

# Importing a Model into SAP Analytics Cloud

## A step-by-step guide — Rolling Forecast accelerator

*Tested end to end in a live SAC tenant. This is the walkthrough that ships with the accelerator — including the three points where a first-time import quietly goes wrong.*

---

### What you'll end up with

A working SAC model loaded with Actual / Budget / Forecast data, and a Story showing all three versions side by side — Business Unit and Month down the rows, Version across the columns, Revenue in the cells. You supply your own figures; the accelerator supplies the structure that's already correct.

### Part 1 — Create the model from the file

- 1 Open the **Modeler** (main menu → Modeler).
- 2 **Create New** → **Model**, then choose **Import a file / from a CSV**.
- 3 Select your CSV, keep **Use first row as column headers** ticked, leave the delimiter on *Auto-detect*, confirm.
- 4 SAC shows the columns. For the sample data: Version, Business Unit, Month, Revenue, EBITDA. Numeric columns become Measures, the rest Dimensions. Save the model (a star by the title means unsaved).

#### Pitfall 1 — the system Version dimension

When you save a Planning model, SAC adds its own **Version** dimension automatically (often shown as *Version2*). This is expected and required — leave it alone. Your own *Version* column (Actual / Budget / Forecast) is a separate dimension, and that's the one your comparison uses. Don't confuse the two.

### Part 2 — Load the data (the step most people get wrong)

Open **Data Management** (the dropdown at the top-left of the model).

#### Pitfall 2 — Upload Job vs Import Job

SAC offers two ways to bring in a local file. An **Upload Job** only *configures* a load — it does not run from the job list (status stays "set up / Not yet run", with no Run button). This is the dead end that traps most first-time users. An **Import Job** is the one that actually loads, ending with a **Run Import** button. Use the Import Job path.

- 1 In the **Import Jobs** section, start a new import from a **File** and select your CSV.
- 2 **Step 1 — Prepare Data:** check rows and columns, then Next.
- 3 **Step 2 — Map to Facts:** confirm *Version* maps to your Version dimension, not the system one. Next.
- 4 **Step 3 — Map Properties:** the "public.Actual" default note refers to the system Version — expected. Next.
- 5 **Step 4 — Review Import:** aim for **0 rows rejected**, then **Run Import**.

- 6 Confirm settings (Import Method *Update*, Reverse Sign *Off*) and **Finish**. Leave the tab open until it completes.

When done, the Import Jobs list shows a real **Last Run** date and green status ticks. That's your confirmation the data is in — not before.

## Part 3 — Build the Story (Actual / Budget / Forecast side by side)

- 1 Go to **Stories** → **New Story**, choose a **Canvas** page.
- 2 In **Add data to story**, pick **Model or Dataset** and select your model (not *File*).
- 3 From the left panel, add a **Table** and draw it on the canvas.
- 4 **Columns** → **Measures**: add **Revenue** (clears the "at least 1 measure required" prompt).
- 5 **Rows** → **Add Dimension**: add **Business Unit**, then **Month**.
- 6 **Columns** → **Add Dimension**: add **Version** (your Actual/Budget/Forecast dimension — not the system one).
- 7 Scroll horizontally to confirm all three versions show, then save the Story (File → Save).

### Pitfall 3 — removing a filter can delete the widget

While editing, deleting a filter can remove the whole table if it's the active object. Build the table first (measure + rows + columns), check the result, and only then touch filters. If a table disappears, check **Outline** → **Pages** before rebuilding.

## Why this guide exists

None of the three pitfalls above shows up in a generated CSV or a prompt's output. They only appear inside a live tenant — which is exactly the work a tested accelerator has already done for you. Generating the model is the easy part; getting it loaded and displayed correctly is the job.