CUSTOMER GUIDE / PRODUCT RELEASE 2.0

How to Install

A Basis runbook for installing release 2.0, activating the selected interface and verifying an allowed and a denied request.

Prepare the delivery and landscape

1. Obtain the CGR-provided installation or upgrade package, its release-specific delivery notes and a license text file for the target licensing system identifier.
2. Confirm the supported SAP release, SPAM/SAINT level, application client, installation permissions, HTTPS address and planned change window.
3. Choose the SICF or Gateway OData interface and the execution backend. Agree a named integration user, narrowly scoped object grants and the backend execution identity.
4. For an upgrade, follow the supplied upgrade sequence and preserve existing client configuration and retained logs. Verify the planned maintenance and recovery procedure with Basis.

Supported systems start with SAP NetWeaver 7.4 on premises, including on-premises SAP S/4HANA, and SAP private cloud. Public cloud is not supported. Package prerequisites in the delivery notes remain part of installation readiness.

Use the package supplied for your release

A source ZIP is not a SAP-installable add-on. SAINT instructions apply to a genuine SAP add-on installation or upgrade package. If the delivery contains transport files instead, use the Basis import procedure specified in its delivery notes. Do not reuse transport numbers from older documentation.

Install the add-on

1. For a SAINT package, sign in to client 000 with the Basis installation identity and open SAINT.
2. Load the supplied archive from the front end or application server using the supported SAINT package-loading procedure for your release.
3. Select the correct CGR installation or upgrade package, review the calculated queue and prerequisites, then complete the import.
4. Resolve installation errors before continuing. Verify software component CGR release 200, or the precise release stated in your delivery notes.
5. Continue configuration in the application client that will receive the API calls. Installation in client 000 does not configure every application client.

Activate the SICF interface

1. In SICF, locate default_host > cgr > api. Activate the service and any required parent nodes.
2. Confirm handler class /CGR/CL_API_HANDLER on the api node. The /func, /table and /report paths are routed below it; they are not separate SICF services.
3. Configure HTTPS, certificates and HTTP Basic authentication. Verify that calls run as the intended integration user.

4. For a same-system setup, retain local routing. For a separate backend, install the required Connector objects there and configure the RFC destination and approved execution identity.

```
GET https://sap.example.com/cgr/api/table
    ?id=MARA&select=MATNR&top=1&sap-client=100
```

The example URL is displayed on two lines for readability. Send one URL, using your actual HTTPS host, port and application client.

Configure a separate SICF backend

Maintain the approved destination using /CGR/RFC_DEST and key BACKEND where the landscape uses a default backend. If requests select a sap-system alias, validate the alias and its destination before exposing it. Test connectivity and the actual backend execution user. The current execution path rejects an RFC mapping to a different user rather than silently attributing the call to the original HTTP user.

An externally reachable gateway can terminate inbound HTTPS while execution occurs in an approved internal SAP backend over RFC. The Connector itself does not initiate outbound HTTP/HTTPS requests.

Register the Gateway OData interface

1. Ensure the supplied Connector objects are installed in the intended Gateway/backend landscape using the release delivery instructions.
2. In /IWFND/MAINT_SERVICE, add technical service /CGR/CONN_SRV for the intended system alias. Assign the production package and transport required by your landscape.
3. Activate and verify the corresponding ICF service beneath /sap/opu/odata. Check the service root and Gateway metadata with an authorized identity.
4. Configure the OAuth flow supported by your SAP system, the client and generated service scope. Assign the appropriate SAP roles and Connector object grants.
5. Verify the selected Gateway backend. The Connector executes locally in the backend selected by Gateway; it does not add its own RFC rerouting step.

```
https://sap.example.com/sap/opu/odata/CGR/CONN_SRV/
```

Complete application-client setup

1. Run /CGR/LICENSE with product SAP_CONNECTOR. Select the original CGR-issued .txt license file and execute; verify the installation result and validity date.
2. In /CGR/BO, create or verify FUNCTION, TABLE and REPORT entries. TABLE uses POST Input Structure /CGR/ST_TABLE_INP; REPORT uses /CGR/ST_REPORT_INP. Leave the other generic structure fields blank as shown in the Quick Setup Guide.
3. In /CGR/API_AUTH, allow only the exact integration user and object names required. Assign the separate SAP roles required by the interface and business operation.
4. Grant the monitoring identity /CGR/MONI activity 03. Agree retention before storing production request or response data.

Verify and hand over

- Run a small, read-only table request with positive top. HTTP 200 with [] can be a valid result for an empty selection.
- Copy X-CGR-Message-ID and find the matching request in /CGR/MONI. Confirm user, backend, client, object and output.
- Using a fresh authenticated session, verify that a known existing object with no exact or wildcard grant is denied.
- Record installed release, endpoint, client, integration user, approved grants, license expiry and a successful message ID.
- Enable optional classification, notifications or scheduled cleanup only after their own configuration and preview. Importing the package creates no operational job.

CGR SOFTWARE
cgrsoft.com · sales@cgrsoft.com
Use these release 2.0 instructions with the supplied release delivery notes. Release 1.3 has a separate documentation and PDF set.