Amtiri

USER MANUAL

Moving Configuration Between Installations

Generated on November 25, 2026

An instance's configuration can be taken out as a zip archive and brought into another instance: all of it, or only the records you choose. This guide covers both directions, and how to take one plant out of an instance that holds several.

Both actions are in the **Configuration** section of the Site's menu: **Export** and **Import**. Both need the engine stopped - press **Stop** first - and both need the Configuration role.

What travels	What never travels
Schemas, peers, variables, modules and panels, chosen record by record	Users and roles
A persistent variable's current value, when you include it	The licence and the instance's identity
A logger's recorded history, when you include it	Anything outside the configuration
A peer's shared secret, when you include it	

Exporting

Export opens a list of every record in the instance, grouped by kind, each with a checkbox. Everything is ticked to begin with, which exports the whole configuration. Groups with more than 50 records start collapsed and show 100 at a time.

Control	What it does
Search code or name	Filters every group at once
Select matches / **Clear matches**	Ticks or clears every record the search and filters show
Module type and view kind	Narrow the list to one kind of module or view
Include what it needs	When on, ticking a record also ticks what it refers to. Each added row says *needed by* and names the record that needed it
All	Ticks or clears a whole group
Dependencies	Ticks this record and everything it needs, even with **Include what it needs** off
Users	Ticks every record that refers to this one

Ticking a record never ticks the records that *use* it. Exporting a variable does not bring every panel that shows it; press **Users** on its row when you want those too. A record added by **Include what it needs** can be unticked, and stays out.

Some parts of a record are chosen on their own row:

Part	Offered on	Default on export
Value	Persistent variables	Included - a persistent variable's current value is kept in its own record, so it always travelled
History	Loggers	Not included - a logger's database can be large
Secret	Peers	Not included - a peer without its secret cannot connect until someone types it in again

Each group also has an **All values**, **All history** or **All secrets** switch.

Below the list, *Expected on the target* names what the ticked records refer to but the archive will not carry. The instance you import into should already have those records.

Taking one plant out of an instance that holds several

Suppose an instance runs three plants, and one of them - the talc dosing station - has codes that all start with `Talco_`.

1. Press **Stop**, then **Export**.
2. Press **Clear matches** with the search empty. Nothing is ticked now.
3. Type `Talco_` into **Search code or name**, and press **Select matches**.
4. With **Include what it needs** on, the station's variables and its `Talco_HMI` panel are ticked, and so is anything else they read. A variable the panel shows but whose code does not start with `Talco_` is added too, and its row says *needed by Talco_HMI*.
5. Read *Expected on the target*. If it names only a role, the station is self-contained.
6. Press **Export**. The summary counts what the archive holds.

The records the other two plants share with the station - a logger that records its variables, say - are not added, because they use the station rather than being used by it. Press **Users** on a row to bring them.

From the **Program** screen you can also start from the diagram: select nodes, then press **Export the selection to another installation**. The export opens with those records ticked and what they need added. Save pending changes first; the export reads the saved configuration.

Importing

Import asks for an archive, then lists every record in it, and says on each row what importing it would do.

Row says	Meaning
New	Nothing here answers to its code; it is created
Replaces	It replaces the record here that answers to its code, which keeps its id
Duplicate as `X_IMP`	It comes in beside the record here, renamed
Unchanged	The record here already holds exactly this; nothing would be written

Where a code is already in use decides between the last three: **Bring it in alongside, as _IMP**, or **Replace what is there**. It applies to the whole import. Codes are matched without regard to case, and variables and modules share one set of codes.

What starts ticked:

- Every record, except one that would be **Unchanged**.
- A persistent variable's **Value**, when the record is new or brought in alongside. On **Replace what is there** it starts unticked, so the instance keeps its live value. A value the archive did not carry cannot be ticked, and never empties the value here.
- A logger's **History**, when the archive carries it. Replacing a history moves the old database aside under a new name, and the summary gives that name.

Changing **Where a code is already in use** updates these defaults, but never a tick you set yourself.

Where references land

A ticked record may refer to one that is not being imported. Only records coming in are renamed, so after the import that reference names whatever this instance has under the code. Each row says which of three things will happen:

Row says	What it means
X will be this installation's own	This instance has a record under that code, and the reference will use it. Check the name shown: it may be a different record that happens to share the code
X is in the archive but not ticked	Tick it to bring it along
X is neither here nor in the archive	The reference will name nothing. The import still goes ahead, and the warning stays in the summary

Press **Import** to write. If any record fails to write, everything already written is put back. Press **Start** afterwards for the imported configuration to take effect.

Archives from other versions

An archive stays readable in both directions. An older version of the Site imports an archive that carries only some records, and a newer version imports an archive made before records could be chosen, reading it as the whole configuration with every value included.

Troubleshooting

Message	What to do
The formula in X could not be read	The formula does not compile here, so what it refers to is unknown. Open it and check its references by hand
X refers to Y, which is neither here nor in the archive	Tick Y if the archive carries it, import it separately, or create it after the import
X is restricted to a role this installation does not have	The view arrives but nobody can see it until the role exists. Roles never travel
Export and Import are greyed	The engine is running. Press **Stop**