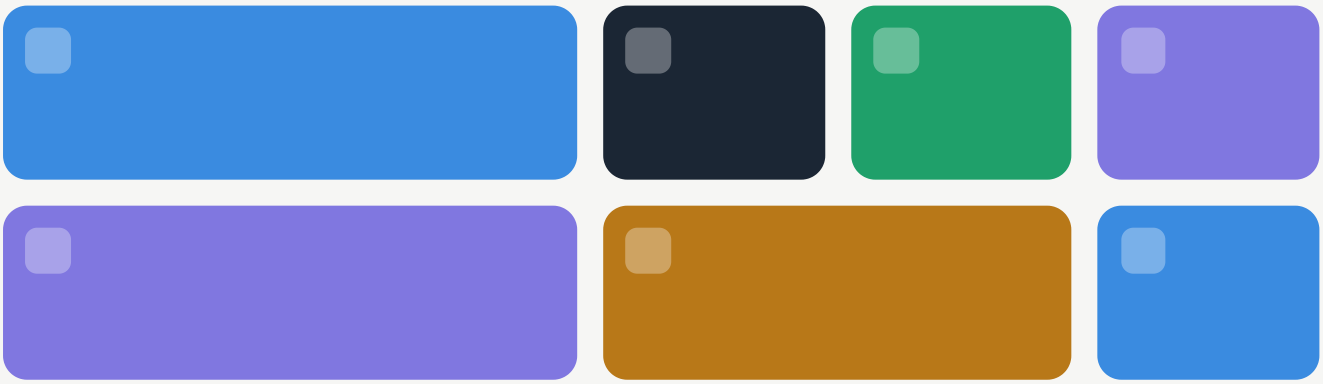
BPMSquare

CUSTOMER EDITION

Data Workbench

How to connect other systems to BPMSquare: the REST API, the change feed and webhooks, connectors, the integration middleware and Data Workbench.

New way of customer relationship



Data Workbench

Provided by **Snasquare**

Powered by **BPMSquare** · Snapshot 2026-10-29

CHAPTER 6

Data Workbench

> Screen names and button words may be refined as the product improves. If a name here is slightly different on your screen, look for the same idea nearby.

About this chapter

This chapter is for the person who loads, extracts or corrects records in bulk, and for the consultant who plans a migration. Data Workbench works from spreadsheets. It has three modes:

- **Import** creates new records.
- **Export** downloads records.
- **Update** changes records that already exist.

Data Workbench is a screen, not an API. It signs in as a person, not with an API key. If a system must create or change records on its own, use the REST API (Chapter 2). Only quotations can be written there. For other objects, a person runs Data Workbench.

The examples use a made-up company, Example Trading LLC.

Contents

1. Where it is and who can use it
2. Objects available
3. Import
4. Export
5. Update
6. Custom fields
7. Limits and rules
8. A checklist for a migration

---

1. Where it is and who can use it

Open **Admin > Data Workbench**. The mode is a tab at the top (**Import**, **Export**, **Update**). It stays when you reload the page.

- It needs the **Data Workbench** module, and a business role that includes the Data Workbench entry.
- Each object also needs its own module. If your workspace does not own Cases, Cases do not appear in the list, and the server refuses it too.

- Every row created or changed is recorded in **Change history**, with the name of the person who ran it.

2. Objects available

The list on the left shows only the objects your workspace owns. In **Update**, it shows only those that can be updated.

Object	Import	Export	Update	Notes
Accounts	Yes	Yes	Yes	Import first. Other objects link to it by name.
Contacts	Yes	Yes	Yes	Needs an existing account, by name.
Assets	Yes	Yes	Yes	A blank account makes loaner stock.
Suppliers	Yes	Yes	Yes	
Products	Yes	Yes	Yes	Includes quantity breaks.
BOM components	Yes	Yes	Yes	Adds components to a BOM or kit product that already exists.
Inventory	Yes	Yes	Yes	
Employees	Yes	Yes	Yes	Includes custom fields.
Users	Yes	Yes	No	Sends an email invitation to each person. Never updated in bulk.
Consents	Yes	Yes	No	An append-only ledger. A change is a new answer. Needs Marketing.
Quotes	Yes	Yes	Yes	One row for each line, with the header values repeated. Import creates the quote with its lines in one file.
Quote lines	Yes	Yes	Yes	Adds lines to an existing quote.
Standard quotes	Yes	Yes	Yes	Header only. Add lines with Standard quote lines.
Standard quote lines	Yes	Yes	Yes	Only for existing draft documents, found by their Quote ID.
Sales orders	Yes	Yes	Yes	Header only. Add lines with Sales order lines.
Sales order lines	Yes	Yes	Yes	Only for existing draft orders, found by their Order ID.
Cases, Work orders, Invoices, Purchase orders	Yes	Yes	Yes	
Deals (pipeline)	Yes	Yes	Yes	Amount and probability are computed, not imported.

Object	Import	Export	Update	Notes
RFQs	No	Yes	Yes	An RFQ is raised from a record, with its lines and suppliers. Export, then update header fields.
Estimates	Yes	Yes	Yes	Header only. Import makes a draft from a template name.
ABM programs, ABM target accounts, ABM buying groups	Yes	Yes	Yes	Needs ABM.
Projects (workforce)	Yes	Yes	Yes	Import parents before children.

Contacts, tenders and contract projects are not available through the API (Chapter 1). Contacts are available here.

3. Import

Import creates new records. It never needs an id, because a new record has none yet.

Steps

1. Choose **Import**, then pick the object on the left, for example **Accounts**. The panel says "Import … first" for anything it depends on. For example, contacts need accounts.
2. Under **Or start from the template**, press **Download Accounts template**. The template carries every column and example rows. It always matches your current field setup, including your custom fields.
3. Replace the examples with your data. Save as `.csv`, `.xlsx` or `.xlsm`.
4. Under **Upload your file**, drop the file or click to choose it. It can be up to 15 MB. Your own column names are fine.
5. On **Match columns**, point each column of your file at a BPMSquare field. Columns whose names match a field's label, or a common alias such as "qty" or "description", are matched for you. Check the rest.
6. On **Review**, read "N ready" and "N with problems". Rows with problems are left out. Press **Show only problems** to see them, and **Download problem list** to fix them in your spreadsheet.
7. Press **Import N rows**.
8. The **Result** step shows how many rows were imported, skipped and failed, and the reason for each. Press **Download report** to keep it.

The four steps are named **Upload**, **Match columns**, **Review** and **Result**.

Required columns

A row is rejected if a required column is empty. The required columns are:

Object	Required
Accounts	Name, Type

Object	Required
Contacts	Name, Account (an account that exists, by name)
Assets	Name, Kind
Suppliers	Name
Products	Name
Inventory	Name
Cases	Account, Type, Equipment label, Complaint
Work orders	Account, Authorizing quote (a Quote ID)
Invoices	Account
Purchase orders	Supplier
Quotes	Account. The quote name is optional. If you leave it blank, one is made from the account name.
Deals	Title, Account
Estimates	Name, Template (the name of an active estimate template)
ABM programs	Name
Projects (workforce)	Name
Users	Name, Email, Role

The template marks required columns. Your workspace may also make other fields required in its field setup.

Linking to other records

You link a new record to an existing one by a business key you type: an account name, a Quote ID such as `QT-2026-0042`, a program ID such as `ABM-0001`, a template name. The names are matched case-insensitively. The match is made within your workspace only.

- A name that is not found fails that row, with the reason ("Account … was not found, import it first").
- A name that is new in the same file does not count. Import the parent first, in an earlier file.
- Quotes: put the quote header on the first row of each quote and one line on every row. Repeat the same value in the optional **Quote name** column on each row of a quote if you want to name it yourself. To add lines to a quote that exists, use **Quote lines**.
- Quotes can also be created from a PDF or Word document. The product reads it, you review the rows, then import.

What happens with duplicates

Import does not replace anything.

- **Accounts** skip a row whose name already exists, and a name that appears twice in the file. The result says so.
- **Users** skip an email that already has an account.
- **Every other object creates a new record for every row.** Products, contacts, suppliers, assets and inventory do not check for an existing SKU or name. If you load the same file twice, you get two copies.

To correct records that are already in, use **Export** and **Update**.

---

4. Export

Export downloads records as a CSV file.

1. Choose **Export**, then an object.
2. Under **Choose columns**, tick the columns you want. **Record ID** is always included.
3. Under **Filters**, press **+ Add filter** to export only matching rows. Pick a field, a condition and a value. Rows must match every filter. Leave it empty to export everything.
4. Press **Download … CSV**.

Conditions depend on the field: *contains*, *is*, *is not*, *is empty*, *is not empty* for text; =, >, < for numbers; *on*, *before*, *after* for dates.

The first column of the file is **Record ID**. That is the BPMSquare id (Chapter 1, section 6). Keep it. It is what Update and your own systems match on.

Cells that start with `=`, `+`, `-` or `@` are protected with a leading mark, so a spreadsheet does not run them as formulas.

An export contains real data, including personal fields such as phone and email. The cost price of products is in the Products export, but never in the API.

---

5. Update

Update changes records that already exist. It always needs the **Record ID**.

The rule

A file for Update must come from an earlier **Export**. It must have a **Record ID** column, filled by the export. A file with only names or references is refused. Do not type ids by hand.

The reason is safety. A name can be shared by two records, or be renamed. An id cannot. A row whose id is not in your workspace fails by itself ("No record with id … found in this workspace"). It is never turned into a new record, and it can never touch another workspace.

Steps

1. **Export** the records you want to change. Keep the **Record ID** column.
2. Edit the values in the file. Do not edit the **Record ID**.

3. Choose **Update**, pick the object, and upload the file under **Or use a file you exported**.
4. On **Match columns**, check the mapping. **Record ID** is matched for you.
5. On **Review**, read "N ready" and "N with problems". Rows with problems are left out.
6. Press **Update N rows**. The **Result** step reports updated, skipped and failed rows with reasons.

What Update changes

- **Only the columns you map.** A column you leave unmapped is not touched.
- **A blank cell in a mapped column clears that value.** If you do not want a field cleared, delete its column from the file, or leave it unmapped.
- **Plain fields only.** Fields that link to other records (for example the account of a contact) are not updated in bulk. Change those on the record.
- **Computed columns are not editable.** Totals, statuses set by a workflow and the reference (such as the Quote ID) are export-only.
- **Quotes: header fields only,** one row for each quote. Use **Quote lines** to correct lines.
- Each row that changes something writes a **Change history** entry with the old and new values. Personal fields are recorded as redacted.

Rows are applied one at a time, a few in parallel. Updating thousands of rows takes longer than importing the same number.

---

6. Custom fields

Custom fields you make in **Settings > Custom fields** appear as columns in the template, in the Match columns step and in exports, with the prefix `cf_` on their key. You do not do anything special. If you hide a field in your field setup, it leaves the template.

Custom fields exist for accounts, contacts, assets, suppliers, products, inventory, cases, quotations, deals, employees and more. The template for each object lists the ones it has.

Download a fresh template whenever you add or change a custom field. An old template does not have the new column.

---

7. Limits and rules

- **Formats.** `.csv`, `.xlsx` and `.xlsm`. The old `.xls` is not supported. Open it and save as `.xlsx` or `.csv`.
- **Size.** 15 MB for each file. Split larger files.
- **Sheets.** Only the first sheet of a workbook is read. Empty rows are ignored.
- **Rows.** There is no row limit in the product. The whole file is processed in one request, so a very large file can run for a long time. For more than a few thousand rows, split the file into parts of about 1,000 rows and load them one after another. Check the **Result** of each before the next.

- **Order.** Load parents before children: accounts, then contacts and assets, then quotes, then the things that point at quotes. The panel tells you what to import first.
- **No hidden overwrite.** Import adds. Update changes only what you mapped. Neither deletes anything.
- **Users.** Importing users sends each person an email invitation. If **Email output** is set to a test inbox, the invitations go there. Check that setting first (Chapter 5, section 13).
- **No API.** Data Workbench cannot be called with an API key.
- **Cost price.** It is in the Products file. Treat the file as confidential.

---

8. A checklist for a migration

1. **Plan the order.** Suppliers, accounts, contacts, assets, products, inventory, then quotes, deals, cases, work orders, invoices and purchase orders.
2. **Set Email output to your own inbox** before any user import (Chapter 5).
3. **Download the template** for each object from your own workspace, not from an older project.
4. **Fill a small file first.** Ten rows. Import it. Open the records. Fix your mapping.
5. **Clean the data in the spreadsheet,** not afterwards. Names must match exactly between files. A contact's account must be spelled as the account is.
6. **Load in the right order.** One object at a time. Read **Review** before you press **Import**.
7. **Keep every Result report.** They are your audit of what loaded.
8. **Check the counts.** After each load, compare the number of rows you meant to load with the list. With the API you can count without reading rows: `GET /accounts?count=only` returns `meta.total` (Chapter 2).
9. **Do not load a file twice.** Except for accounts and users, a second load creates copies. To correct, Export and Update.
10. **Export a backup** of the loaded records, with **Record ID**. Your own systems can store those ids (Chapter 1).
11. **Switch Email output back** to **Send to business partners** when you go live.