

## Info Editor for VectorWorks®

[BACK TO PRODUCTS](#)[ADD TO CART](#)[VIEW CART](#)

### Price

\$20.00

### System Requirements

This plug-in requires VectorWorks 12.5.0 or above.

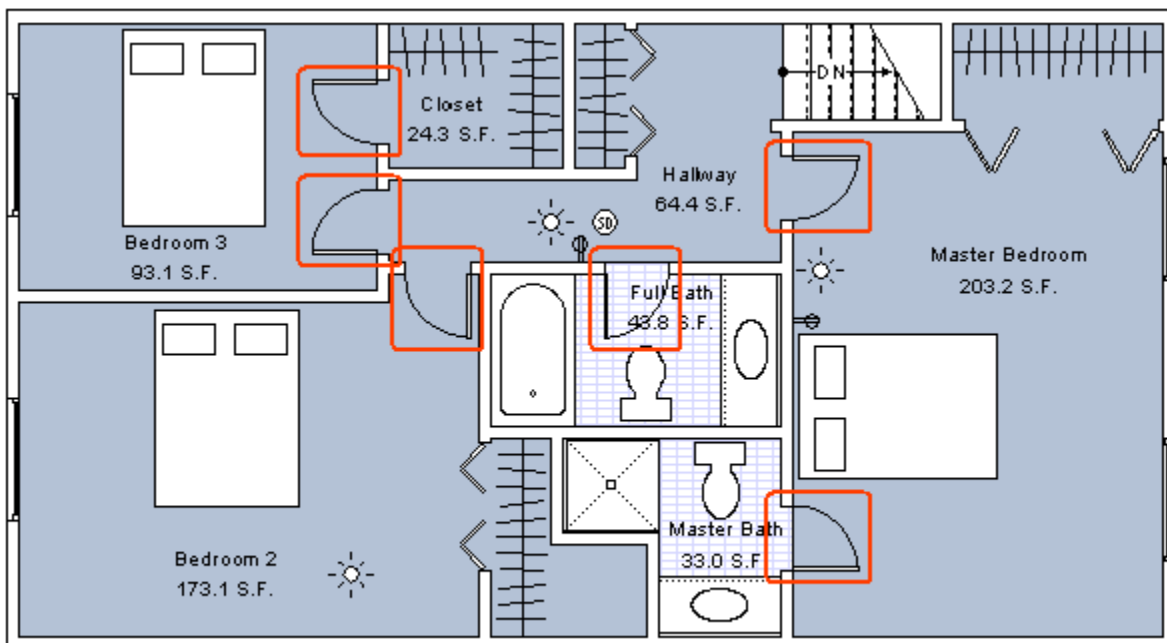
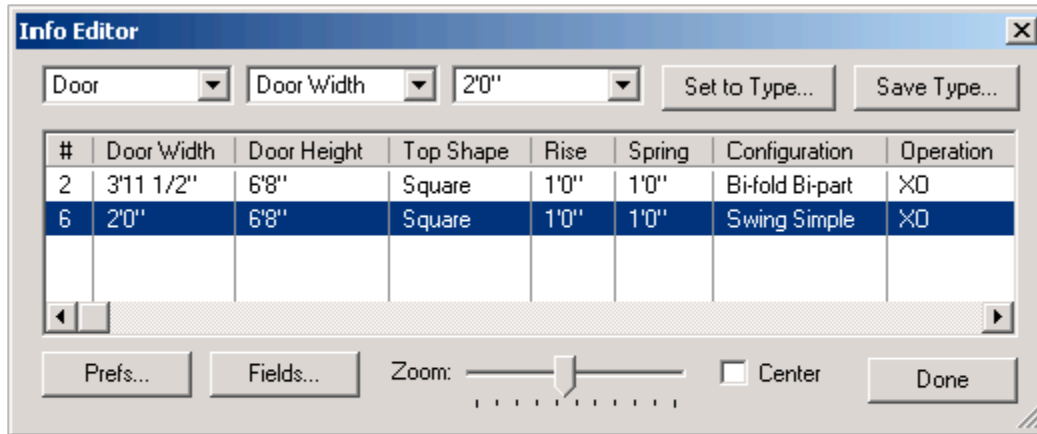
### Installation

- Copy the entire "Info Editor" folder into the Plug-Ins folder in your VectorWorks installation. Alternatively, you can copy this folder into your "user" folder, which will make the Info Editor available to any VW 12 installation on your system, but only to users logged on as you.
  - Macintosh: /Users/[username]/Library/Application Support/VectorWorks/12/Plug-Ins
  - Windows: C:\Documents and Settings\[username]\Application Data\Nemetschek\VectorWorks\12\Plug-Ins
- Re-start VectorWorks.
- Open the workspace editor.
- Add the *Info Editor* menu item to the workspace. It's in the SCS category.
- Close the workspace editor.

### Description

Large, complex drawings in VectorWorks typically contain many plug-in objects, such as windows and doors, and many symbols with attached records. Making global changes to these objects can be difficult sometimes. If the objects are inside walls, or inside different groups, or on different layers, making a global change requires navigating to many places in the document before the objects can be selected and changed. Just *finding* all of the objects that need to be changed can be difficult, because you have to set the class and layer visibilities properly before you can even get to them.

The Info Editor cuts across the structural boundaries in VectorWorks documents, providing direct access to field values in plug-in objects, and in data records attached to symbols. Simply choose the type of information that needs to be edited, and the Info Editor will show a list of all of the objects with that information, regardless of where they are in the document. In this list, one or more rows can be selected, and changes can be made to one or more fields. Upon closing the Info Editor, the changes are applied to the objects in the drawing.



But the Info Editor is way more than just an editable list of objects in the drawing — it has powerful features for presenting the information in a variety of ways, and for facilitating edits to one or more fields in one step.

- The list can be sorted by columns, making it easy to find objects with the same value in a particular field.
- Rows can be grouped on the basis of identical values in one or more fields. For example, there may be hundreds of doors in the drawing, but there may be only three different *sizes* of doors. To make a change to one of the door sizes, just group the records on the basis of width and height. Now there will be only three rows in the list, one for each distinct size, along with the count of doors represented by each row. If one of the rows is selected and a change is made, this change will be applied to all of the doors with that width and height.
- To make repetitive editing easier, you can define look-up lists for individual fields, so that when editing a particular field, you can pick from a list of pre-defined values,

rather than having to enter in the values each time.

- **Types** can be defined, as combinations of field values. These types can then be applied to objects, and all of the fields will be changed at once. This is very useful when making a change that involves more than one field, and it is important that the change be applied consistently across all of the objects. For example, the width, height, manufacturer, and part number might all go together, and when changing one, you will want to change all of them. If these four values are stored together as a type, they can be applied to other objects in one step, eliminating the possibility of mismatched values. These types are automatically stored in an external text file, so that they can be re-used in other projects, and they can be accessed by anybody on the local network. Types can be created directly from objects already in the drawing, or they can be created by importing information from tab-delimited text files. So if you can get a spreadsheet listing the characteristics of products from a vendor, you can easily convert these to Info Editor types, and apply these types to objects in the drawing.

Put together, these features make it easy to find and correct inconsistencies in drawings, and to explore design alternatives by making global changes without having to do a lot of work.

### To edit fields by picking from a list, or by manually entering values:

1. Click the **Info Editor** menu command. The Info Editor dialog opens. The dialog contains a large list box showing objects in the drawing with data that can be edited.
2. Above the list, at the left, are three pop-ups: one for the record name, one for the field name, and one for the current field value. Select the record and the field to edit. The field value pop-up will show the existing value for the selected object(s).
3. Select the row(s) to edit in the list box. The object(s) in the drawing associated with the selected row(s) are shown surrounded by a red box.
4. Click the value pop-up. It displays a list of all of the values already in use for this field by objects in the drawing.
5. Select from one of the existing values.
6. If the desired value is not already present in the pop-up, click the Add New Value... item to enter a new value. Click **OK** to add that value to the list and apply it to the selected object(s).

**Note:** *If the selected field is a pop-up field in a plug-in object, only the valid pop-up values will be available, and there will be no way to add a new value to the list.*

7. Click **Done**. Data records and/or plug-in object parameters will be updated with the selected values.

### To define look-up lists for individual fields:

1. Click the **Info Editor** menu command. The Info Editor dialog opens.
2. Click the **Prefs** button. The Info Editor Preferences dialog opens.
3. In the Data Folder pop-up, select the folder in which the Info Editor will store the data entered, or click the Edit List item to edit the list of available folders.
4. Click **OK** to close the Preferences dialog.
5. Select the record and field to edit at the top of the Info Editor dialog.
6. Click the Value pop-up. At the bottom of the list, click the Edit Data File... item.
7. The Edit List dialog opens.

8. Click the **Add** button to enter values that should appear in the list.
9. Click **OK** to close the Edit List dialog.
10. The values entered will now appear in the Value pop-up, and will appear as choices in the future whenever this record and field are selected.
11. Select the desired choice.
12. Click **Done**. Data records and/or plug-in object parameters will be updated with the selected values.

#### To edit records using types:

1. Click the **Info Editor** menu command. The Info Editor dialog opens.
2. Click the **Prefs** button. The Preferences dialog opens.
3. Click the **Choose Types File** button to select the file to which type definitions will be saved.
4. Click the **OK** button to close the Preferences dialog.
5. Select the record to be edited in the record name pop-up.
6. Select an object in the main list which has the appropriate values for each field.
7. Click the **Save Type** button. The Save Data as Type dialog loads.
8. Enter the name of the type in the box at the top.
9. Click in the Save column for each field which is to be saved with this type. A checkmark in this column denotes that the field value will be saved with the type.
10. Click **OK**. The type has now been saved in the selected types file.
11. Select other object(s) to which to apply a saved type.
12. Click the **Set to Type** button. The Select Type dialog loads.
13. Select the desired type. The fields and values associated with the selected type appear in the bottom list.
14. Click **Rename Type** to rename a type.
15. Click **Remove Type** to remove a type from the type file.
16. Click **OK** to accept the selected type. The values from the selected type are applied to the selected row(s) in the main list.
17. Click **Done**. Data records and/or plug-in object parameters will be updated with the selected values.

#### To import types from a tab-delimited text file:

1. Obtain or create a tab-delimited text file containing the type definition information. The first line of the file should contain a tab-delimited list of field names, and the rest of the file should contain tab-delimited lists of field values. One of the fields should contain a description which will work well as a type description, which you will use when selecting types to apply to objects. Values in this field must all be unique.
2. Click the **Info Editor** menu command. The Info Editor dialog opens.
3. In the record name pop-up, select the record to which the imported types will apply.
4. Click the **Prefs** button. The Preferences dialog opens.
5. Click the **Choose Types File** button to select the file into which the types will be imported.
6. Click the **Import Types** button. The standard file locator dialog opens.
7. Select the tab-delimited text file to import.
8. Click **OK**. The Import New Types for [Record Name] dialog opens.
9. Select the field to use for the description in the pop-up at the top of the dialog.
10. In the list box, match the fields in the text file to fields in the record. The fields are initially mapped on the basis of identically matching field names. The mappings can be changed by clicking in the Source Field column, which toggles among the available source field names.

11. Click **OK** to accept the settings. The information in the text file is converted to Info Editor types and stored in the selected types file.
12. Click the **OK** button to close the Preferences dialog.
13. Use the **Set to Type** button to apply types to selected row(s) in the main list.
14. Click **Done**. Data records and/or plug-in object parameters will be updated with the selected values.

Once types have been imported, they can be updated simply by re-importing the same text file. If a type being imported matches the description of a type already defined, the existing definition is updated. This means that the data can be maintained in a tab-delimited text file, and the Info Editor types file can be easily updated from it whenever necessary.

#### **To edit objects grouped by distinct values:**

1. Click the **Info Editor** menu command. The Info Editor dialog opens.
2. Click the **Fields** button. The Group By and/or Show Fields dialog opens.
3. Select the fields which all have to be identical in order for objects to be considered the same.
4. Click **OK**.
5. The list is repopulated with only the objects which have distinct values in each of the selected fields.
6. Select the row(s) to be edited, then the field to edit, and the value to apply. The change is applied to all of the objects represented by the selected row(s) in the list.
7. Click **Done** to close the **Info Editor** dialog. All of the affected objects will be updated.

#### **Extras**

In addition to editing information otherwise editable on the Object Info palette, the Info Editor also allows editing of hidden PIO fields and hidden data records. Just click the **Prefs** button, and check the appropriate boxes in the Preferences dialog. But caution should be exercised when editing information that was deliberately hidden by the programmer. In some cases, this information was hidden simply because there is some other way of editing it, and it's OK to edit this information using the Info Editor. In other cases, manually editing this information can result in corruption of the object instance. If the nature of the information is not 100% clear, be sure to make a back-up of the file before editing the hidden information, and thoroughly inspect the drawing before proceeding with other tasks.

There is also a preference named "Only Show Records in Use on Those Layers." When this is checked in the Preferences dialog, the Info Editor will filter out any record formats not used by objects within the selected layer scope. In larger drawings, this can be very slow, so use this preference carefully. Note that when this is not checked, the Info Editor will show all of the record formats in the document, even if there are not any objects anywhere in the document using that format, and even if that particular record format does not even have objects associated with it. For example, menu commands which were created in VectorScript will add a record format to the document, even though menu commands do not have objects associated with them.