Tag Reference Open Display Example

How to have one display to share between many equipment, instead of having one display for each equipment.

How-to **Examples** **Feature** **UNS** Tag Reference Open Display Example

---

Two distinct "reference" concepts in FrameworX — do not confuse

This page is about (a) the `Reference` tag type — a runtime tag whose `.Link` property points to another tag of the same UserType. It is a runtime pointer / alias mechanism used to share a single display across many equipment instances.

The other concept is a **`Reference`-type UDT member**: a member declared with `Type=Reference` whose per-instance value points to another tag's `/Attr` envelope (for example, `Mixer_M101.feedsInto = Reactor_R101/Attr`). That second concept is the FX representation of an OWL `owl:ObjectProperty` — an ontology relation between named individuals — and round-trips through the RDF /OWL importer and exporter.

If you are modeling ontology relations (asset-to-asset edges, equipment-to-process links), see Industrial Ontology Integration How-to and Import an OWL/RDF ontology into your UNS instead.

Download the solution: Tag Reference Display Open Example.dbsln

This example demonstrates the use of the Reference Tag type to open displays.

Overview

When you click any of the buttons, a display will open and a reference tag will be linked to the corresponding machine. This way, you can use a single display for all machines — instead of creating one display per machine — simply by changing the reference tag.

Data Model (UNS)

UserType: Machine

All machines share the same UserType, which defines the following process variables:

Member	Type
Temperature	Integer
Pressure	Integer
Power	Integer

Tags

Tag	Type	Domain	Description
Machine1	Machine	Server	Instance of the Machine UserType
Machine2	Machine	Server	Instance of the Machine UserType
Machine3	Machine	Server	Instance of the Machine UserType
RefMachine	Reference (Machine)	Server	Reference tag — points to whichever machine is currently selected
EquipNumber	Integer	Server	Drives the reference tag during Previous/Next navigation

Simulated Values

Machine	All Process Variables
Machine 1	10

Machine 2	20
Machine 3	30

Displays

MainPage

The home screen. Contains three instances of the **MyButton** symbol, one per machine.

When clicked, the button:

1. Sets `@Tag.RefMachine.Link` to the name of the selected machine tag
2. Sets `@Tag.EquipNumber` to the corresponding machine number
3. Opens the **MachineStatus** display

MachineStatus (Shared Display)

This is the single shared display used for all machines. All bindings reference **RefMachine**, never a specific machine tag directly.

Element	Binding
Temperature TextBox	@Tag.RefMachine.Temperature
Pressure TextBox	@Tag.RefMachine.Pressure
Power TextBox	@Tag.RefMachine.Power

Because all bindings go through `RefMachine`, the same display automatically reflects whichever machine is currently linked, no need for 3 different displays.

MyButton Symbol

The buttons on MainPage are instances of the **MyButton** symbol. Using a Symbol instead of a plain button means any visual or behavioral change made to the symbol definition is automatically applied to every instance across all displays.

Symbol Labels (Inputs)

Label	Description
Equipment	Tag reference for the target machine (e.g. @Tag.Machine1)
EquipmentNumber	Integer identifying the machine (e.g. 1, 2, 3)

Action Dynamic (on click)

The button executes three expressions on `MouseLeftButtonDown`:

Expression	Result
`#Equipment:(@Tag.Machine1).GetName()`	Written to `@Tag.RefMachine.Link` — links the reference tag to the selected machine
`#EquipmentNumber:1`	Written to `@Tag.EquipNumber` — updates the current equipment number
`@Display.MachineStatus.Open()`	Opens the MachineStatus display

Dynamic Label

The button label is built at runtime:

```
{#Equipment:(@Tag.Machine1).GetName().Replace("Tag.", "")}
```

This strips the `Tag.` prefix so the button displays a clean name like **Machine1**.

To inspect the symbol configuration: right-click any button **Edit Symbol**, then check **Settings** **Label Text** and **Dynamics** **Action**.

For more information about Symbols and how to use mnemonics (#Equipment and #EquipmentNumber), refer to the Symbols Control Reference.

Navigation Header (Previous / Next)

The Header display includes **Previous** and **Next** buttons that are only visible when the **MachineStatus** display is open. They allow the operator to cycle through machines sequentially without returning to the main page.

An `EquipNumber` tag (Integer) drives the reference tag change.

Next Button Action

Expression	Result
`TK.IIf(@Tag.EquipNumber > 2, 1, @Tag.EquipNumber + 1)`	Written to `@Tag.EquipNumber` — increments with wrap-around
`"Tag.Machine" + @Tag.EquipNumber.ToString()`	Written to `@Tag.RefMachine.Link` — updates the reference tag

Previous Button Action

Expression	Result
`TK.IIf(@Tag.EquipNumber < 2, 1, @Tag.EquipNumber - 1)`	Written to `@Tag.EquipNumber` — decrements with wrap-around
`"Tag.Machine" + @Tag.EquipNumber.ToString()`	Written to `@Tag.RefMachine.Link` — updates the reference tag

Both buttons use `VisibilityDynamic` with condition:

```
@Display.MachineStatus.IsOpened == true
```

So they only appear when the MachineStatus page is active.

Reference Tags — How They Work

A Reference tag is a special tag type that acts as a pointer to another tag of the same UserType. Instead of binding displays or actions directly to a specific machine tag, you bind them to the reference tag — and simply change what it points to at runtime.

In this demo, `RefMachine` is a Reference tag of type **Machine**. All displays bind through it, so switching which machine it points to is enough to update the entire screen.

The `.Link` Property

The `.Link` property is what you read or write to control which tag the reference is pointing to. It holds the target tag name as a string.

Usage Examples

Assigning directly from a tag name:

```
@Tag.RefMachine.Link = Tag.Machine1.GetName()
```

Building the link dynamically by concatenating a string with a tag value:

```
@Tag.RefMachine.Link = "Tag.Machine" + @Tag.EquipNumber.ToString()
```

Where This Pattern Appears in This Demo

Location	Element	How Reference Is Used
MyButton Symbol	Action Dynamic (Expression 1)	`#Equipment:(@Tag.Machine1).GetName()` written to `@Tag.RefMachine.Link` — links the reference to the clicked machine
Header — Next Button	Action Dynamic (Expression 2)	`"Tag.Machine" + @Tag.EquipNumber.ToString()` written to `@Tag.RefMachine.Link` — links by composing a string
Header — Previous Button	Action Dynamic (Expression 2)	Same pattern as Next, with decrement logic
MachineStatus Display	TextBox bindings	`@Tag.RefMachine.Temperature`, `@Tag.RefMachine.Pressure`, `@Tag.RefMachine.Power` — reading members through the reference
MachineStatus Display	Debug TextBox	`#Value:@Tag.RefMachine.Link` — displays the current link value (which tag is active)

In this section...
