

Best Practices for Bulk Queries

Last Modified on 11/18/2021 8:43 am EST

Why Best Practices Are Important

Bulk queries are a powerful tool to retrieve records in greater numbers than possible when using standard SAP endpoints. Because of bulk's potential to return large amounts of records from providers, bulk jobs themselves can often take a long time to complete, making it even more important to utilize best practices when performing queries and fetching their results. Here are some of our recommendations:

Use the Where Clause

Make sure you are crafting your bulk queries to return only the records you are interested in seeing. For instance, instead of using the query `select * from contacts`, use `select * from contacts where lastModifiedDate > '2018-05-01'` to return only those records that were modified after May 1st, 2018. For more information on how to use the where clause, see [Querying with OCNQL](#).

Retrieve Only the Columns You Need

Many connectors support field selection. In your query, you can list only the columns that you want returned to improve the speed of the bulk job. For example, instead of `select * from contacts`, use `select firstName, lastName from contacts where ...` to only return the `firstName` and `lastName` fields of your contacts. This also minimizes the size of your bulk results file which decreases the time needed to fetch your results. Another way to achieve this is to use a common resource with only the fields mapped that you care about.

Note: Field selection is not supported for all connectors. Check the Bulk article for the connector for field selection support.

Use a Limit Statement

You can easily limit the number of results returned by your query by adding a limit statement. For example, `select * from contacts limit 500`. This retrieves only the first 500 records returned by the API provider, greatly reducing the time needed for the bulk job to complete if the unlimited query would return many more than 500 records.

Set a Callback URL

When creating a bulk query, you can provide a value for the header `Elements-Async-Callback-Url` to receive a notification upon the bulk job's completion. The notification also sends information about the bulk job, such as whether the bulk job was successful and the number of records returned. Incorporating this callback URL functionality into your app allows your app to process bulk job results as soon as they are done as well as handle any errors.

Note: In bulkv3, data with nested fields is recommended to be downloaded in `.JSON` and not in `.CSV` as the nested fields may not appear in the CSV format.