Customer success platforms: Gainsight and ChurnZero

From the SourceLace manual: How to connect Gainsight CS and ChurnZero, so agents and people can read customer health, success plans, tasks and surveys next to your CRM, service desk and Gong.

SourceLace version 0.1.0 · Made November 22, 2026
The latest version is at https://sourcelace.ai/docs

Customer success platforms: Gainsight and ChurnZero

How to connect Gainsight CS and ChurnZero, so agents and people can read customer health, success plans, tasks and surveys next to your CRM, service desk and Gong.

Status: Pilot. New, offered to pilot customers. Built on each vendor's published API and covered by automated tests against simulated versions of them. SourceLace has not yet tried either one against a real Gainsight or ChurnZero account, so the first connection is checked together with you. Where the vendor's own documentation does not say something, this page says so and tells you who can confirm it.

At a glance

Source	Kind	Reads	Changes	People sign in with	First test question
Gainsight CS	`gainsight`	Companies and any other Gainsight object (scorecards, surveys, custom objects), CTAs (calls to action) and success plans	None (read-only)	One **shared** access key an admin makes	"Which Gainsight objects does gainsight:cs offer?"
ChurnZero	`churnzero`	Accounts, contacts, churn scores, surveys and NPS, journeys, tasks, notes, meetings, segments and events	None (read-only)	One **shared** API user and key an admin makes	"Which ChurnZero entities does churnzero:cs offer?"

Neither is needed by the ready-made customer success agents: they keep working with only your CRM, service desk and Gong, and use Gainsight or ChurnZero too when you add one.

Both are shared access. Neither vendor signs each person in separately for these APIs, so everyone who may use the source acts as the one key SourceLace was given. The vendor cannot tell people apart, and neither can limit a key to what each person may see. Narrow who can use the source with access by group, and hide or mask personal data with data protection. The key is kept encrypted and never shown again.

SourceLace only reads. Gainsight's access key can also change data in Gainsight, because Gainsight has no read-only key. SourceLace's connector sends only the read calls it lists (queries, lists and descriptions) and refuses anything else before it is sent. For ChurnZero, a user whose permission group has only read permissions makes the key read-only in ChurnZero itself, which is the better protection: step 2 below.

As everywhere in SourceLace, queries are checked before anything is sent, results are held for at most 30 minutes and never written to a database (see How long things are kept), and the vendor's address is called only if it is a public address on the internet. A SourceLace Runner can reach one that is private (Enterprise plan only, on request; see Runners).

Gainsight CS

What your admin does in Gainsight

Once. About 10 minutes.

Who: a Gainsight **Super Admin**. Only an active Super Admin can make the access key.

1. **Open the connection page.** In Gainsight CS, go to **Administration → Connectors 2.0**. If you cannot find it, search the Administration menu for "Connectors 2.0": Gainsight renames menus from time to time.
2. **Make the access key.** Choose **Create Connection**, then the connector type **Gainsight API**, set **Authentication type** to **Access_Key**, and choose **Generate Access Key**. Copy the key. **An organization has only one access key.** If one already exists, other tools may use it: ask who first, because generating a new one stops the old one working everywhere.
3. **Find your Gainsight address.** It is the address you sign in on, such as `https://yourcompany.gainsightcloud.com`, or your own domain. Gainsight's API documentation also names separate API addresses (for companies and for custom objects) that an admin sets up first. If SourceLace cannot reach Gainsight on your sign-in address, ask Gainsight support which address your account's API uses: Gainsight's documentation does not say how to find it.
4. **Allow SourceLace's addresses, if you limit them.** If your Gainsight limits which addresses may call it (**Administration → Security**, Trusted IPs; this moved from Network Access in the July 2026 release), ask Gainsight support whether that applies to the access key, and if it does, add SourceLace's addresses (see What your network needs).
5. **Give your SourceLace admin** the address and the key. They add the source below.

Check that it worked: ask "Which Gainsight objects does gainsight:cs offer?". You should see Company and your other objects, plus `cta` and `success_plan`. If you see "Gainsight did not accept the access key", look it up under When something goes wrong.

Gainsight's own guide: Gainsight support (search for "Generate REST API Access Key").

Add the source (SourceLace admin)

Kind: **Gainsight** (`gainsight`).

Option	Type	Default	Example	What it is
`base_url`	Web address	(required)	`https://yourcompany.gainsightcloud.com`	Your Gainsight address (step 3). Must start with `https://`, with no path and no `?query`.
`secret`	Secret	(required)		The access key (step 2). Stored encrypted and never shown again. Leave it empty when editing to keep the saved one.

What people can do

`search_schema` lists Gainsight's own objects, so custom, scorecard and survey objects appear under their real names, and `describe_object` lists an object's fields. `query` (language `gainsight_json`) takes one small JSON object:

What	Example	Notes
An object	`{"object": "Company", "select": ["Name", "Csm__gr.Name"], "limit": 100}`	`select` names the fields to return. A lookup field is followed by a dot and a field of the object it points to. At most 5,000 records a call; SourceLace reads further pages up to the limit you set.
A filter	`{"object": "Company", "select": ["Name"], "where": {"conditions": [{"name": "Status", "alias": "A", "operator": "EQ", "value": ["Active"]}], "expression": "A"}}`	Operators: EQ, NE, LT, GT, LTE, GTE, BTW (two values), IS_NULL, IS_NOT_NULL, CONTAINS, DOES_NOT_CONTAINS, STARTS_WITH, ENDS_WITH, IN, NOT_IN, INCLUDES, EXCLUDES. Give each condition a one-letter alias and join them in `expression` with AND, OR and brackets.
CTAs and success plans	`{"resource": "cta", "select": ["Name", "Status"], "limit": 100}`	`select` and `limit` only: filter the rows afterwards.

`get_record` reads one record of an object by its Gsid. Gainsight does not document a read for health scores directly: read them from the scorecard objects your Gainsight has (find them with `search_schema`). Gainsight allows about 100 calls a minute, and SourceLace stays under it for your whole organization together; when Gainsight asks SourceLace to slow down it waits and tries twice more.

Gainsight and ChurnZero in one question across systems

Gainsight objects (such as `Company`) and ChurnZero entities (such as `Account`) can be a table in one question across your systems, so an AI app can put customer health next to your CRM accounts or your service desk in a single answer, for example "which of our largest Salesforce accounts have a ChurnZero score above 70?". Name the table as `"churnzero:cs"."Account"` or `"gainsight:cs"."Company"` and ask for the fields you need. Four things to know:

- **CTAs and success plans** are not available this way, because Gainsight does not list their fields: read them with a query on their own. Asking for one in a question across systems answers that the table "has no fields SourceLace can read".
- **Dates arrive as text.** Compare them as text (`"Created" >= '2025-01-01'`), not with `DATE '2025-01-01'`.
- **Gainsight reads at most 100 fields of an object at a time.** An object with more needs the fields named instead of `*`, and the answer says so when it is asked for too many.
- **A very large object or entity is read only up to a limit.** When SourceLace stops at that limit, the answer says it is incomplete and says why, instead of looking whole. ChurnZero allows about 100 requests a minute, so a large entity takes a while: add a filter on names, numbers, yes/no fields or ids: SourceLace asks the system for those, while date and "is empty" filters are applied by SourceLace after reading, so they do not make a large read smaller.

When something goes wrong with Gainsight

What you see	What it means	What to do
"Gainsight did not accept the access key. A Gainsight Super Admin checks it ..." (HTTP 401 or 403, Gainsight code `GS_APIG_2401`)	The key is wrong, was re-generated, or the address is wrong. A key made in a Gainsight sandbox does not work on production.	A Super Admin checks the key (step 2) and the address (step 3), then you enter the current key on the source.

What you see	What it means	What to do
"gainsight:cs"."cta" has no fields SourceLace can read. Check it with describe_object.	CTAs and success plans have no field list, so they cannot be a table of a question across systems (describe_object shows none either).	Read them with a query on their own (the table above).
"Gainsight reads at most 100 fields of an object at a time..."	The question needs more than 100 fields of one object (usually a *).	Name the fields you need.
"Gainsight does not know that address or object. Check the base_url and the object name." (HTTP 404)	The address is not an address that answers Gainsight API calls, or the object does not exist.	Check the address (step 3) and the object name with `search_schema`.
"Gainsight refused the query. Gainsight said: ..." (HTTP 400, for example `COCKPIT_5101` for a CTA query)	A field or condition Gainsight does not accept.	Check the field names with `describe_object`. The AI usually fixes the query itself.
"Gainsight: ... (GSOBJ_...)"	Gainsight's own words for an object query it refused, such as an unknown field.	As above.
"Gainsight is limiting how fast SourceLace may call it. Wait a minute and try again." (HTTP 429, code `GU_2400`)	You are at Gainsight's limit of about 100 calls a minute, shared with your other tools.	Wait and ask again, or ask for less at once.
"Gainsight answered, but SourceLace could not read the fields of ... from it."	Gainsight's description of the object has a shape SourceLace did not expect.	Query the object with the fields you know, and tell support@sourcelace.com.
"Gainsight answered with something other than data (HTTP ...). Check the base_url."	The address leads to a web page, not to Gainsight's API.	Check the address (step 3).
"... needs the option base_url ..." or "... needs the option secret ..."	An option is missing.	Fill it in (see the table above).
"The base_url of ... must start with https://." or "... must not have a ?query in it."	The address is in the wrong form.	Use the plain address from step 3.
"'...' is not a Gainsight object name." or a refusal naming a key such as "Extra inputs are not permitted"	The query used something SourceLace does not accept.	Use the example shapes above.
"Gainsight found no ... record with the Gsid ...."	No record has that Gsid, or the access key cannot see it.	Use a Gsid from a query's results.
"SourceLace refused to make that call to Gainsight."	The call was to an address outside the base URL you gave, which SourceLace never makes.	An admin checks the source's base URL.

ChurnZero

What your admin does in ChurnZero

Once, plus a wait if the API is not switched on yet. About 15 minutes.

Who: a ChurnZero **admin**, and your ChurnZero **customer success manager (CSM)** for step 1.

1. **Have the REST API switched on.** ChurnZero's REST API is an add-on that ChurnZero enables for your account. Ask your CSM to switch it on, and ask which address your API uses (step 4). ChurnZero's older "app key" is for sending events to ChurnZero and cannot read data: do not use it here.

2. **Make a user for SourceLace, with read permissions only.** In ChurnZero, make a new user (for example `sourcelace-reader`) in its own **User Permission Group**, with only **Developer Tools** and the **Bulk Read** permission of each thing SourceLace should read (for example **Bulk Read - Accounts** and **Bulk Read - Contacts**). The API key can do only what its user's group allows, so this makes it read-only. ChurnZero's menu names can differ: your CSM can show you where permission groups are.
3. **Make the API key.** In ChurnZero, as that user, go to **Admin → API Keys** and create a key. Copy the user name and the key.
4. **Find your API address.** ChurnZero's REST API reference (app.churnzero.net/developers) shows the form `https://<your-name>.us1app.churnzero.net/public/v1`. The part in the middle depends on your account's region, and ChurnZero's documentation does not say how to find it: your CSM can confirm it.
5. **Give your SourceLace admin** the address, the user name and the key.

Check that it worked: ask "Which ChurnZero entities does churnzero:cs offer?", then "Show five ChurnZero accounts". A "ChurnZero refused this" answer means a Bulk Read permission is missing: see When something goes wrong.

ChurnZero's own guide: ChurnZero support (search for "API").

Add the source (SourceLace admin)

Kind: **ChurnZero** (`churnzero`).

Option	Type	Default	Example	What it is
`base_url`	Web address	(required)	`https://yourcompany.us1app.churnzero.net/public/v1`	Your ChurnZero REST API address (step 4). Must start with `https://`, with no `?query`.
`username`	Text	(required)	`sourcelace-reader`	The user name of the ChurnZero user from step 2, without a colon.
`secret`	Secret	(required)		The user's API key (step 3). Stored encrypted and never shown again. Leave it empty when editing to keep the saved one.

What people can do

`search_schema` lists the ChurnZero entities SourceLace reads, and `describe_object` shows an entity's fields (from one real row, because each account's fields differ). `query` (language `odata`) takes an entity and its options:

What	Example	Notes
A filtered list	`Account?$filter=NpsScore lt 7&$select=Name,NpsScore&$top=50`	Options: `$filter`, `$select`, `$expand`, `$orderby`, `$top`, `$skip`, `$count`. Pages hold 100 rows and SourceLace reads further pages up to the limit.
One record	`Account(12345)`	By its id.
The activity stream	`Event?$filter=...`	`Event` is very large: it needs a `$filter` or a `$top`.

Entities: Account, Contact, UserAccount, ChurnScore (0 is low risk, 100 is high risk) and its calculations and factors, Event and EventType, Journey, JourneyInstance and JourneyMilestone, Meeting, Message, Note, Segment, Survey and SurveyResponse (NPS), and Task with its priorities and statuses. Aggregation (`$apply`) is not offered. ChurnZero allows about 100 requests a minute for each instance (more are delayed, and over 200 are refused), and SourceLace stays under it for your whole organization together.

When something goes wrong with ChurnZero

What you see	What it means	What to do
"ChurnZero did not accept the user name and API key. ..." (HTTP 401)	The user name or key is wrong, or the key was not made for that user.	Check both on the source and in ChurnZero under Admin → API Keys (steps 2 and 3). Also check the address (step 4).
"ChurnZero refused this: the API user may not read it. ..." (HTTP 403)	The user's permission group lacks Developer Tools or the Bulk Read permission of that entity, or the REST API add-on is not on.	Add the permission in ChurnZero (step 2), or ask your CSM to switch the API on (step 1).
"ChurnZero does not know that address or record." (HTTP 404)	The address is wrong, or the id does not exist.	Check the address (step 4) and the id.
"ChurnZero refused the query. Check the $filter and $select names with search_schema." (HTTP 400)	A field name or filter ChurnZero does not accept.	Check the names with `describe_object`. The AI usually fixes the query itself.
"ChurnZero is limiting how fast SourceLace may call it. Wait a minute and try again." (HTTP 429)	You are at ChurnZero's limit for your instance.	Wait and ask again, or ask for less at once.
"ChurnZero returned HTTP 500."	A problem on ChurnZero's side.	Try again in a minute. If it repeats, tell your CSM.
"ChurnZero sent SourceLace to another address for the next page, so it stopped."	A next-page link pointed away from your ChurnZero address. SourceLace never follows it.	Tell support@sourcelace.com.
"... needs the option username ..." or "... needs the option secret ..." or "... needs the option base_url ..."	An option is missing.	Fill it in (see the table above).
"A ChurnZero query starts with an entity name ..." or "'...' is not allowed. Use only $filter, ..."	The query used an entity or option SourceLace does not accept.	Use the examples above.
"Event is very large. Add a $filter ... or a $top."	`Event` was asked for without a bound.	Add a date range as a `$filter`, or a `$top`.
"ChurnZero found no ... with the id ...."	No record has that id.	Use an id from a query's results.