

Rendering from the frontend

Every report configuration in VeloxFactory has a built-in **Generate PDF** page: a small workstation for one report template that renders it directly from the browser, without writing a single line of code or touching the API. It is the fastest way to produce a PDF, test a configuration, or trigger a print job on demand.

The page follows one principle: **settings once, then only the data**. How the output should be produced, preview or print, which printer, how many copies, whether a history record is kept, is decided once per report and then stays out of the way. From there, every render is nothing but filling in the values that actually change and pressing one button.

Screenshot placeholder: `generate-pdf.page.png`

Generate PDF page with the header row (report name, settings summary, Output Settings button), parameter table and the action row.

Opening the Generate PDF Page

There are two ways to reach the page. In the report configuration overview, each row has an action menu that contains **Generate PDF**, taking you straight to the render page. Alternatively, open a report configuration and use **Generate PDF** from within the report view.

The header row shows the report name and, as a compact summary line, the active data adapter and the stored output settings: mode, printer, copies, broadcast ID and whether a history record is created. The data adapter is either the assigned `ReportConnectionConfig` (including driver and database) or **dyn. Array** when no SQL connection is configured, and it decides which input sections appear further down.

Screenshot placeholder: `report-config.index.action-menu.png`

Report configuration overview, opened action menu with Generate PDF and Download.

Output Settings

The first time you open the page for a report, the **Output Settings** dialog opens by itself, because VeloxFactory does not yet know what you want to do with the result. Afterwards it only opens when you click the **Output Settings** button in the header.

Setting	Default	Description
Mode	Preview and download only	Decides what happens with the rendered PDF, see the table below.
Printer	-	Printer picker over the active printer master data, free text is allowed as well. Required in both printing modes.
Copies	1	Copies passed to the print service, between 1 and 100. Does not trigger multiple renders.
Broadcast ID	-	WebSocket channel of the print service. Filled in automatically when the chosen printer has one stored.
Create History Record	On	Saves request, response, PDF and thumbnail of every render to the report history.
Send results via Mail	Off	Mails every render of this report, see Mailing the result below.

The three modes:

Mode	Preview	Print task
Preview and download only	yes	no
Print only, no preview	no	yes
Preview and print	yes	yes

In *Preview and download only* the printer and broadcast fields are hidden, there is nothing to print to. In the two printing modes a printer is mandatory: saving without one fails and the dialog reopens with everything you entered still in place.

Settings are kept **per report configuration in your session**. Two report templates therefore keep independent settings, they survive leaving the page and coming back, and they end when your session does. **Clear settings** at the bottom of the dialog removes them, the dialog then opens by itself on the next visit.

i The settings are yours, not the report's. They live in your session, so a colleague configuring the same report for a different printer never affects your workstation.

Screenshot placeholder: 

Output Settings dialog with mode dropdown, printer picker, copies, broadcast ID, history toggle, the mailing block and the Clear settings button.

Mailing the Result

Send results via Mail turns every render of this report into a mail as well. While the switch is on, the dialog shows the mailing fields:

Field	Description
Mailer / Mail Template	The SMTP account and the template the mail is built from. Only active records are offered.
To, CC, BCC	Comma separated address lists. Placeholders are allowed, an address containing one is checked after the render, not in the dialog.
Contact Name	Free text for a personal greeting, available in the template as <code>[contactName]</code> .
Attach PDF Report	On by default, with an optional name the attachment carries in the mail.
Attach Excel Report	Attaches an xlsx built from the rendered rows, with an optional attachment name and a switch for the parameter block above the table.
Send in background	Hands the mail to the queue instead of waiting for the SMTP server.

The values stay in the form even while the switch is off, so turning mailing off for one render loses nothing. After the render the notification says what happened to the mail: sent, queued, or a warning with the reason. A mail that fails never turns a successful render into an error.

i The mailing block only appears when you can use it. It needs the permission to create mail tasks plus at least one active mailer and one active mail template. The same block sits in the Scan2Print configuration and on a render job in the Job Scheduler. Details on [Report Mailing](#).

Report Parameters

If the report defines parameters, a parameter input table is shown. Each parameter gets its own typed input field, derived automatically from the Java class declared in the `.jrxml`:

- A `java.lang.String` parameter becomes a text field.
- A `java.sql.Date` parameter becomes a date picker.
- A `java.lang.Integer` parameter becomes a number input with integer constraints.
- A `java.lang.Boolean` parameter becomes a toggle switch.
- And so on for all supported types.

Parameters marked as **required** in the report configuration must be filled in before the form can be submitted. Optional parameters can be left empty, VeloxFactory silently drops empty parameter values and does not include them in the render request.

Report Lines - Manual Data Entry

The **Report Lines** section only appears when the report has **no SQL connection assigned**, i.e. when the data adapter is `dyn. Array`. In this case, VeloxFactory has no database to query, so the detail band data must be entered manually in the browser.

The section shows a table with one column per field defined in the report. Each cell contains a typed input matching the field's data type. You fill in one row of values per data record you want to appear in the report.

To add more rows, use the **Add Row** button, it clones the input row and appends a new empty one. Individual rows (except the first) can be removed with the delete button on the right. Empty fields are not transferred to the render request.

i When a SQL connection is assigned, the Report Lines section is hidden. VeloxFactory fetches the data automatically from the database using the configured query, no manual input needed.

Less typing: prefill, example values, clear

Three small helpers remove most of the repetitive work when the same report is rendered again and again:

- **Prefill from the last run.** Parameters and report lines come back filled with the values of your previous render in this session. The values are stored on every submit, successful or not, so a failed render can be corrected instead of retyped.
 - **Fill example values.** Fills every parameter and the first line row with the example values stored in the report configuration. This is the frontend counterpart of the API's `useExampleValues`, useful for a quick smoke test of a new template.
 - **Clear inputs.** Empties all parameters and removes every line row except the first one.
-

Resources

If the report has image resources, a collapsible **Resources** section is available on the page. It shows a preview thumbnail of each resource file, its parameter name, and its file name. Resources are handled automatically at render time, you do not interact with them during rendering. The section is informational only, confirming which image files are currently assigned.

Generating

Once the values are in place, click **Generate**. There is no confirmation dialog: the stored output settings already say what is supposed to happen with the result. The request is processed synchronously, the page waits for the result and displays it immediately.

The Result

Success

In the two modes with a preview, the page shows a green confirmation banner, a **Download PDF** button and the generated PDF as an inline preview, sized to the report's actual page dimensions. If a history record was created, the download button serves the stored file, otherwise the PDF is

offered straight from the response.

In **Print only**, no preview is rendered at all. The banner then reads *Sent to <printer> (<n>x)*, which is the whole point of that mode: a station that prints and moves on.

If a history record was created, a **View History** button links to the new `ReportHistoryRecord`. If a print task was dispatched, a **View Print Task** button links to the `ReportPrintTask` where you can follow its status.

⚠ **Reloading the page renders the report again.** The result lives in the response, so a browser reload repeats the render, including a new print task if the mode creates one.

Screenshot placeholder: `generate-pdf.result.png`

Generate PDF page after a render, with the result block and the inline PDF.

Errors

If the render fails, the page shows a red error banner listing all error messages returned by VeloxFactory, and everything you entered stays on screen. Common causes are missing required parameters, a SQL query that returns no rows, or a resource file that was removed after the configuration was last saved.

If a history record was created before the error occurred, the failed attempt is recorded including the error details, which is useful for diagnosing what went wrong.

Revision #10

Created 2026-05-10 10:44:05 UTC by Benjamin Fischer

Updated 2026-09-21 10:54:30 UTC by Benjamin Fischer