

# ProviewR

OPEN SOURCE PROCESS CONTROL



## Example Library

Copyright (C) 2005-2026 SSAB EMEA AB

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

## 2 Introduction

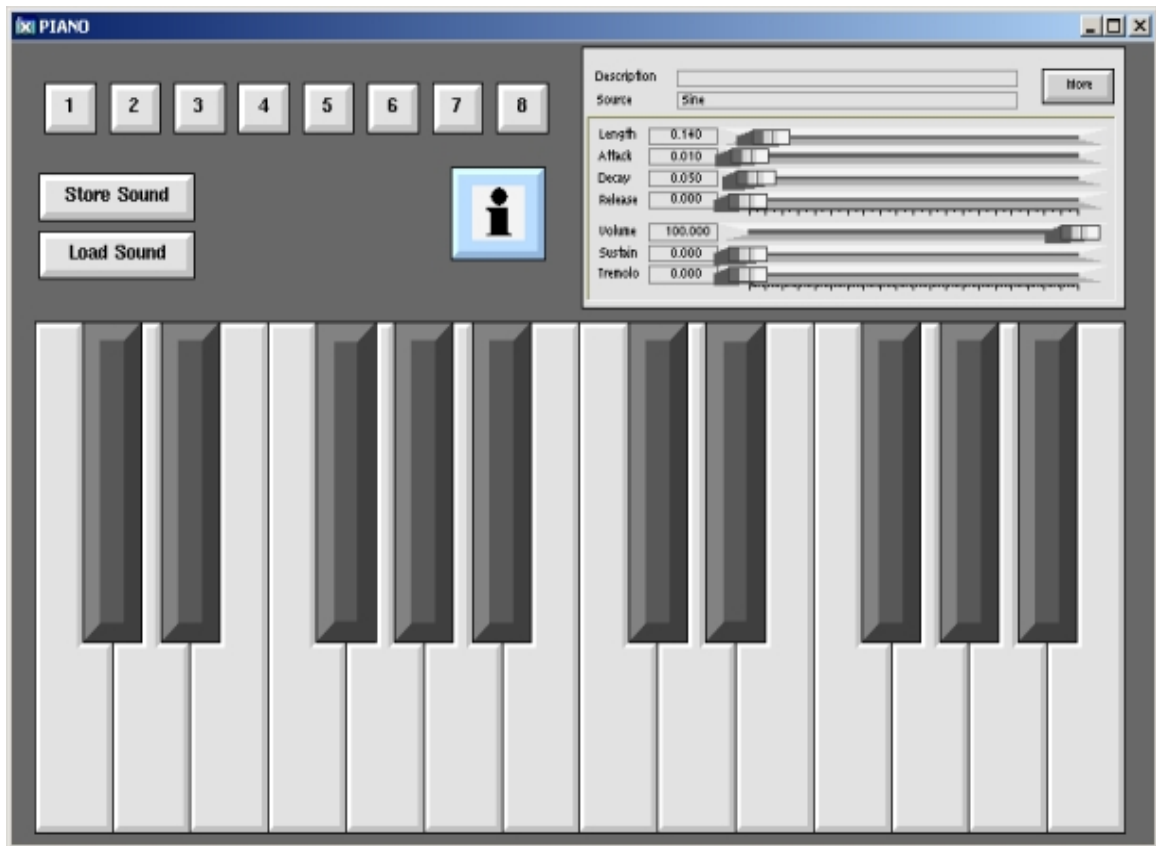
This document is a collection of Proview examples and applications.

### **3 Plc Functionobject examples**

## 3.1 DataFRead and DataFWrite

DataFRead and DataFWrite are used to store the content of a data object on file, and later read back the content to the same dataobject, or to another dataobject of the same class.

In this example we show how the files can be managed by the operator from a fileview. The operator has made some settings that is stored in a dataobject, and he can store different settings with different filenames, this is done from a fileview in save mode. When he wants to restore a setting, he opens a fileview in open mode, and can select a file from a list of files he has previously stored.

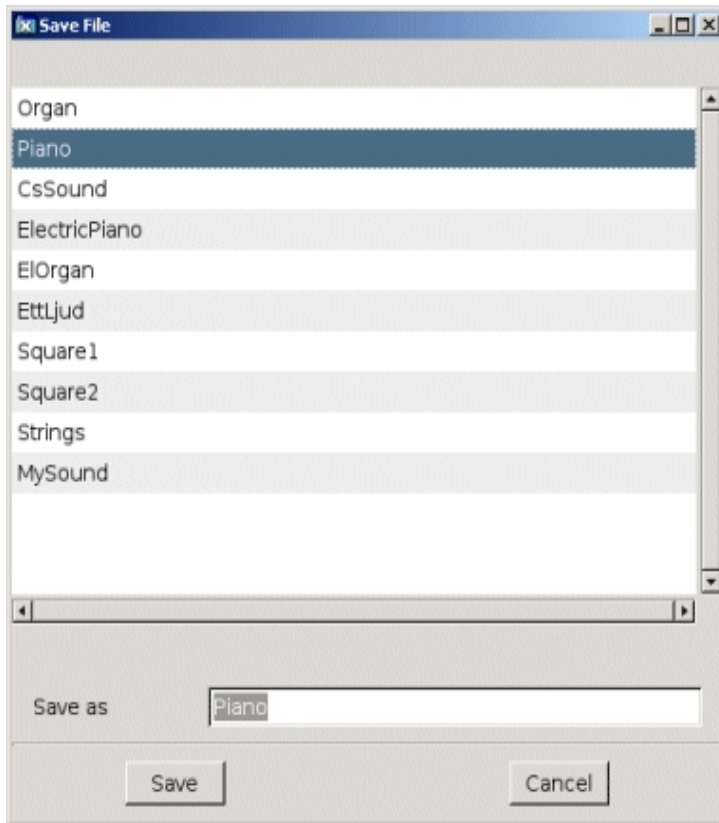


**Piano graph**

In this simple Proview synthesizer the user can affect the sound with the sliders to the right in the Piano graph. When he is satisfied, and wants to store the settings, he presses the "Store Sound" button. This is a CommandButton with the command

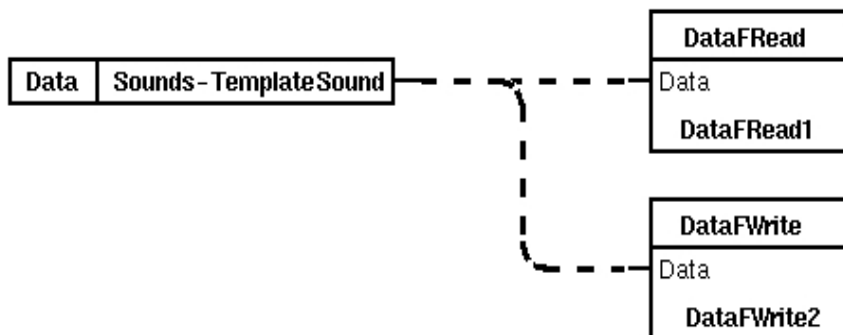
```
open fileview /type=save /file="$pwrp_load/sound/*"  
                /target=Piano-Plc-W-DataFWrite2.FileName  
                /trigg=Piano-Plc-W-DataFWrite2.Condition
```

When the button is pressed the Xtt fileview is opened in save mode. It will list the files "\$pwrp\_load/sound/\*" in the file list.



### Fileview to store data

When a filename is entered, the filename is stored in the DataFWrite object specified in the target switch, and Condition attribute in the same object is set with the /trigg switch. This will trigger the DataFWrite object to perform a write of the TemplateSound object, connected to it by a GetData.



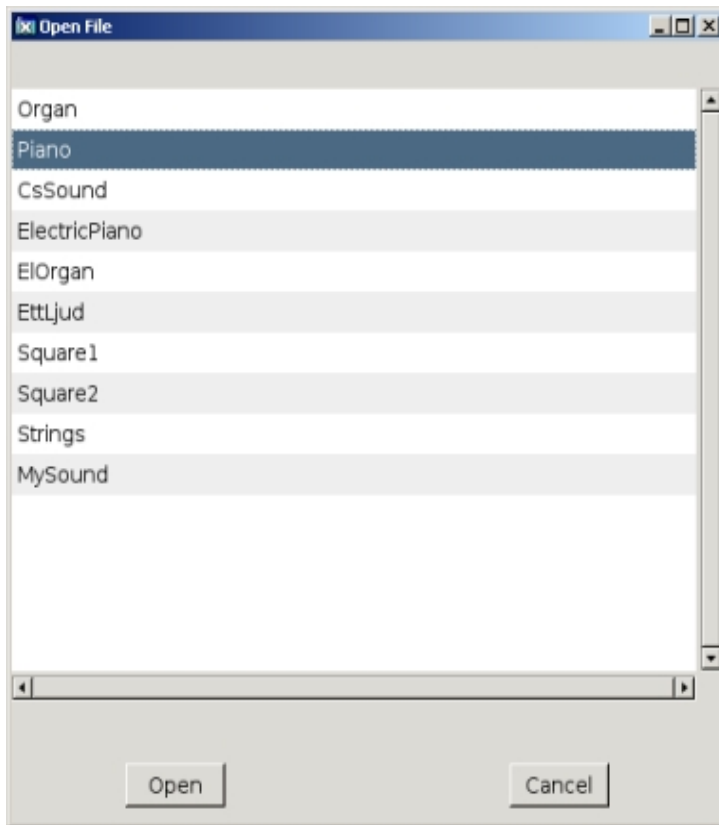
### Plc code

To restore a setting the "Load Sound" button is pressed. This is a CommandButton with the command

```

open fileview /type=open /file="$pwrp_load/sound/*"
                /target=Piano-Plc-W-DataFRead1.FileName
                /trigg=Piano-Plc-W-DataFRead1.Condition
  
```

It will open the fileview in open mode.



#### **Fileview to restore data**

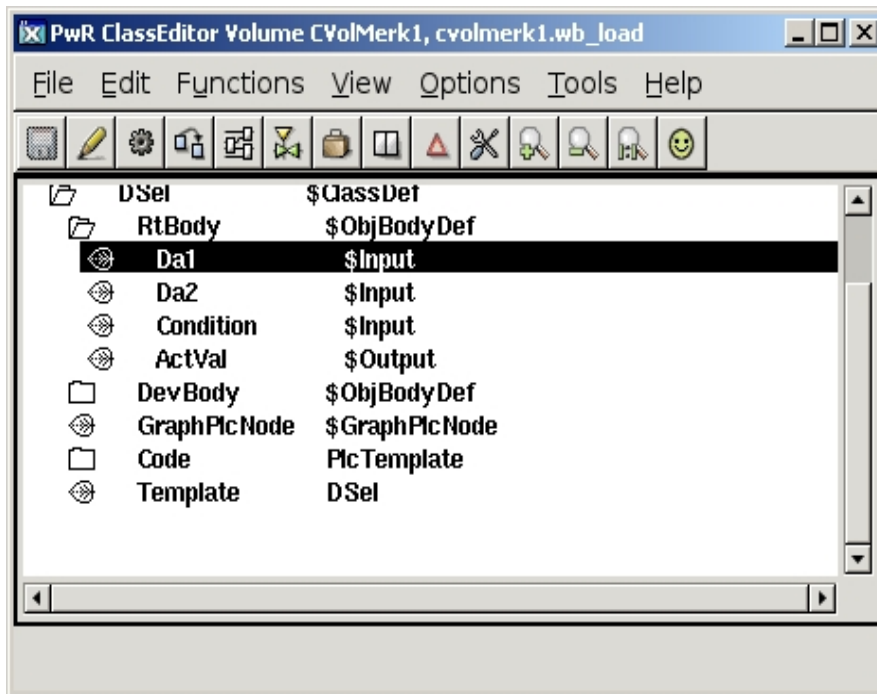
When a file is selected, the filename is written to the DataFRead object in the plc program, and also the Condition attribute is set. This will trigger a read of the specified file, and the read data is store in the TemplateSound object connected to the DataFRead object.

Note that the plcprogram (usually with owner pwrp) has to have write access to the directory where the files are stored, and the operator has to have read access.

# 3.2 PlcTemplate

## Example of PlcTemplate with Data inputs

The data inputs Da1 and Da2 are defined by \$Input attributes with TypeRef \$Float32 and with the pointer bit set in Flags. These inputs can be connected to data output pin in for example GetData, DataArithm, DataSelect, DataCopy etc.



Classdefinition of data input pins

In the plc template code the inputs are fetched with GetDataInput objects that has a symbolic reference to the input attributes with \$PlcFo.

