign=...&utm_source=...`), not an object, and has the value null when no UTM tag was saved.

⚠ Important: the report returns only the **dates** of the last meeting, e-mail and phone call. The full list of activities is in the `activities-list` data source.

Available filters (names always with hyphens):

- `lead-created-date-from/DATA_OD` — lead creation date from (YYYY-MM-DD)
- `lead-created-date-to/DATA_DO` — lead creation date to (YYYY-MM-DD)
- `user-id/ID_LUB_ID_PO_PRZECINKU` — ID of the advisor assigned to the **lead**, not to the client; several comma-separated IDs can be given
- `investment/NAZWY_PO_PRZECINKU` — limits the result to selected investments among those assigned to the key

Filters can be passed either as segment pairs in the URL path or as query string parameters — the result is identical.

Example:

`https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/leads-report/lead-created-date-from/2026-03-01/lead-created-date-to/2026-08-31`

Page 16 / 53

© 2026 voxCRM <kontakt@voxdeveloper.com> | 14-10-2026 16:45

URL: <https://voxcrmdu.com/content/0/246/en/power-bi-endpoints---list-of-available-data-sources.html>

⚠ **Important:** names with underscores — `lead_created_date_from`, `lead_created_date_to`, `user_id` — are not read by the API. A request using them ends with HTTP 200 and returns the full default 6-month window, without any error message. On the demo instance: a call without filters returned 196 records, a call with hyphenated filters — 15 records, and the same call with underscores — 196 again. The name of the data source itself also uses a hyphen: `leads-report` works, `leads_report` returns HTTP 404.

⚠ Important: providing the name of an investment that is not assigned to the key, or a non-existent name, ends with HTTP 403 and an empty body. The investment filter does not remove leads with no assigned investment from the result — they are always returned, with an empty `investment_name`, so the results for individual investments will not add up to the unfiltered result.

6. Leads linked to contracts (lead-list-agreements)

A list of leads linked to contracts. It combines lead data with information about the contracts concluded.

7. Google Ads conversions (gclid-conversions)

A list of leads with Google Ads conversion data (GCLID). It is used to integrate offline conversions with Google Ads. It returns converted leads, i.e. leads linked to a client, that have a non-empty GCLID field.

💡 **Tip** This endpoint is particularly useful for measuring the effectiveness of Google Ads campaigns — it lets you track which ad clicks turned into real sales contacts.

Response: an array of rows with five fields: `gclid`, `conversion_name` (fixed value “Konwersja offline”), `conversion_time` (conversion date and time), `email_address`, `investment` (may be null).

⚠ Important: lead classification is not a default condition of this data source — the response does not contain a classification field at all. If you need only leads with a selected classification, e.g. “Ciepły Lead” (Warm Lead), add the filter `lead-classifications/Ciep%C5%82y%20Lead`.

Available filters:

- `conversion-date-from/DATA_OD` — date of lead-to-client conversion from (YYYY-MM-DD)
- `conversion-date-to/DATA_DO` — conversion date to (YYYY-MM-DD)
- `from/DATA_I_GODZINA` — conversions from the given moment, format RRRR-MM-DD GG:MM:SS (encode the space as %20); used for incremental retrievals
- `investments/NAZWY_PO_PRZECINKU` — investment names; the parameter is **plural**
- `lead-classifications/NAZWY_PO_PRZECINKU` — lead classifications, e.g. `lead-classifications/Ciep%C5%82y%20Lead`

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/gclid-conversions/conversion-date-from/2026-01-01/conversion-date-to/2026-08-31

⚠ **Important:** this data source does not use the standard date-from and date-to names. Providing them does not return an error — the API responds with HTTP 200 and returns the entire history without filtering. To narrow by investment, use the plural `investments`; the singular `investment` parameter is checked by this data source primarily against the key's permissions — a name outside the permissions ends the request with HTTP 403 — and does not remove conversions with no

assigned investment from the result.

△ Important: an incorrect format of the from parameter returns HTTP 400 with the message {"error": "Invalid date format for parameter from. Required format: YYYY-MM-DD HH:MM:SS"}.

8. Conversions for Google Ads (google-ads-conversions)

An export of leads in a format prepared for importing offline conversions and “enhanced conversions for leads” into Google Ads. One record = one lead linked to one investment. Instead of plain-text contact details, it returns SHA-256 hashes (hashed_email, hashed_phone_number), marketing consents, the conversion type and the GCLID, if the lead has one.

△ **Important:** this is NOT the same report as gclid-conversions (item 7). Differences: gclid-conversions returns five fields, including the PLAIN-TEXT e-mail address (email_address), and covers only leads with a GCLID filled in; google-ads-conversions returns twelve fields, covers all leads assigned to investments, including those without a GCLID, and does not reveal the address.

Response: an array of rows with the fields: investment, conversion_type (values: form-kontakt, klik-tel, call-page-koniec-rozmowy), conversion_name, conversion_time, hashed_email, hashed_phone_number, ad_user_data, ad_personalization, lead_quality, gclid, crm_id, extra_info.

Available filters:

- date-from/DATA_OD — lead creation date from (YYYY-MM-DD)
- date-to/DATA_DO — lead creation date to (YYYY-MM-DD)
- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection (names exactly as in the CRM)

Without dates, the report returns the entire lead history. A date in a format other than YYYY-MM-DD ends with HTTP 400 and the body {"error": "Invalid date format, expected Y-m-d"}.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/google-ads-conversions/date-from/2026-08-01/date-to/2026-08-31

Limitations:

- conversion_time is the lead creation date, not the date of contact or sale. It contains no time zone — importing into Google Ads requires adding one.
- A lead assigned to several investments appears in several rows with the same crm_id — count leads as COUNT DISTINCT on crm_id.
- gclid is sometimes empty; records with neither a GCLID nor hashes are useless for import — filter them out on your side.
- ad_user_data and ad_personalization come from a single marketing consent and always have the same value.
- The investment and lead_quality fields are written in lowercase, but the investment filter requires the original spelling from the CRM; an unknown name ends with HTTP 403.
- A phone number without a country prefix is normalised to +48 before hashing — save foreign numbers in the CRM with their prefix.

9. Lead history from the RynekPierwotny.pl portal (leads-activities-history-rp)

A feedback report for the RynekPierwotny.pl portal. For each lead with an RP identifier filled in, it

Page 18 / 53

returns contact details, the client status in the CRM, the status in the RP dictionary and the dates of key events: lead creation and conversion, last phone call, last meeting, status change, oral reservation, reservation contract, preliminary contract, developer contract and final contract.

Response: an array of rows with the fields: rynekpierwotnyid, lead_id, contact_name, contact_phone, contact_email, contact_status, rp_status, lead_created_date, lead_conversion_date, contact_last_call_date, contact_last_meeting_date, last_status_change, oral_reservation_date, reservation_agreement_date, preliminary_agreement_date, developer_agreement_date, final_agreement_date.

Available filters:

- date-from/DATA_OD — lead creation date from (YYYY-MM-DD), name with a **hyphen**
- date-to/DATA_DO — lead creation date to (YYYY-MM-DD), name with a **hyphen**
- contact_status/ID_STATUSÓW_PO_PRZECINKU — client status, name with an **underscore**; the value -1 means no client status
- rp_status/KODY_PO_PRZECINKU — status in the RynekPierwotny dictionary, name with an **underscore**

The default range is the last 12 months — dates are optional. A range longer than 12 months or a reversed range ends with HTTP 400 and the body {"error":"Invalid date-from or date-to parameter, or range exceeds 12 months"}.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/leads-activities-history-rp/date-from/2026-01-01/date-to/2026-08-31

⚠ **Important:** this report mixes naming conventions — dates with a hyphen, statuses with an underscore. Do not use the names interchangeably: a variant written differently is silently ignored, and the response has HTTP status 200.

Limitations:

- The report covers ONLY leads with a RynekPierwotny.pl identifier. On an instance without this integration the response is an empty array [] — this is neither a key error nor a report failure.
- There is no investment filter: RP leads from the entire instance are returned.
- The same RP identifier may appear for several leads — do not treat it as a unique key.
- A phone number consisting only of digits is returned as a number; load this column as text so that leading zeros are not lost.
- The response contains first and last name, phone number and e-mail — do not publish it in public materials.

10. Company records (company)

A complete list of companies saved in the CRM: registration data (NIP tax ID, REGON, KRS), contact details, addresses, type and industry, assigned account manager and the descriptive field opis_firma. It covers both corporate clients and contractors or subcontractors — they are distinguished by the kontrahent_firma field (Boolean value true/false).

Response: an array of rows. Keys are mostly Polish (written in snake_case with the _firma suffix), and the identifier is account_id. On the demo instance the report returns 23 companies with 30 fields each.

Selected fields: account_id, uuid_firma, nazwa_firma, nazwa_sktrocona_firma, nip_firma, regon_firma, krs_firma, email_firma, nr_telefonu_firma, tel_kom_firma, url_strony_firma, adres_siedziba_firma, adres_korespondencyjny_firma, typ_wlasnosci_firma, typ_firma, branza_firma,

obszar_aktywnosci_firma, wielkosc_firma, tagi_firma, kontrahent_firma, instytucja_finansowa_firma, kategorie_zgloszen_kontrahent_firma, przypisana_do_firma, opis_firma.

Available filters: none. The report always returns the entire company register, without pagination, date range or breakdown by investment. The /date-from/, /date-to/ and /investment/ segments are ignored — but an investment name that is unknown or not assigned to the key still ends with HTTP 403.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/company

⚠ **Important:** the response contains sensitive data — numer_konta_bankowego_firma, numer_karty_bankowej_firma and numer_podatkowy_firma, and for sole proprietorships also personal data. Do not publish raw responses from this data source and do not pass them to external tools.

Limitations:

- No company-person link: the endpoint does not return contact_id, and clients-list and clients-report do not return account_id. The API therefore offers no way to join company records with contacts.
- Addresses are concatenated into a single text (street and number, postal code, city, voivodeship, country). City and postal code must be extracted on your own.
- Multi-value fields (industry, area of activity, phone numbers, fax numbers, ticket categories) are a single comma-separated text, and the dictionary names themselves may also contain commas — do not split them automatically.
- Empty text and the value "0" are returned as null; url_strony_firma is sometimes the placeholder "http://" — treat it as a missing value.
- Only kontrahent_firma and instytucja_finansowa_firma are Boolean values; all other fields are text or null.
- No creation or modification date — this data source is not suitable for incremental synchronisation.
- Row order is not guaranteed; sort on your side.

Contracts and reservations

11. Contract register and contract values (agreements-registry)

The register describes contracts and their values. Each row is a row of the contract report; check the uniqueness of the contract UUID before counting. The same unit may appear at several stages, e.g. in a reservation contract and in a development contract.

Response: An array of rows. Many key names come from the report's translations.

Dates and status: the Data field is the contract date. The Podpisana column, if present, is a date, not a boolean value. Do not infer that a contract has been signed merely from the presence of a date — check Status.

URL path parameters — selected scope:

Parameter	Value and meaning	Required
investment	investment names; URL-encode the values	No

Parameter	Value and meaning	Required
agreement_date_from	YYYY-MM-DD; contract date from (underscores)	No
agreement_date_to	YYYY-MM-DD; contract date to (underscores)	No
first-agreement-date-from	YYYY-MM-DD; date of the first contract on the unit, from — this name uses hyphens	No
first-agreement-date-to	YYYY-MM-DD; date of the first contract on the unit, to — this name uses hyphens	No
include_canceled	1 adds canceled contracts to the result	No
status_id_filter	Comma-separated IDs of contract stages — filters the Typ umowy column, not the Status column. Base dictionary: 1 Zainteresowany (Interested), 2 Umowa rezerwacyjna (Reservation contract), 3 List intencyjny (Letter of intent), 4 Umowa deweloperska (Development contract), 5 Umowa przedwstępna (Preliminary contract), 6 Umowa końcowa (Final contract), 7 Obsługa posprzedażowa (After-sales service), 8 Rezygnacja (Withdrawal), 9 Akt notarialny (Notarial deed), 10 Odbiór (Handover), 11 Umowa zmian lokatorskich (Tenant changes contract), 12 Umowa najmu (Lease contract). Stage names are sometimes changed in a given deployment.	No
last_modified	YYYY-MM-DD; returns rows changed since the given date — includes changes to the contract, shares and contact data (incremental synchronization)	No
remote-sync	1 adds the columns “Dane klientów” (Client data — buyers' contact details), “Contiguities” and “Discount”; this name uses a hyphen . Use only where processing	No

Parameter	Value and meaning	Required
	personal data is justified.	

△ The parameter names of this report mix conventions: agreement_date_from, agreement_date_to, include_canceled, status_id_filter and last_modified use underscores, while first-agreement-date-from, first-agreement-date-to and remote-sync use hyphens. A variant written in the other convention is ignored without an error, and the response still returns HTTP status 200.

The Status column (W toku / Podpisana / Anulowana — In progress / Signed / Canceled) cannot be narrowed down by any parameter of this endpoint — filter it after retrieving the data.

Example URL:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/agreements-registry/agreement_date_from/2026-08-01/agreement_date_to/2026-08-31

Selected fields: the table is not the full response schema. Types describe the expected interpretation; confirm the JSON type and whether the value can be empty on a sample. A missing required field stops the calculations.

Field	Expected form	Meaning
Nr	number string	transaction_id; not the ordinal number (L.P.)
UUID	string	Contract UUID
Typ umowy	string	Contract stage/type
Status	string	Transaction status, independent of the contract type
Data	string	Contract date; Podpisana is a duplicate of the date, not a boolean
ID lokalu	number string	Internal unit ID
UUID lokalu	string	Unit UUID; a different space from the contract UUID
ID Klient	string number	Aggregated client identifiers; do not assume a single ID
Wartość BRUTTO	number string	Value of the main unit; not the whole contract
Końcowa BRUTTO	number string	Contract value; a different measure from Wartość BRUTTO

Illustrative JSON — fictitious values, selected fields:

```
[
  {
    "UUID": "11111111-1111-4111-8111-111111111111",
    "Typ umowy": "Umowa deweloperska",
    "Status": "Podpisana",
    "Data": "2026-08-12"
  },
  {
    "UUID": "22222222-2222-4222-8222-222222222222",
    "Typ umowy": "Umowa rezerwacyjna",
    "Status": "Podpisana",
    "Data": "2026-08-19"
  }
]
```

How to interpret the example: For these two records, the number of unique signed contracts is 2. This does not mean that two different units were sold. For a monthly series, define the set of contract types, select the Podpisana (Signed) status and group by Data. An unknown type or status should be visible in the quality check.

Limitations:

- By default, the report **omits canceled contracts**. On the demo instance (panel), a call without a filter returns 610 rows, and with the include_canceled/1 parameter — 1472 rows, i.e. 862 more canceled contracts. Without this parameter, you cannot count either withdrawals or the full contract volume from this report.
- The stages of a single sale may appear as different contracts. The sum of their values is neither revenue nor the value of unique units sold.
- Do not subtract the number of returns from the number of arbitrary contracts without aligning the populations and identifiers.
- Polish keys depend on translations; confirm them with a sample from your instance.

12. Contract list export (transaction-list)

An export of the contract list in the same shape the user sees in the Contracts (Umowy) module. It includes additional columns: "Budynek" (Building) and "Cena całkowita netto" (Total net price).

⚠ Important: contrary to its name, the report **does not return all contracts**. Without the all-agreements/1 parameter it omits canceled contracts — on the demo instance this is 629 rows without the parameter versus 1505 with it, meaning the default call shows less than 42% of contracts. Without all-agreements/1, the "Data anulowania umowy" (Contract cancellation date) column is empty in all rows, and "Status umowy" (Contract status) never takes the value "Anulowana" (Canceled). If you build any volume indicator on this report, add this parameter and filter out canceled contracts on your side.

💡 Tip The "Cena całkowita netto" column is visible only when the "Enable additional columns for data export" (Włącz dodatkowe kolumny do eksportu danych) option is enabled in the contract list settings. The "Budynek" column is always present.

Available filters:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection (URL-encode the values)
- all-agreements/1 — includes canceled contracts; without this parameter the report does not return them
- first-agreement-date-from/DATA_OD — date of the **first** contract on the unit, from (hyphens, YYYY-MM-DD)
- first-agreement-date-to/DATA_DO — date of the **first** contract on the unit, to (hyphens, YYYY-MM-DD)

The endpoint has no filter by the date of the contract itself — the date-from and date-to segments are ignored here without an error. The first-agreement-date-* filter works on the date of the first contract concluded for a given unit, so a development contract with a later date will also fall into the window if the reservation of that unit is within the range. Narrow down by the “Data podpisania umowy” (Contract signing date) column on your side.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/transaction-list/all-agreements/1/investment/Osiedle%20Zielony%20Staw

Example with a first contract date range:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/transaction-list/first-agreement-date-from/2026-01-01/first-agreement-date-to/2026-08-31

13. Contract list (agreements-list)

Detailed information about contracts, together with nested client and unit data. Among the contract reports, this one returns the most complete set of technical identifiers: transaction_id, client_crm_id, account_crm_id and agreement_local_crm_id. The response is an object with a transactions array.

By default, the report omits canceled contracts. On the demo instance, a call without a filter returns 629 records, and with status/inprogress,signed,canceled — 1505, including 876 with the status “Anulowana” (Canceled).

Available filters:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection (URL-encode the values)
- status/STATUSY_PO_PRZECINKU — contract status; values must be given **in English only**: inprogress (W toku), signed (Podpisane), canceled (Anulowane). Case-insensitive, multiple comma-separated values.
- agreement_created_date_from/DATA_OD — date the contract was **created** in the CRM, from (underscores, YYYY-MM-DD)
- agreement_created_date_to/DATA_DO — date the contract was **created** in the CRM, to (underscores, YYYY-MM-DD)

⚠ Important: Polish status names are ignored — silently, without any error message. On the demo instance, the URL with status/signed returned 157 records, while the same URL with status/Podpisane returned 629 records, i.e. the full set of non-canceled contracts, exactly the same as a call without any filter. If your “signed contracts” report returns a suspiciously large number of rows, first check whether the status value is in English.

The date filters work on the date the record was created in the CRM (agreement_created_date), not on the contract date (agreement_date). An invalid date format results in an HTTP 400 response with the message “Invalid date format. Required format: YYYY-MM-DD”.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/agreements-list/status/inprogress,signed,canceled/agreement_created_date_from/2026-01-01/agreement_created_date_to/2026-08-31

14. Units with the current contract stage (agreements)

The report returns one row per **unit** — with the contract stage the unit is currently at (the `agreement_type` field). It is not a list of contracts: earlier stages of the same unit, e.g. a reservation contract preceding the development contract, do not have separate rows. This is exactly what distinguishes it from the other reports in this section: `agreements-registry`, `transaction-list` and `agreements-list` count contracts, while `agreements` counts units.

Response: an object with a data array. A record contains only descriptive fields: `investment`, `rooms`, `first_agreement_date`, `agreement_type`, `agreement_status`, `realestate_type`, `realestate_area`, `realestate_m2_price`, `assigned_to` and the nested contiguities array (adjacent units (appurtenances) with prices).

⚠ **Important:** the report **contains no identifiers** — contrary to earlier descriptions, it has neither `transaction_id`, nor `client_crm_id`, nor a unit ID or UUID. A record consists of exactly the ten fields listed above. Joining with other reports is possible only by investment name, and that is not a stable key. If you need contract and client identifiers, use `agreements-list`.

Canceled contracts are omitted — the response never contains `agreement_status = "Anulowana"`. The set of stages included in the report is hard-coded and differs between deployment versions, so before you count anything, check on a sample from your instance which `agreement_type` values are actually returned. On the demo instance, 458 rows are returned: `Umowa rezerwacyjna 265`, `Umowa deweloperska 155`, `Umowa końcowa 21`, `Umowa przedwstępna 17`; the `agreement_status` field contains only `"W toku"` (In progress, 372) and `"Podpisana"` (Signed, 86).

`first_agreement_date` is the date of the **earliest** contract on this unit, while `agreement_type` and `agreement_status` describe the **current** contract — the two values may come from different contracts.

Available filters:

- `investment/NAZWY_INWESTYCJI_PO_PRZECINKU` — investment selection (URL-encode the values)
- `date_from/DATA_OD` — current contract date from (underscores, YYYY-MM-DD)
- `date_to/DATA_DO` — current contract date to (underscores, YYYY-MM-DD)
- `sources/NAZWY_ZRÓDEŁ_PO_PRZECINKU` — source of origin of the **contract** (the contract source dictionary, not the contact source used in the `agreements-and-reservations` report)

This report uses the names `date_from` and `date_to` with an underscore. The `date-from / date-to` variant is ignored without an error — the response then covers the entire history, despite HTTP status 200. The date filter works on the current contract date, so the result may include rows in which `first_agreement_date` is earlier than the given start date. A source name outside the dictionary does not disable the filter — the response is then empty.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/agreements/investment/Osie dle%20Przykładowe/date_from/2026-01-01/date_to/2026-08-31

15. Units with reservations (reservations-list)

A list of units with reservations. The response is an array of rows with Polish key names: Id, Lokal, Metraż m2, Inwestycja, Etap, Status, Typ rezerwacji, Wstępna cena rezerwacyjna, Doradca, Data utworzenia, Od, Wygasa. Without the status parameter, the report returns active reservations only.

Available filters:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection (URL-encode the values)
- status/STATUSY_PO_PRZECINKU — reservation status; values must be given **in English only**: active, canceled, expired (case-insensitive, multiple comma-separated values)
- date-from/DATA_OD — reservation creation date from (hyphens, YYYY-MM-DD, inclusive condition)
- date-to/DATA_DO — reservation creation date to (hyphens, YYYY-MM-DD, inclusive condition)

⚠ **Important:** Polish status names (Aktywne, Anulowane, Wygasłe) **are not recognized**. A request with a Polish value ends with HTTP status 200 and an empty array — without any error message. On the demo instance, the URL with status/Aktywne,Anulowane,Wygasle returned **0 records** (a 2-byte response), while the same URL with status/active,expired,canceled returned **490 records** (Wygasłe/Expired 254, Anulowane/Canceled 236, active 0).

In the response, the Status field is translated into the instance language (Aktywne / Wygasłe / Anulowane), so the input and output values have different forms. Do not copy values from the response back into the URL.

The endpoint does not support the all-agreements parameter. Unrecognized path segments are ignored without an error — HTTP status 200 does not confirm that the filter worked. The date filters work on the reservation creation date in the CRM, not on its start date (the Od field) or expiry date (the Wygasa field); the underscore variant (date_from / date_to) is ignored here.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/reservations-list/status/active,expired,canceled

Example with a reservation creation date range:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/reservations-list/status/active,expired,canceled/date-from/2026-01-01/date-to/2026-08-31

16. Clients and registered unit handling stages (agreements-and-reservations)

The report combines contacts and the registered handling stages of a subject, e.g. a unit. The model groups data by contact and subject. A single contact may appear in multiple rows; this is neither a complete list of all CRM clients nor a list of contracts alone.

Response: An object with success and a data array. Check success before reading the rows.

Important date distinction: date_from and date_to refer to the creation date of the subject record in the report model (subject_to_order.date_created). They do not filter the contact acquisition date or the signing dates of all contracts shown. Fields ending in _created_date describe the creation of a stage in the CRM; contract date fields have a different meaning.

Buying client — canonical definition: a buying client is a unique, non-empty contact_id for which at least one of the fields preliminary_agreement_date, developer_agreement_date or notarial_deed_date is populated in at least one row. A reservation alone does not make a client a buying client. Count this as COUNT DISTINCT on contact_id, not as the number of rows. On the demo instance, 465 rows correspond to 226 clients, of whom 117 meet this definition.

Source: source is an attribute of the contact (contact.source_of_origin_id → contact_source_of_origin.name), not of the contract or unit — so one contact_id has exactly one source value, regardless of how many rows it appears in. On the demo instance, none of the 226 contacts had two different sources, so a “source conflict” does not occur in practice and there is no need to build a rule for it. The real issue is the catch-all value “Inne” (Other): 38 of 226 clients have it, making it the second-largest “source” in the summary — in channel analysis, show it as a separate, unassigned item rather than distributing it among channels. Exclude empty IDs from the count of unique contacts and show them as a separate count of quality errors (on the demo instance there were no empty contact_id values).

URL path parameters — selected scope:

Parameter	Value and meaning	Required
investment	investment names; URL-encode the values	No
date_from	YYYY-MM-DD; sto.date_created	No
date_to	YYYY-MM-DD; sto.date_created	No
sources	names of contact sources of origin, comma-separated, exactly as in the instance dictionary (case-insensitive); URL-encode the values	No

A source name outside the dictionary does not disable the filter — the response is then empty with HTTP status 200. The same sources parameter in the agreements and lead-list-agreements endpoints uses a different dictionary (contract source), so do not copy values between reports.

Example URL:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/agreements-and-reservations/investment/Osiedle%20Przykladowe

Selected fields: the table is not the full response schema. Types describe the expected interpretation; confirm the JSON type and whether the value can be empty on a sample. A missing required field stops the calculations.

Field	Expected form	Meaning
contact_id	number string null	Contact ID, not lead ID
investment	string	Investment name
property	string	Unit name; the endpoint does not export the subject_id used for grouping
source	string null	Current contact source: contact.source_of_origin_id → contact_source_of_origin.name; one value per contact

Field	Expected form	Meaning
contact_method	string null	Contact method; not a marketing source
reservation_created_date	string null	Reservation creation date
developer_agreement_created_date	string null	Date the development contract was created in the CRM
preliminary_agreement_date	string null	Preliminary contract date; one of the three fields that define a buying client
developer_agreement_date	string null	Development contract date; one of the three fields that define a buying client
notarial_deed_date	string null	Notarial deed date; one of the three fields that define a buying client
current_gross_price	string null	Formatted price of the subject; not client revenue

Illustrative JSON — fictitious values, selected fields:

```
{
  "success": true,
  "data": [
    {
      "contact_id": "101",
      "investment": "Osiedle Przykładowe",
      "property": "A/01",
      "source": "Polecenie",
      "contact_method": "Telefon",
      "developer_agreement_date": "2026-08-12"
    },
    {
      "contact_id": "101",
      "investment": "Osiedle Przykładowe",
      "property": "A/02",
      "source": "Polecenie",
      "contact_method": "Telefon",
      "developer_agreement_date": null
    },
    {
      "contact_id": "102",
      "investment": "Osiedle Przykładowe",
      "property": "A/03",
      "source": "Inne",
      "contact_method": "Formularz WWW",
      "developer_agreement_date": null
    }
  ]
}
```

```
}
```

How to interpret the example: The example has 3 rows and 2 unique contacts, and there is one buying client — contact 101, because one of its rows contains a development contract date. Contact 102 has the source “Inne” (Other), i.e. an item not assigned to any channel; do not replace it with the contact_method value, because Formularz WWW (Web form) describes the contact method, not the acquisition channel. These numbers do not determine the number of transactions or the channel conversion.

Limitations:

- COUNT(*) counts links, not clients. For clients, use COUNT DISTINCT on non-empty contact_id.
- A later stage does not prove an earlier one. Do not add reservations to clients who only have a notarial deed.
- The lack of a stable unit ID in the response makes safe joining with other reports difficult.
- Do not sum current_gross_price per contact as sales value without checking for duplication.

17. Canceled contracts / Returns (returns)

A list of **returns**, i.e. contracts that first had the status “Podpisana” (Signed) and were subsequently canceled. It contains data on the unit, the contract and cancellation dates, and the contract value (net/gross).

Response: an array of rows with eight fields: ID umowy, Data anulowania umowy, Data podpisania umowy, Typ umowy, Numer lokalu głównego na umowie, Inwestycja, Wartość umowy netto, Wartość umowy brutto. Row order is not part of the contract — if you need it, sort the data on your side.

⚠ **Important:** the report **does not contain all canceled contracts** — a contract canceled directly from the “W toku” (In progress) state does not appear here. On the demo instance, this is 173 returns versus 862 canceled contracts in total visible in agreements-registry with include_canceled/1. You can retrieve the full set of canceled contracts from agreements-registry with the include_canceled/1 parameter or from agreements-list with status/canceled.

Scope: always the entire instance history. The endpoint does not support any date filters, so it returns all returns since the data began, not a selected period. On the demo instance, 173 records cover cancellation dates from 2024-12-06 to 2026-09-03 and signing dates from 2017-09-26 to 2026-08-20. Cut out the time window on your side, by the Data anulowania umowy field, and record next to the indicator which date you chose.

Do not subtract this report from the contract register “by eye”. The response contains neither a contract UUID nor a unit UUID, and the only identifier, ID umowy, has no match in the default agreements-registry response — on the demo instance, none of the 173 values appear among the 610 rows of the Nr column, because the register omits canceled contracts by default. To compare the two reports, retrieve the register with include_canceled/1 and only then join by Nr; without this step, there is no key that allows returns to be unambiguously subtracted from the number of contracts.

The set includes all contract types present in the deployment, including non-sales ones — on the demo instance, among others, Umowa najmu (Lease contract, 11) and Umowa zmian lokatorskich (Tenant changes contract, 2), alongside development (68), reservation (52), final (31) and

preliminary (9) contracts. For sales indicators, narrow down the list of types after retrieving the data.

Watch out for two pitfalls in this report. First, Data anulowania umowy is sometimes **earlier** than Data podpisania umowy — on the demo instance this applies to 7 of 173 records. Do not calculate contract lifetime from the difference of these dates without checking for negative values. Second, Numer lokalu głównego na umowie may be returned as a number (e.g. "5.20" as 5.2), so when joining with other reports, cast it to text and expect the loss of trailing zeros.

Available filters:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection (URL-encode the values). The endpoint does not support any date filters.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/returns/investment/Osiedle%20Zielony%20Staw

18. Sales targets — plan and achievement (investments-sales-targets)

The only source of **sales targets** in the API. For each investment and each month for which a target has been entered in the CRM, it returns the planned number of contracts, contracts actually signed, contracts in progress, and the value of signed contracts — monthly and cumulatively. Separately for unit sales contracts (primary_transactions) and for tenant changes contracts (tenant_change_transactions).

Response: a single object with an investments array, **not an array of rows**. The structure is nested: investment → years → months → two blocks of numbers. In Power BI, expand investments, years, months in turn, and then the primary_transactions and tenant_change_transactions records.

Illustrative JSON — fictitious investment name, figures from the demo instance:

```
{
  "investments": [
    {
      "city": "Łódź",
      "investment_id": 12,
      "investment_name": "Osiedle Przykładowe",
      "years": [
        {
          "year": 2025,
          "months": [
            {
              "month": 1,
              "primary_transactions": {
                "goal_monthly": 5,
                "signed_monthly": 1,
                "planned_monthly": 1,
                "goal_cumulative": 15,
                "signed_cumulative": 18,
                "value_monthly_pln": 287841.46,
                "value_cumulative_pln": 4630702.33
              },
              "tenant_change_transactions": {
                "goal_monthly": 1,
                "signed_monthly": 0,
                "planned_monthly": 0,
                "goal_cumulative": 4,

```

```

        "signed_cumulative": 2,
        "value_monthly_pln": 0,
        "value_cumulative_pln": 2650
      }
    ]
  }
}

```

Field meanings (identical in both blocks):

Field	Meaning
goal_monthly	monthly target entered manually in the CRM (plan)
signed_monthly	contracts signed with a contract date in this month
planned_monthly	contracts in progress, not yet signed, dated in this month; this is not the plan
goal_cumulative	sum of targets since the first month with a target
signed_cumulative	contracts signed, cumulative
value_monthly_pln / value_cumulative_pln	value of signed contracts in PLN (a number, not text)

⚠ **Important:** treat a month in which goal_monthly is 0 as “target not set”, not as a zero target. The same applies to a month that is missing from the response altogether — a gap does not mean zero sales. Do not calculate plan achievement percentages from such months and do not include them in averages; show them as missing data.

Available filters:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection; the filter narrows the result. An investment without entered targets returns {"investments":[]}

The report has no date filters — it returns the entire target history; narrow the range by the year and month fields on your side.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/investments-sales-targets

Limitations:

- An investment without any entered target does not appear in the result at all.
- signed_cumulative and value_cumulative_pln also include contracts signed before the first month with a target, so cumulative achievement can exceed 100% as early as the first

month. For achievement percentages, compare monthly values or subtract the baseline from the first month.

- The definition of a signed contract is narrower here than in agreements-registry: only transactions with the status “Podpisana” (Signed) whose unit is at the development contract or preliminary contract stage are counted, and the month is taken from the contract date. Reservations and notarial deeds are not included in the figures.
- tenant_change_transactions refers to tenant changes contracts and is counted under a separate condition; do not add these values to unit sales values.

Payments and finance

19. Tranche schedule and balance (schedule-payment-list)

The report presents the payment schedule: the tranche amount, its due date, and the settled and outstanding amounts. It shows the settlement status of tranches within the retrieved range. Before summing, check that the tranche identifier is unique.

Response: a flat array of rows, 28 columns. Some key names are translated into the instance (panel) language (Termin, Kwota, Zapłacono), while others remain in English (Schedule UUID, Client / Company full data).

Default range: the date-from and date-to filters apply to the **tranche due date** — the Termin field (due_date in the database), i.e. the planned payment date, not the date the funds were received. Omitting date-from **does not return the complete data set:** the server then substitutes a date one year back, so older arrears disappear from the response without any warning. Omitting date-to sets no upper limit, so the response also contains the entire future payment plan.

⚠ Important: the two ends of the range cut off different things, and both must be set deliberately. date-from cuts off arrears older than one year, date-to cuts off future tranches of the plan. Measurement on the demo instance from 10.09.2026:

Retrieval	Number of tranches	Due date range	Overdue tranches (due before the retrieval date, Pozostało greater than 0)
no date filters (default one year back)	703	2025-09-11 – 2028-07-25	438 totalling 30,561,646.16 PLN
date-from/2015-01-01 (full history)	1138	2018-08-16 – 2028-07-25	789 totalling 48,489,995.26 PLN
date-to set to the retrieval date	496	2025-09-11 – 2026-09-10	438 totalling 30,561,646.16 PLN

On the demo instance, the default one-year due-date window hid **351 overdue tranches totalling 17,928,349.10 PLN**, i.e. 37% of all arrears. If you are calculating arrears, explicitly provide an early start date — move date-from back to the beginning of the investment's history. Setting date-to to the retrieval date does not change the arrears, but removes the entire future plan from the response

(here 207 tranches) — do not do this if you are forecasting inflows.

hide-lower is enabled by default — the filter is active for every value except a literal 0, including when the parameter is absent altogether. With the filter enabled, only the tranches of the contract with the highest stage are returned for each unit; tranches of lower-stage contracts superseded by a later contract (for example the reservation fee after the development contract has been concluded) are hidden. On the demo instance, hide-lower/0 returns 800 rows instead of 703. Set hide-lower/0 if you also settle fees from earlier-stage contracts.

only-not-paid/1 narrows the response to unsettled tranches (636 of 703 rows on the demo instance). This filter restricts the population, so do not use it to calculate the share of paid amounts in the whole schedule — the denominator must be retrieved separately, without this filter.

When comparing two retrievals, always use the same date range and the same hide-lower and only-not-paid settings. For each metric, record the actual range of retrieved due dates.

Tranche identifier: the row key is the Schedule UUID column — it contains the UUID of a **single tranche** (transaction_schedule_payment.uuid), not the UUID of the whole schedule. The label in the response is untranslated and reads exactly "Schedule UUID", also on a Polish instance.

On the demo instance, 703 rows had 703 distinct Schedule UUID values across 255 distinct contracts. Do not replace this field with either the ID umowy column or Transaction UUID — one contract has many tranches and both of these columns repeat across many rows.

URL path parameters — selected range:

Parameter	Value and meaning	Required
investment	comma-separated investment names; URL-encode the values	No
date-from	tranche due date from (YYYY-MM-DD); defaults to one year before the retrieval date	No
date-to	tranche due date to (YYYY-MM-DD); no upper limit by default	No
only-not-paid	1 — unsettled tranches only	No
hide-lower	enabled by default; only the literal value 0 disables it	No
agreement-type	comma-separated contract type names, in the instance language (e.g. Umowa deweloperska)	No
cancelled-agreement	1 — includes tranches of cancelled contracts	No
include-cancelled	1 — includes tranches marked as deleted	No

Parameter	Value and meaning	Required
last_modified	date; △ the only parameter of this report written with an underscore , not a hyphen	No

Default population: the report returns tranches of contracts with status W toku (In progress) and Podpisana (Signed) (541 and 162 of 703 rows on the demo instance). Cancelled contracts are included only with cancelled-agreement/1. If you calculate arrears without this parameter, you omit cancelled contracts — this is usually the intent, but it must be stated in the metric definition.

△ Important: the remote-sync/1 parameter **does not change the response**. In the code it assigns the same column label that the report sets anyway — it does not remove any field. The Client / Company full data column with full contact data (first name, last name, phone, e-mail, contact_id, UUID) is **always** returned, also with remote-sync/1 (on the demo instance the field is populated in 670 of 703 rows both with and without this parameter; in 33 rows the column is empty together with "Klient / Firma" and "ID klienta"). There is no parameter that would disable it.

If you are building an export without personal data, remove this column on your side immediately after retrieval — do not rely on a filter on the API side.

Example URL:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/schedule-payment-list/date-from/2015-01-01/date-to/2026-08-31

Selected fields: the table is not the full response schema (a row has 28 columns). Formats confirmed on the demo instance on 10.09.2026.

Field	Format	Meaning
Termin	string "YYYY-MM-DD"	Planned payment due date of the tranche. This is the field that date-from and date-to apply to
Kwota	number	Tranche amount in PLN
Zapłacono	number	Amount settled in this tranche. This is not the inflow in the due-date month
Pozostało	number	Amount minus paid. A positive value with a past due date indicates arrears
Data płatności	string; empty text when there is no value	Date of the payment assigned to the tranche; empty in 615 of 703 rows on the demo instance. For monthly inflows use the payments report (the payment_date field)
ID umowy	number	Contract identifier (transaction.id); repeats across many rows

Field	Format	Meaning
Transaction UUID	string	Contract UUID; repeats across many rows
Schedule UUID	string	UUID of a single tranche — the natural row key. The label is untranslated
Status umowy	string	Translated contract status: by default only W toku and Podpisana
Client / Company full data	string containing JSON	Full contact data (first name, last name, phone, e-mail, contact_id, UUID). Untranslated label, field always returned

Illustrative JSON — fictitious values, selected fields:

```
[
  {
    "Termin": "2026-08-15",
    "Kwota": 100000,
    "Zapłacono": 75000,
    "Pozostało": 25000
  }
]
```

How to interpret the example: For this illustrative tranche the balance is $100,000 - 75,000 = 25,000$ PLN. As of 2026-09-10 the balance is overdue. This does not mean that 75,000 PLN was received in August — a report grouped by Termin does not give the date of receipt. The example omits the identifier: it is not meant for testing deduplication.

Limitations:

- The response **always** contains the Client / Company full data column with full contact data in nested JSON. It cannot be disabled with a parameter — remove it on your side before publishing anything.
- The default one-year due-date window omits older arrears. On the demo instance the default retrieval showed 438 overdue tranches totalling 30,561,646.16 PLN, while the full history showed 789 totalling 48,489,995.26 PLN — 351 tranches totalling 17,928,349.10 PLN were left outside the default window.
- This is a schedule, not a payment ledger. The Kwota, Zapłacono and Pozostało columns describe the settlement status of the tranche as of the retrieval date, broken down by tranche due date. For monthly inflows use the payments report and its payment_date field.
- By default the report omits cancelled contracts and tranches from lower-stage contracts (hide-lower). Describe every total with the due-date range and the hide-lower, only-not-paid and cancelled-agreement settings.

20. Payment list (payments)

A list of **actual payments** recorded on contracts, with the posting date — i.e. what was actually received, as opposed to the schedule-payment-list report, which shows the planned tranche schedule. Enables integration with ERP systems.

Response: an object with a data array in the form {"data": [...]}, not a flat array. Row fields: payment_id, invoice_id, document_number, amount, payment_date, investment_id, transaction_id. The investment name is not returned — only investment_id.

Available filters:

- date_from — payment start date (YYYY-MM-DD); ⚠ name with an **underscore**, not a hyphen
- date_to — payment end date (YYYY-MM-DD); ⚠ name with an **underscore**
- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection
- include_cancelled_transactions/1 — includes payments from cancelled contracts

Default range: none. Without dates the endpoint returns the entire payment history, without pagination — on a large instance always provide date_from and date_to. Payments from cancelled contracts are omitted by default; to reconcile refunds, add include_cancelled_transactions/1.

Quality control: the payment_date field may contain the value 0000-00-00 (a payment without a filled-in date; one such record on the demo instance). It is returned when retrieving without a date filter, but disappears with any date_from — as a result, the total for the whole set and the total of monthly retrievals may differ, without any message. For time series, discard such records and count them separately as a data quality error: do not convert them to zero and do not drop them silently. Negative amounts in the amount field are refunds and corrections (28 of 299 records on the demo instance) — count them separately from inflows.

Example:

https://NAZWA_PANELU.voxdeveloper.com/web/service/power-bi/source/payments/date_from/2026-01-01/date_to/2026-01-31

21. Invoice list (invoices)

A list of invoices issued in voxCRM. Enables integration with ERP systems.

Response: an object with a data array in the form {"data": [...]}, not a flat array. A row contains invoice_id, document_number, invoice_type, parent_invoice_id, correct_invoice_id, agreement_id, agreement_number, issue_date, sale_date, description, net_value, gross_value, vat_value and the nested arrays contractors (counterparty data) and items (invoice line items). The row key is invoice_id; document_number is not unique (412 distinct numbers across 440 invoices on the demo instance).

Available filters:

- date_from — invoice issue start date (YYYY-MM-DD); ⚠ name with an **underscore**, not a hyphen
- date_to — invoice issue end date (YYYY-MM-DD); ⚠ name with an **underscore**
- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection

Default range: none. Without dates the entire invoice history is returned, without pagination. The date filter applies to the issue date (issue_date).

Limitations:

- Corrections are separate records (invoice_type = Korekta, with parent_invoice_id populated)

and carry their own full amounts, equal to the amount of the original invoice. Summing gross_value over the whole set doubles the value of corrected invoices — on the demo instance this affects 34 of 440 records.

- The invoice_type value is translated into the instance language — on a Polish instance: Zaliczkowa, Ostateczna, Zwykła, Proforma, Korekta. Do not filter by English names.
- The issue_date and sale_date fields return empty text, not null, when there is no value. On the demo instance 2 records had an empty issue_date and 38 an empty sale_date.
- The response contains counterparties' personal data (first name, last name, e-mail, phone) — do not publish it in raw form.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/invoices/date_from/2026-01-01/date_to/2026-01-31

22. Sales forecast (sales-forecast)

A register of active contracts broken down by unit: unit data, prices, discounts, payment schedule and adjacent units (appurtenances). One row is one main unit with a current, non-cancelled contract — including a reservation contract and a contract in progress. **This is not a list of sold units.** On the demo instance, 468 rows correspond to 468 distinct main units.

Response: a flat array of rows, 54 keys in Polish. The fields "Każda płatność rozpisana" and "Anulowane umowy - szczegóły" are nested arrays of objects. The response can be very large — in the order of megabytes: about 1.5 MB for 468 rows on the demo instance — and has no pagination, so always narrow it with a date range or an investment.

Available filters:

- date_from/DATA_OD — date the current contract was concluded, from (YYYY-MM-DD); ⚠ name with an **underscore**, not a hyphen
- date_to/DATA_DO — date the current contract was concluded, to (YYYY-MM-DD); ⚠ name with an **underscore**
- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection

⚠ Important: the hyphenated variants date-from and date-to are ignored without any message — the request ends with HTTP status 200 and the full data set. The date used for filtering is not itself returned in the response as a separate column.

The four management price columns ("Cena zarządcza brutto", "Cena zarządcza netto", "Cena m2 zarządcza brutto", "Cena m2 zarządcza netto") are populated only for units that have a management price filled in on their record — on the demo instance all four are empty in all 468 rows. A missing value is not an error; check coverage before building a metric on them.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/sales-forecast/date_from/2024-01-01/date_to/2025-01-01

Units and properties

23. Unit list (realestate-list)

The endpoint returns a unit dictionary — a mapping "unit UUID → unit name", grouped by

Page 37 / 53

investment. Its purpose is identifying units, not reporting.

Response: a nested object keyed by investment names — {nazwa inwestycji: {UUID lokalu: {"name": "..."} } }. It is neither an array nor an object with a data field, so in Power BI it must be flattened into rows (investment, UUID, name). The number of first-level elements is the number of investments, not the number of units: 3 investments and 1596 units on the demo instance.

```
{
  "Osiedle Zielony Staw": {
    "0f5c1c2e-...": { "name": "1.1" },
    "3a77b9d4-...": { "name": "1.2" }
  },
  "Osiedle Przykładowe": {
    "b214e0af-...": { "name": "A/12" }
  }
}
```

Scope: all non-deleted units in non-deleted investments assigned to the API key — all kinds (apartments, parking spaces, garages, storage units, boxes, commercial units) and all statuses, including sold and withdrawn. The query has no sorting, so the order is not guaranteed.

Available filters:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection. The filter **narrows** the response: providing one name returns one investment, two names — two. Separate multiple names with a comma and URL-encode the values.

⚠ Important:

- The only unit field is name — there is no type, status, area, price or numeric identifier.
- The unit name is not unique within an investment; only the UUID is a key. On the demo instance, 36 (investment, name) pairs occur at least twice.
- Other reports (available-for-sale — field "ID lokalu", sales-forecast — "ID lokalu głównego") use a numeric identifier that is not present here. Do not join these reports by unit name.

Example:

https://NAZWA_PANELU.voxdeveloper.com/web-service/power-bi/source/realestate-list/investment/Osiedle%20Zielony%20Staw

24. Units available for sale (available-for-sale)

A list of units available for sale. The report takes into account **four** unit record statuses, not two: Dostępne (Available), Rezerwacja ustna (Verbal reservation), Wstrzymane (On hold), and a fourth status marked in the code as Reserved (its label depends on the instance translation). On the demo instance, the 733 returned records are 728 Dostępne, 4 Wstrzymane and 1 Rezerwacja ustna.

Response: a flat array of rows. Fields: ID lokalu, Nazwa lokalu, Inwestycja, Etap, Typ, Status, Powierzchnia, Liczba pokoi, Piętro, Cena brutto, Cena netto and four management price columns.

⚠ **Important:** the record unit is **any property**, not an apartment. The endpoint has no type filter, so filtering is done on the client side — by the Typ field. On the demo instance, apartments account for only **131 of 733 records (17.9%)**; the rest are garages (162 — 22.1%), storage units (134 —

18.3%), parking spaces in seven variants (293 in total — 40.0%), bicycle boxes (7), cellars (5) and 1 commercial unit. **The number of rows in this report is not the number of available apartments** — counting rows instead of filtering by type inflates this metric 5.6 times.

☐☐ **Tip** to get the number of **available apartments**, apply two conditions on the client side (in Power BI): Typ = Mieszkanie and Status = Dostępne. Units on hold and reserved units are in the response even though they are not available for sale — a type filter alone will not remove them.

Available filters:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection

The endpoint has no date filter, sorting or pagination — it always returns a full snapshot at the time of retrieval.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/available-for-sale/investment/Osiedle%20Zielony%20Staw

25. Sold and unsold report (sold-and-not-sold)

A summary of sold and unsold properties broken down by unit kind, with quantities, areas, values and prices per m2.

Response: a nested object — neither a flat array nor an object with a data field. The first level consists of three groups: Sprzedane (Sold), Niesprzedane (Unsold), Podsumowanie (Summary). The second level is the unit type name (Mieszkanie, Garaż, Komórka lokatorska...) plus the pseudo-type Suma (Total). The third level is a single-element array with the metrics. In Power BI, expand in turn: the group record, the type record, the list.

```
{
  "Sprzedane": {
    "Mieszkanie": [ { "Ilość": 457, "Łączna ilość (%)": "78,52", ... } ],
    "Garaż":      [ { ... } ],
    "Suma":      [ { "Łączna sprzedaż (wartość netto) PLN": "...", "Łączna sprzedaż (wartość brutto) PLN": "..." } ]
  },
  "Niesprzedane": { "Mieszkanie": [ { "Ilość": 125, ... } ], ... },
  "Podsumowanie": { "Mieszkanie": [ { "Ilość": 582, ... } ], ... }
}
```

⚠ **Important:** Podsumowanie is the sum of the Sprzedane and Niesprzedane groups, while Suma is the sum of type values within a group and contains only two amount columns (net and gross), with no Ilość field. Neither of these rows may be added to the others.

Group definitions: depend on the instance configuration. By default, Sprzedane covers every unit whose current non-cancelled contract is at least a reservation contract — and therefore **also contracts in progress and mere reservations**. Niesprzedane are units with record status Dostępne or Rezerwacja ustna, minus the sold units; instance variables determine, among other things, whether units on hold are counted as unsold. The same report on two instances may therefore have different definitions.

Why this report shows a different sales percentage than the investment status. This is the most common source of discrepancies in portfolio reporting. By default the report considers a unit

sold if it has **any** contract — including a reservation that never went through. Both reports count "sold", but according to different definitions, and on the same data they give results that differ almost sevenfold. Measurement on the demo instance:

Report	What it counts as sold	Result on the demo instance (apartments)	Share sold
sold-and-not-sold	a unit with any non-cancelled contract that is at least a reservation contract — including in progress	457 sold, 125 unsold, 582 in total	78.5%
investment-status	only a unit with a contract whose transaction status is Podpisana	68 sold, 39 reserved, 492 remaining, 599 in total	11.4%

How to reconcile the two figures. To get a result consistent with the portfolio status, add a contract status filter:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/sold-and-not-sold/agreement-status/Podpisana

On the demo instance the query then returns **100** sold apartments instead of 457 — which reconciles with the portfolio status from the investment-status report (68 sold and 39 reserved) and with the 105 unique units with signed contracts in the register.

⚠ **Important:** the values of the agreement-status filter are **translated into the instance language** — on a Polish installation you enter Podpisana or W toku, not the English names. The comparison is literal and the system normalizes only the first letter, so case matters for the remaining letters (Podpisana will work, PODPISANA will not). You cannot select the Anulowana (Cancelled) status with this filter — it is excluded from the list of allowed values. With every published sales percentage, state which definition you are showing: anyone who opens the "Sprzedane i niesprzedane" (Sold and unsold) report in the CRM will see the default value.

Which report to use:

- **"How many units are already contracted, what stage the project's sales are at"** — sold-and-not-sold. It includes contracts in progress, so it shows the real state of commercialisation.
- **"How many contracts are closed, how much can we recognise as sales"** — investment-status or sold-and-not-sold with the agreement-status/Podpisana filter.
- **"How many apartments can be bought today"** — neither of these two. Use available-for-sale with a filter on the Typ and Status fields.

Do not combine these figures in one chart and do not calculate shared metrics from them. The populations are not identical either: sold-and-not-sold reports by type name ("Mieszkanie"), while investment-status reports by the residential group, excluding commercial units, storage units and bicycle boxes.

Available filters:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection
- investment-stage/NAZWY_ETAPOW — stage (works when a single investment is selected)
- investment-buildings/NAZWY_BUDYNKOW — building (works when a single investment is selected)
- agreement-type/TYPY_UMOW_PO_PRZECINKU — contract type (Akt notarialny, Umowa deweloperska, Umowa przedwstępna, Umowa rezerwacyjna)
- agreement-status/STATUSY_PO_PRZECINKU — contract status (Podpisana, W toku)
- date-from/DATA_OD — contract signing date from (⚠ **hyphens**, not underscores)
- date-to/DATA_DO — contract signing date to (⚠ **hyphens**)

⚠ **Important:** the date-from, date-to, agreement-type and agreement-status filters apply **only to the Sprzedane group**. The Niesprzedane group is a current snapshot of units with status Dostępne or Rezerwacja ustna, minus the units found in the Sprzedane group — these filters do not narrow it, and filtering out some of the sold units may even slightly enlarge it. A unit sold outside the date range disappears from the report entirely, and Podsumowanie combines the filtered Sprzedane with the unfiltered Niesprzedane, so percentages and totals with a date range do not describe the whole investment.

On the demo instance, a retrieval with the range 2026-01-01 – 2026-08-31 gave 160 sold apartments instead of 457, with Niesprzedane at 127 instead of 125 — the share sold dropped from 78.5% to 55.8%. **For sales in a period use the agreements-registry report, and retrieve this report without dates.** The investment, investment-stage and investment-buildings filters apply to both groups.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/sold-and-not-sold/investment/Osiedle%20Zielony%20Staw

26. Sold adjacent units (sold-associated-units)

A list of adjacent units (storage units, boxes, parking spaces) linked to **non-cancelled** contracts — and therefore also to contracts in progress and reservation contracts, not only signed ones.

Response: a flat array of rows with six fields: ID lokalu, Powierzchnia, Cena zarządcza brutto, Cena zarządcza netto, Cena m2 zarządcza brutto, Cena m2 zarządcza netto. Only the net price and per-m2 conversions are calculated automatically — the gross management price is a unit record field and may be empty (on the demo instance all 380 records have these fields empty).

⚠ **Important:** a record does not contain the investment name, unit name, adjacent unit type or the linked contract — based on it you cannot tell a storage unit from a parking space. Add context using the numeric ID lokalu: via sales-forecast (the "ID miejsca postojowego" and "ID komórki lokatorskiej/boksu" columns) or via available-for-sale (the "ID lokalu", "Typ", "Inwestycja" columns). Do not join it with realestate-list — that report provides only the UUID and name, without a numeric identifier.

Available filters:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection
- date_from/DATA_OD — main unit contract date from (YYYY-MM-DD); ⚠ name with an **underscore**
- date_to/DATA_DO — main unit contract date to (YYYY-MM-DD); ⚠ name with an **underscore**

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/sold-associated-units/investment/Osiedle%20Zielony%20Staw

27. Current status and value of units (investment-status)

The report presents the current investment status broken down by sales category. One record describes an investment-status category pair, not a single unit. The values refer to the moment of retrieval, not to the sales history in a selected month.

Response: an object with the fields data (rows) and meta (metadata). A row has 14 fields: the investment name, the status category and metrics for five groups of unit kinds — count, area and value for the residential group and commercial units, and count and value for storage units, parking spaces and bicycle boxes. When the status filter does not match a category, the response is an empty array [], not a data/meta object.

Unit kind scope: the report breaks the status down into five disjoint groups, each with its own fields:

- realestate_* — residential group, **excluding** commercial units, storage units and bicycle boxes
- commercial_units_* — commercial units
- storage_units_* — storage units
- parking_spaces_* — parking spaces and garages
- bike_boxes_* — bicycle boxes

⚠ **Important:** realestate_count means **apartments only** (the residential group). Storage units, garages, parking spaces, bicycle boxes and commercial units have **separate columns** and **must not be summed** with realestate_count or with each other. On the demo instance, summing all the *_count columns gives 1523 "units" instead of 599 apartments, i.e. a metric inflated two and a half times. Moreover, the total of the five groups is not the number of all units in the investment: property kinds hidden in the instance configuration and kinds outside the residential and parking groups (e.g. cellars, shares) are not counted in any column.

URL path parameters — selected range:

Parameter	Value and meaning	Required
investment_name	comma-separated investment names; URL-encode the values. ⚠ name with an underscore	No
status	one or more comma-separated category names, with no space after the comma . The allowed values are exactly those the report returns in the sales_progress field — on a Polish instance Sprzedane, Zarezerwowane, Pozostale. The comparison is literal and only the first letter is normalized, so writing in capitals will not work. A name outside this list does not return an error, just an empty array []	No

Tip the filter that narrows this report to selected investments is **only investment_name** (with an underscore) — that is the name of the parameter read by this endpoint. The investment segment, used by the other reports, is checked only globally on this endpoint: a name not assigned to the API key or a non-existent name results in HTTP code 403, but a valid name **will not narrow** the result. Do not use both at once.

Example URL:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/investment-status/investment_name/Osiedle%20Przykladowe

Selected fields: the table is not the full response schema (a row has 14 fields).

Field	Format	Meaning
investment_name	string	Investment name; a label, not a persistent identifier
sales_progress	string	Status category, translated into the instance language. Sprzedane = unit with a signed development contract, preliminary contract or notarial deed. Zarezerwowane = unit with a signed reservation contract; a verbal reservation on the unit record does not belong to this category. Pozostale = no matching signed contract
realestate_count	number	Apartments only — the residential group, excluding commercial units, storage units and bicycle boxes. Do not sum with the other count columns
realestate_area	number	Area of residential group units in m2
realestate_value	number	PLN; for the Pozostale category the unit's list price, for units with a contract the value of the contract subject
commercial_units_count	number	Separate count of commercial units
storage_units_count	number	Separate count of storage units
parking_spaces_count	number	Separate count of parking spaces and garages
bike_boxes_count	number	Separate count of bicycle

Field	Format	Meaning
		boxes

Illustrative JSON — fictitious values, selected fields:

```
{
  "data": [
    {
      "investment_name": "Osiedle Przykładowe",
      "sales_progress": "Sprzedane",
      "realestate_count": 40,
      "realestate_value": 28000000
    },
    {
      "investment_name": "Osiedle Przykładowe",
      "sales_progress": "Zarezerwowane",
      "realestate_count": 10,
      "realestate_value": 7200000
    },
    {
      "investment_name": "Osiedle Przykładowe",
      "sales_progress": "Pozostałe",
      "realestate_count": 50,
      "realestate_value": 37500000
    }
  ],
  "meta": {
    "total_investments": 1,
    "generated_at": "2026-09-10 13:38:34",
    "status_note": "STAN AKTUALNY NA DATĘ WYGENEROWANIA RAPORTU"
  }
}
```

How to interpret the example: 40 + 10 + 50 = 100 **apartments** in the `realestate_count` field, of which 40% are sold. The total value of 72,700,000 PLN combines different valuation bases, so it should not be called revenue. The example omits the adjacent unit and commercial unit columns — they are not included in these 100.

Limitations:

- Only units with a contract whose transaction status is Podpisana are counted in the Sprzedane and Zarezerwowane categories. A unit with a development contract in W toku status falls into the Pozostałe category. On instances where most contracts have the W toku status, the Sprzedane count will be much lower than in the sold-and-not-sold report — this is not an error, it is a different definition. A detailed comparison of both reports is in section 25.
- The Pozostałe category is **not** the same as "available on offer". The report does not check the unit record status, so Pozostałe also includes withdrawn units, units on hold, units not released for sale and units with a contract in progress. On the demo instance, Pozostałe is 492 apartments, while available-for-sale returns 131 apartments — an almost fourfold difference. For the question "how many apartments can be bought today", use available-for-sale with a filter on the Typ and Status fields.
- Do not call `realestate_count` the number of all unit kinds — it is apartments only. Do not sum `realestate_value` as revenue either: for the Pozostałe category it is the list price, and for units with a contract the value of the contract subject, so the total combines different valuation bases.

- The values describe the status at the moment of retrieval (the meta.generated_at field), not the sales history in a selected month. The report has no date filter.
- No results after the status filter gives an empty array [] instead of a data/meta object — handle both shapes.

Activities and contact

28. Meeting report (meeting)

Information about meetings with clients. The date filter applies to the meeting date (Data spotkania), not the creation date. The default range — the current calendar year from 1 January to 31 December — applies only when neither date-from nor date-to is provided; in that case it also includes meetings scheduled in the future. Providing only one of the date parameters gives a range that is open on the other side. The endpoint does not limit the length of the range.

Available filters:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection
- date-from/DATA_OD — meeting date from
- date-to/DATA_DO — meeting date to

⚠ **Important:** the date-to parameter is compared with the meeting date at 00:00, so meetings on the date-to day itself — except those at midnight — are not included in the result. To cover the whole last day, set date-to one day later. When retrieving data in consecutive windows, remember that a meeting at 00:00 on the boundary day will then fall into both windows — filter out duplicates by meeting date.

Example (meetings for the whole of 2024):

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/meeting/date-from/2024-01-01/date-to/2025-01-01/investment/Osiedle%20Zielony%20Staw

29. Client activity list (activities-list)

A list of client-related activities (tasks, meetings, phone calls, etc.).

Available filters:

- activity_date_from/DATA_OD — start date (YYYY-MM-DD); defaults to today minus 12 months
- activity_date_to/DATA_DO — end date (YYYY-MM-DD); defaults to today
- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection
- activity_status/NAZWA_STATUSU — activity status; names exactly as in the status field of the response (e.g. Zaplanowane, Zakończony, Odbyło się, Wysłano); separate multiple values with commas
- activity_type/NAZWA_TYPU — activity type; allowed values: telefon, spotkanie, email, zadanie, zaplanowany email (case-insensitive); separate multiple values with commas
- user_id/ID_UZYTEKOWNIKA — restriction to the specified sales advisors (IDs separated by commas)

⚠ **Important:** the date range cannot exceed 12 months. A longer range, or an activity_date_from later than activity_date_to, results in an HTTP 400 response with the body {"error":"The period between activity_date_from and activity_date_to cannot exceed 12 months"}. Retrieve longer history

in consecutive 12-month windows.

⚠ **Important:** activity_type has its own dictionary of values, and these are NOT the labels from the type field of the response. In particular, activity_type/Zaplanowany%20E-mail — i.e. the label visible in the response — does not work; the correct value is zaplanowany%20email. A value outside the dictionary is silently ignored: the response has HTTP status 200 and the full set of records, so after setting the filter always check total_records.

💡 **Tip** activity_status works differently from activity_type — it accepts status names exactly as they appear in the status field of the response (e.g. activity_status/Zaplanowane, activity_status/Wysłano), regardless of letter case.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/activities-list/activity_date_from/2024-09-01/activity_date_to/2024-09-30

Response shape: an object {"success": true, "timestamp": "YYYY-MM-DD HH:MM:SS", "total_records": N, "data": [...]}. Row fields: id, activity_date, type, status, user_id, user, contact_id, contact, priority, notice, description, created_date, investment_name.

⚠ **Important:** the response is not paginated — with the default 12-month window it covers all activities in the instance (panel), and its size grows in direct proportion to their number (on the demo instance, 612 activities amount to approx. 190 kB, so on a large deployment expect a response many times larger). For recurring retrievals, narrow the window (e.g. to one month) and set the investment or user_id filters. Providing in the investment filter a name that the key cannot see results in an HTTP 403 response, not an empty result.

30. Client transaction history (contact-transactions)

A summary of a client's sales events broken down by investment. Record = client + investment + event type (the event_type field). The endpoint returns only three families of events: client creation and lead-to-client conversion, online offers (numbered sequentially: "Oferta 1", "Oferta 2", ...) and contracts (reservation, preliminary, developer, final) together with the number, area and value of units and adjacent units (appurtenances).

Response shape: an object {"data": [...], "meta": {"total_records": N, "generated_at": "YYYY-MM-DD HH:MM:SS"}}. Row fields: contact_id, investment_name, event_type, date, realestate_count, realestate_area, realestate_value and adjacent unit counters (commercial_units_*, sold_storage_units_*, sold_parking_spaces_*, sold_bike_boxes_*).

⚠ **Important:** this is not a history of contact with the client. The endpoint does not return phone calls, meetings, e-mails, tasks or notes — this data is retrieved from activities-list (item 29), meeting (item 28) or clients-report (item 4).

💡 **Tip** the date filter works independently in each event family and applies to the date of that event: for contracts — the contract date, for offers — the offer issue date, and for the "Client creation" ("Utworzenie klienta") and "Conversion to client" ("Konwersja na klienta") rows — the creation date of the client or lead. A client created before the start of the window will not have a "Utworzenie klienta" row in the response, although their contracts from that window will still appear. So do not count new clients from this report if the date window was set for contracts.

Available filters:

- date_from/DATA_OD — start date (YYYY-MM-DD); defaults to today minus 6 months

- date_to/DATA_DO — end date (YYYY-MM-DD); defaults to today
- investment_name/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection (separate multiple names with commas)
- contact_id/ID_KLIENTA — restriction to a single client

⚠ **Important:** here the date parameter names use an UNDERSCORE (date_from, date_to), unlike in most reports. The date range cannot exceed 6 months: a longer range, or a date_from later than date_to, results in an HTTP 400 response with the body {"error": "The period between date_from and date_to cannot exceed 6 months"}. You can provide only one of the parameters — the missing one takes its default value, but the 6-month limit still applies, so date_from/2024-01-01 on its own will also return HTTP 400. Retrieve history longer than 6 months in consecutive windows on the client side.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/contact-transactions/date_from/2024-01-01/date_to/2024-06-30

31. Incoming call identification (incoming-call)

An auxiliary endpoint for the telephony integration (Cludo). For a given phone number, it returns the linked contact, its account manager and the active users from that account manager's branches. It does not return a list or history of calls — you will find that data in mobile-call-list (item 32).

Available filters:

- self_target/NUMER_TELEFONU — the caller's number (required parameter; before searching, spaces, hyphens, dots, commas and underscores are removed, as well as the +48 or 48 prefix)

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/incoming-call/self_target/500600700

Response shape: {"contact_id": ..., "account_manager_id": ..., "account_manager_email": ..., "user_branch": [{"name": "...", "users": [{"id": ..., "full_name": "..."}]}]}

⚠ **Important:** a call without self_target always returns HTTP 200 and an object containing only null values, with user_branch equal to an empty array. This is neither an error nor a lack of data in the instance (panel). The response looks the same when the number could not be matched to any contact. The number stored in the contact record is not normalized, so matching is exact — a number kept in the CRM in a different format (e.g. with a country prefix) will not be found. When several contacts match the number, only one is returned, and when the contact has no assigned account manager, the user_branch field may be null.

The number is compared with the contact's work, home and mobile phone numbers, as well as its additional numbers. The user_branch list omits deleted and inactive users (deactivated in the CRM); the account_manager_email field comes directly from the contact record and may also point to an account that is already inactive.

32. Mobile app call list (mobile-call-list)

A list of phone calls from the mobile app. Each call includes: date, phone number, status (Outgoing / Incoming / Missed — Wychodzące / Przychodzące / Nieodebrane), whether it is linked to a client and

whether the client has a contract.

Available filters:

- date-from/DATA_OD — call date from (YYYY-MM-DD); defaults to today minus 3 months
- date-to/DATA_DO — call date to (YYYY-MM-DD); defaults to today
- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection
- contact-has-agreement/0|1 — 1 = only calls with clients who have a non-cancelled contract, 0 = only calls with other numbers

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/mobile-call-list/date-from/2026-01-01/date-to/2026-06-30

⚠ **Important:** here the date parameter names use hyphens (date-from, date-to), unlike in activities-list. The contact-has-agreement filter accepts only 0 or 1; any other value — including "Tak" and "Nie" (Yes/No), which appear in the response in the Klient ma umowę column — is converted to 0, i.e. it silently narrows the result to calls with clients WITHOUT a contract instead of disabling the filter. Without dates, the report covers only the last 3 months, not the entire call history.

💡 **Tip:** the investment filter does not exclude calls not linked to any investment — they are included in the result regardless of the selected investments. On the demo instance, 2 out of 586 calls were linked to a contract (the "Linked to client" ("Powiązane z klientem") column), so the vast majority of rows are numbers without a contact in the CRM. The "Client has a contract" ("Klient ma umowę") column has the value "Nie" in all 586 rows there.

33. Users of a specified branch (branch-users)

A list of users (sales advisors) assigned to a single, specified branch.

Available filters:

- name/NAZWA_ODDZIALU — the full branch name from the CRM (required parameter; the name must match in full, a partial name returns nothing)

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/branch-users/name/Oddzia%C5%82%20Warszawa

Response shape: {"users": [{"id": 123, "full_name": "Nowak Jan"}]} — the full_name field has the form "surname first name", and the response does not contain the branch name, so assign it on the integration side.

⚠ **Important:** a call without the name parameter, as well as a typo in the branch name, returns HTTP 200 and {"users": []}. An empty list does not mean that the branch has no employees. The API does not provide an endpoint listing branches — read the names from the CRM or from the user_branch[].name field in the incoming-call response.

The endpoint omits deleted and inactive users (deactivated in the CRM), as well as branches marked as deleted.

Tenant changes

34. Tenant change list (realestate-changes-list)

A list of tenant changes (buyer-requested modifications) relating to properties. Used, among other things, when copying contracts from other instances (panels).

Available filters:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection (names exactly as in the CRM)
- last_modified/DATA — returns changes created or modified on or after the given date (format YYYY-MM-DD or YYYY-MM-DD HH:MM:SS); used for incremental retrievals

Example:

`https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/realestate-changes-list/last_modified/2026-01-01`

Tip here, omitting the last_modified filter means the full change history, not an empty response. On the demo instance, a call without parameters returned 109 changes, the oldest from 2017.

Technical and data warehouse

35. List of available endpoints (endpoints)

A meta-information endpoint returning a list of all Power BI sources granted to a given API key. It allows you to check programmatically which data sources are available.

Response: an array of objects with the endpoint and description fields.

```
[
  { "endpoint": "/webservice/power-bi/source/lead-comparision", "description": "Dane z raportu leadów - porównanie" },
  { "endpoint": "/webservice/power-bi/source/investment-status", "description": "Current sales status of the investment" }
]
```

The endpoint field contains the FULL path, not just the source name. Take the name you insert after /source/ from the last segment of this path — do not append the whole value to the base address.

⚠ Important: the endpoints source itself must also be granted to the key. A key without this permission will get HTTP 403 with an empty body, even though it is valid and can see other reports — discovery will simply fail.

The list is always the complete set of the key's permissions; there are no filters here. Adding a /investment/NAZWA segment does not narrow the result, and an unknown investment name results in HTTP 403. If the response contains a source that is not in the catalogue below, it is available but undocumented; agree its scope with the support team before using it.

Tip Use this endpoint to check which data sources are active for your API key. It is a good starting point for the initial configuration, as well as a simple way to detect changes: save the list of names after stripping the /webservice/power-bi/source/ prefix and compare it on the next retrieval.

36. Contract register — anonymized (transaction-list-anonymized)

Page 49 / 53

© 2026 voxCRM <kontakt@voxdeveloper.com> | 14-10-2026 16:45

URL: <https://voxcrmdu.com/content/0/246/en/power-bi-endpoints---list-of-available-data-sources.html>

A trimmed-down version of the contract list export (transaction-list, item 12) — the same rows, but a narrower set of columns: without client data (client and company name, their identifiers, e-mail, phone) and without the columns "Contract ID" ("ID umowy"), "Unit number" ("Numer lokalu"), "Main unit ID" ("ID lokalu głównego"), "Building" ("Budynek"), "Adjacent unit" ("Przyległość"), "Contract cancellation date" ("Data anulowania umowy"), "Contract discount amount" ("Kwota zniżki z umowy"), "Discount percentage" ("Procentowa wartość zniżki") and "Contract classification" ("Klasyfikacja umowy"). Created for data warehouse purposes.

⚠ Important: this response is not fully anonymized. The "Sales advisor" ("Doradca") column contains the employee's first and last name, and "Investment" ("Inwestycja") — the real investment name. Before sharing materials externally, remove or aggregate these columns.

The response contains neither the contract UUID nor the unit ID — the only identifier is the text "Contract number" ("Numer umowy"), which is not guaranteed to be unique. To join with other reports, use agreements-registry (item 11) or transaction-list (item 12).

37. Contracts by acquisition source (warehouse-contracts-by-source)

An anonymized summary of leads, clients, activities and contracts broken down by investment and acquisition source. Created for data warehouse purposes. One row = investment × acquisition source × month (the period field contains the first day of the month).

Required parameters (the report will not work without them):

- date-from/DATA_OD — date from (YYYY-MM-DD)
- date-to/DATA_DO — date to (YYYY-MM-DD)

Both parameters are mandatory and written with a HYPHEN. This is one of the few reports without a default range: a call without dates returns HTTP 400 with the body {"error":"Missing parameters date-from or date-to"}, and a value the system cannot interpret as a date returns HTTP 400 and {"error":"Invalid date format"}. There is no upper limit on the length of the range.

Optional parameter:

- investment/NAZWY_INWESTYCJI_PO_PRZECINKU — investment selection; provide the REAL names from the CRM, not the I1/I2 aliases visible in the response

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/warehouse-contracts-by-source/date-from/2026-08-01/date-to/2026-08-31

What anonymization means here: only the NAMES are replaced — developer_name and investment_name receive the aliases D1, D2... and I1, I2.... The investment_id, developer_id, city and voivodeship fields contain real values.

⚠ Important: aliases are assigned anew with every call — numbered in ascending order of the identifiers of the entities included in that response. The same investment may therefore be "I3" in one retrieval and "I1" in the next (e.g. after adding the investment filter or changing the date range). Names from this report cannot be joined by name with other reports — do not use the alias as a join key or as a label in a dashboard. The persistent key is investment_id / developer_id.

Fields: developer_id, developer_name, investment_id, investment_name, city, voivodeship, source_of_origin_id, source_of_origin, period, leads_count, unique_clients_count, activities_count, reservation_agreements_count, sale_agreements_count, lead_created_date, contract_created_date.

⚠ **Important:** the date filter works independently for each column and on a different date field:

- leads_count — lead creation date,
- unique_clients_count — client creation date,
- activities_count — the ACTIVITY DATE (the time of a completed call, a closed meeting, a sent e-mail), not its creation date; only completed activities are counted,
- reservation_agreements_count and sale_agreements_count — the contract creation date in the CRM, not its signing date.

A row is therefore not a funnel: a contract in a given row need not have anything to do with the lead in the same row.

Note on the contract counters:

- reservation_agreements_count counts CONTRACTS (transactions) at the "Reservation agreement" ("Umowa rezerwacyjna") stage,
- sale_agreements_count counts unique UNITS at the developer contract, preliminary contract or notarial deed stage.

These are different units of measure — do not add the two columns together as "contracts". The current stage of the item is used, not its history: a unit reserved in August and sold in September will no longer appear as an August reservation. Items without a set source of origin are omitted from all columns, which is why totals from this report may be lower than in leads-list or agreements-and-reservations.

39. Contracts by acquisition source — non-anonymized variant (warehouse-contracts-by-source-stg)

The same report as item 39: the same set of 16 fields and the same filters (date-from and date-to mandatory, investment optional, the same HTTP 400 error handling). The difference is that developer_name and investment_name contain real names instead of the D1/I1 aliases. All other values, including the counters, are identical — on the demo instance, both variants returned the same 34 rows, differing only in the two name fields. This variant is intended for an internal data warehouse.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/warehouse-contracts-by-source-stg/date-from/2026-08-01/date-to/2026-08-31

⚠ **Important:** this variant returns the REAL names of developers and investments, so it is not suitable for materials shared externally or for aggregate summaries covering multiple developers. For such uses, use the anonymized version from item 39.

Access to both variants is granted by the support team; you can check the current list of your key's permissions with the endpoints endpoint (item 37).

39. Daily system usage statistics (daily-statistics-usage)

One row for each day in the range, with counters of work in the system: user logins, created clients and new leads, created contracts, overdue calls and meetings, added defects, available units and the counts at the other stages, sent online offers and automations. The data covers the entire instance (panel) — there is no breakdown by investment or sales advisor.

Required parameters (the report will not work without them):

- date-from/DATA_OD — date from (YYYY-MM-DD)
- date-to/DATA_DO — date to (YYYY-MM-DD)

The names are written with a HYPHEN. Like items 39 and 40, the report has no default range: a call without dates returns HTTP 400 with the body {"error":"Missing parameters date-from or date-to"}, and a value that cannot be interpreted as a date returns HTTP 400 and {"error":"Invalid date format"}. There is no limit on the length of the range.

Example:

https://NAZWA_PANELU.voxdeveloper.com/webservice/power-bi/source/daily-statistics-usage/date-from/2026-08-01/date-to/2026-08-31

Response: an array of rows with Polish column names, sorted in ascending order by date: Data, Liczba logowań użytkowników, Utworzeni klienci, Nowe leady, Leady nieprzekonwertowane, Leady nierozpoznane, Utworzone umowy, Przeterminowane telefony, Przeterminowane spotkania, Dodane usterki, Dostępne lokale, Lokale zarezerwowane, Lokale umowa deweloperska, Lokale odebrane, Lokale umowa końcowa, Wysłane oferty online, Aktywne automatyzacje, Uruchomienia automatyzacji.

Limitations:

- The row for a given day is created only after the nightly job that runs at 23:55. Today's date will not yet be in the response.
- A single row contains three kinds of counters: daily flows (these can be summed across days), end-of-day states (Dostępne lokale, Lokale zarezerwowane and the other stages, as well as Aktywne automatyzacje) and cumulative backlogs (Przeterminowane telefony, Przeterminowane spotkania). Do not sum states and backlogs across days.
- Columns added in later versions of the system show zero for the period before their rollout. In that case, zero means no measurement, not no events.
- The /investment/NAZWA segment does not narrow the result (the data covers the whole instance), but an unknown name still results in HTTP 403.

Guidelines for preparing analyses and working with an AI agent

- **Portfolio status:** count the unit categories in investment-status. "Remaining" ("Pozostałe") does not mean only units available for sale.
- **Contract dynamics:** in agreements-registry, determine the types and statuses, check the UUIDs and group by contract date. Compare full months; mark the current month as incomplete.
- **Client sources:** count unique non-empty contact_id values in the retrieved report. Show a missing source, the value "Other" ("Inne") and conflicting sources for a single contact separately.
- **Arrears:** sum the positive balances of tranches with a due date earlier than the reference date. Under this rule, a tranche due today is not yet overdue. **Omitting the start date does not give you the complete arrears** — when date-from is missing, the server itself goes back exactly one year from the retrieval date, and older tranches are not included in the response at all. To calculate the full arrears, provide date-from explicitly, with a date earlier than the start of the investment's history, and always record the range of retrieved due dates in the metric definition. On the demo instance, the default retrieval showed 438 overdue tranches totalling PLN 30,561,646.16, while the full history showed 789 tranches totalling PLN 48,489,995.26 — 351 tranches totalling PLN 17,928,349.10, i.e. 37% of all arrears, fell outside the default window.
- **Check whether the filter worked:** an unrecognized parameter name is ignored without an error — the response has status HTTP 200 and looks correct. Run the same query twice: with

the filter and without it. If the number of records is identical, the parameter name or value was not recognized and you are working on the full set. On the demo instance, lead-comparison without parameters and the same source with an added /nieistniejacy-parametr/abc segment each returned 24 rows with identical content. The exception is the investment filter: an incorrect name does not return the full set, but HTTP 403 with an empty body.

- **Sales percentage:** always state the definition alongside the figure. sold-and-not-sold and investment-status answer different questions and, on the same demo instance data, give **78.5%** (a unit with any contract, including a reservation that did not go through) versus **11.4%** (a unit with a sale confirmed by its status in the unit record). Both results are correct — they differ in population, not by error. A metric without a stated source and rule for classifying a unit as sold is not comparable.
- **Funnel and conversion:** do not divide totals from independent reports without a common population, period and verified relationships. Do not add an earlier stage just because a later contract exists.
- **Public materials:** use selected aggregates. Raw responses may contain personal data. Renaming investments does not remove the business information contained in amounts and counts.

Instructions for the agent preparing the integration

Read the definition of each report you use. Record its scope, record unit and key. Set your own User-Agent header in the request with the name of your integration — the default Python library header is sometimes rejected by the firewall with 403 before the key is even checked, in which case the diagnosis "the key does not work" is false. Do not convert errors or missing data into zero. Do not join reports based on similar names or ID values alone. Separate illustrative data from data retrieved from the API. Describe each metric with the source name, formula, period, unit and exclusions. Before presenting the result, compare control totals with the CRM for the same filters and permissions.

Updating the integration

After an API change, compare the available endpoints, field names, response shape and value types. Also check the meaning of dates, statuses and populations — they can affect the result even when the column names stay the same. Report any unknown change before recalculating the report further.

In case of problems, provide the support team with the endpoint name, the filters applied, the retrieval date and the error message. Do not include the API key or personal data in a public ticket.

Unique solution ID: #1281

Author: n/a

Last update: 2026-10-14 12:50