

Business apps: Dynamics 365, NetSuite and Gong

From the SourceLace manual: How to connect Microsoft Dynamics 365 (Dataverse), Oracle NetSuite and Gong.

SourceLace version 0.1.0 · Made October 4, 2026

The latest version is at <https://sourcelace.onrender.com/docs>

Business apps: Dynamics 365, NetSuite and Gong

How to connect Microsoft Dynamics 365 (Dataverse), Oracle NetSuite and Gong.

Status: in testing. All three are built against each vendor's documented API and covered by automated tests against simulated services; none has been used with a customer's live account yet.

Source	Kind	Reads	Changes	People sign in with
Microsoft Dynamics 365	dynamics365	OData queries, tables and columns, records with related rows	Create and update one record at a time, after the person confirms (off unless an admin turns it on). No deletes.	Their own work or school account
Oracle NetSuite	netsuite	SuiteQL queries, record types, records with sublists	Create and update one record at a time, after the person confirms (off unless an admin turns it on). No deletes.	Their own NetSuite login and role
Gong	gong	Calls, call details, transcripts, users, activity and interaction stats	None (read-only)	A Gong technical admin's login, company-wide (see Gong)

As everywhere in SourceLace, queries are checked before anything is sent, results are held in memory for 30 minutes and never written to a database, and sign-in tokens are encrypted and never shown. Gong transcripts are read only when someone asks for them.

The redirect URL for all three is:

<https://sourcelace.onrender.com/connect/callback>

Microsoft Dynamics 365

Each person signs in with their own Microsoft work or school account. Dataverse then applies that person's security roles, business units and field security to everything SourceLace reads or changes.

Who sets it up: SourceLace uses its own Microsoft app (the same one as "Sign in with Microsoft"), so there is nothing for you to register. Sign-in asks for `<org_url>/user_impersonation` (Dynamics CRM, delegated: "act as the signed-in person") and `offline_access` (stay connected).

What your Microsoft admin may need to do: if your organization requires an admin to approve apps, Microsoft shows **Need admin approval**. A Microsoft admin then grants consent for SourceLace in the **Microsoft Entra admin center → Enterprise applications**.

What each person needs: a **Dynamics 365 license and a security role** in that environment, as they would to use Dynamics 365 in the browser.

Add the source (SourceLace admin)

Kind: **Microsoft Dynamics 365** (dynamics365).

Option	Type	Default	Example	What it is
org_url	Web address	(required)	https://corvanta.crm.dynamics.com	The environment's address: Power Platform admin center → Environments → (your environment) → Environment URL , or the address bar when Dynamics 365 is open. Regional environments use crm4, crm11 and so on. Only https://<name>.crm<n>.dynamics.com addresses are accepted.

To allow changes, turn on **Allow changes** and list the tables by their entity set names as search_schema shows them, such as accounts, opportunities.

What people can do

- search_schema lists the tables they can see; describe_object accounts shows columns, choice values, lookups and related lists.
- query (language odata) takes an entity set name plus OData options \$select, \$filter, \$top (1 to 5,000), \$expand and \$orderby, such as accounts?\$select=name,revenue&\$filter=revenue gt 100000&\$orderby=revenue desc&\$top=20. Other options, functions, actions and batch calls are refused before anything is sent.
- Choices and lookups also come back with their display text, in a column named <column>_label.
- get_record accounts <id> reads one record (the id is a GUID), and can add related lists such as contact_customer_accounts.
- **Changes:** propose_change shows the current and new values. The update is sent with the record's version (ETag), so Dynamics refuses it if someone changed the record after the preview. Lookups are set with "<navigation property>@odata.bind": "/contacts(<id>)". Deleting is not offered.

Oracle NetSuite

Each person signs in with their own NetSuite login and picks a role. NetSuite then applies that role's permissions and restrictions to every query and every change. Changes are off until a SourceLace admin turns them on (see [Add the source](#)).

Who sets it up: SourceLace's NetSuite integration is installed into your account by your **NetSuite administrator**, once.

Set up (NetSuite administrator)

1. **Turn on the features:** Setup → Company → Enable Features → SuiteCloud. Under SuiteTalk (Web Services) tick **REST Web Services**; under Manage Authentication tick **OAuth 2.0**. Save.
2. **Install SourceLace's integration:** Setup → Integration → Manage Integrations → Install, using the **Application ID** that support@sourcelace.com gives you. Without this step, sign-in fails with "invalid client".

3. **Give people a role that can use it:** each person's role needs the permissions **REST Web Services** and **Log in using OAuth 2.0 Access Tokens** (both under **Setup**), plus view access to the records they should see (and create or edit access to the records they may change, if changes are on). NetSuite may refuse OAuth 2.0 sign-in with the Administrator role, so people should use their everyday role.

A sandbox (such as 1234567_SB1) is the safest place for a first try. NetSuite sign-ins end after the period set on the integration; people then connect again.

Add the source (SourceLace admin)

Kind: **Oracle NetSuite** (netsuite).

Option	Type	Default	Example	What it is
account_id	Text	(required)	1234567 or 1234567_SB1	Setup → Company → Company Information → Account ID. Sandboxes and release previews end in _SB1, _RP and so on.

Sign-in asks for the scope rest_webservices.

To allow changes, turn on **Allow changes** and list the record types as search_schema shows them, such as customer, contact. Nothing new is needed in NetSuite: the sign-in already includes REST web services, so nobody needs to connect again.

What people can do

- search_schema lists record types; describe_object salesorder shows the fields and sublists.
- query (language sql) takes one SuiteQL SELECT (or WITH ... SELECT), such as SELECT id, companyname, email FROM customer WHERE isinactive = 'F'. All transactions are in the transaction table, with a type column such as 'SalesOrd'. Anything that could change data, SELECT ... INTO and more than one statement are refused before NetSuite is called. Statements can be up to 20,000 characters; no LIMIT is needed, since SourceLace pages through results itself.
- get_record salesorder 42 reads one record by its internal id (digits only); add item (or another sublist) to get its lines, up to 50 per sublist.
- **Changes** (when an admin has turned them on for that record type): propose_change shows a preview (for an update, each field's current value next to the new one), and apply_change makes the change after the person confirms. A create returns the new record's internal id.
 - Only a record's own fields can be set: text, numbers, true/false, or another record's internal id for a reference field (such as subsidiary: 1). Sublists (such as a sales order's lines) and subrecords (such as addresses) are not changed. Unknown fields and fields NetSuite sets itself are refused in the preview, before anything is written.
 - The person's NetSuite role decides what is allowed. If the role cannot change that record, NetSuite's own message comes back.
 - If someone changed the record between the preview and the confirmation, the change is not made and the person is asked to preview it again.
- **Deletes are refused.** NetSuite records are usually made inactive rather than deleted: update the record's isInactive field to true where the record type has one.

Gong

Gong is **read-only**: calls, call details (participants, Gong's brief, key points, topics, trackers and linked CRM records), transcripts, users, and activity and interaction stats.

Important: Gong's access is company-wide, not per person. Gong lets only its **technical administrators** authorize an app, and the sign-in gives access to the whole company's Gong data; Gong does not apply each person's own Gong permissions to API calls. So:

- Add a Gong source only where everyone who can use it may see all calls and transcripts. Narrow who can use it with [access by group](#).
- The person who connects must be a Gong technical admin.
- Gong does not always say which user signed in, so the audit trail may name the Gong company (gong:<api address>) instead of a person.

Who sets it up: SourceLace has its own Gong app. A Gong technical admin connects and approves it. Sign-in asks for `api:calls:read:basic`, `api:calls:read:extensive`, `api:calls:read:transcript`, `api:users:read`, `api:stats:interaction` and `api:stats:user-actions`.

Each Gong company has its own API address (such as `https://us-12345.api.gong.io`). Gong sends it with the sign-in, and SourceLace only ever calls that address (it must end in `.gong.io`).

Add the source (SourceLace admin)

Kind: **Gong** (gong).

Option	Type	Default	Example	What it is
api_base	Web address	(none: taken from the sign-in)	<code>https://us-12345.api.gong.io</code>	Optional. Pins the source to one Gong company: a sign-in to any other company is refused.

What people can do

query (language gong_json) takes one small JSON object:

Object	Example	Notes
calls	<code>{"object": "calls", "from": "2026-09-01", "to": "2026-10-01", "details": true}</code>	from and to default to the last 30 days (at most a year). Filter with <code>call_ids</code> , <code>user_ids</code> (the call's host) or <code>workspace_id</code> . <code>details</code> adds participants, brief, key points, topics and trackers.
transcripts	<code>{"object": "transcripts", "call_ids": ["7782342274025937895"]}</code>	Up to 20 calls. One row per stretch of speech, with the speaker's name.
users	<code>{"object": "users"}</code>	Everyone in the Gong company.
activity_stats	<code>{"object": "activity_stats", "from": "2026-09-01"}</code>	Calls hosted, attended, feedback and scorecards, per user.
interaction_stats	<code>{"object": "interaction_stats", "user_ids": ["2345994848423"]}</code>	Talk ratio, longest monologue, interactivity, patience, question rate, per user.

Ids are given as text. `get_record` calls `<id>` reads one call with its details and linked CRM records, and adds the transcript when transcript is asked for as a related list. Gong allows about 3 calls a second; when it asks SourceLace to slow down, SourceLace waits and tries again twice.

When something goes wrong

What you see	What to do
"... has no org_url. An admin sets it to the environment's address, such as <code>https://corvanta.crm.dynamics.com</code> ."	Fill in <code>org_url</code> .
"The org_url for ... must be a Dynamics 365 address such as <code>https://corvanta.crm.dynamics.com</code> ."	Copy the Environment URL exactly.
"Microsoft sign-in failed: ..."	Microsoft's own reason follows. If it mentions admin approval or consent, ask your Microsoft admin. If it mentions the scope, check <code>org_url</code> matches the Environment URL.
"Dynamics 365 has no table '...' that you can see. Use <code>search_schema</code> to find one."	Use the entity set name (accounts, not account), or the person's security role does not include that table.
"...' is not allowed. Use only <code>\$select</code> , <code>\$filter</code> , <code>\$top</code> , <code>\$expand</code> , <code>\$orderby</code> ."	Remove the other OData options.
"This record changed in Dynamics 365 after the preview. Propose the change again."	Someone else changed the record; propose the change again.
"Dynamics 365 accepts create and update through SourceLace, not delete."	Delete in Dynamics 365 itself.
"Dynamics 365: not found, or you cannot see it."	Check the id; the person's security role may not allow it.
"... has no account_id. An admin sets it to the NetSuite account id ..." or "The NetSuite account_id for ... must look like 1234567 or 1234567_SB1."	Fix <code>account_id</code> .
"NetSuite sign-in failed: ..."	NetSuite's own reason follows. "invalid client" means the integration is not installed in your account (step 2). Otherwise check the person's role has the two permissions in step 3, and is not Administrator.
"NetSuite did not say which user signed in."	Contact support.
"SELECT ... INTO is not allowed in a read-only query."	Only plain reads are allowed.
"NetSuite: not found, or you cannot see it."	Check the record type and id; the person's role may not allow it.
"NetSuite: your NetSuite role does not allow this. ..."	The person's role lacks create or edit access to that record type. Ask your NetSuite administrator.
"... has no field '...'. Use <code>describe_object</code> to see them." or "... is set by NetSuite and cannot be changed."	Use the field names <code>describe_object</code> lists, and leave out fields NetSuite fills in itself.
"... is a sublist; SourceLace changes a record's own fields only." (or subrecord)	Change lines and addresses in NetSuite itself.
"This ... record changed in NetSuite after the preview. Propose the change again."	Someone else changed the record; ask for a new preview.
"SourceLace does not delete NetSuite records. ..."	Set <code>isInactive</code> to <code>true</code> instead.
"Your NetSuite sign-in has expired." (or Dynamics 365, or Gong)	Connect again.
"You signed in to a different Gong company than ... uses. Connect again with the right Gong account."	Sign in to the Gong company named in <code>api_base</code> .

What you see	What to do
"The Gong api_base for ... must be Gong's API address, such as https://us-12345.api.gong.io."	Fix api_base, or clear it.
"Gong refused this: the sign-in lacks the permission it needs."	The person who connected is not a Gong technical admin, or did not approve every permission. Connect again.
"...' is not a Gong object. Use calls, transcripts, users, activity_stats, interaction_stats."	Use one of those objects.
"The dynamics365 connector is not available yet." (or netsuite, gong)	SourceLace's app for that system is not set up on this server yet. Contact support.