

Introduction

This tutorial demonstrates the basics of creating a simple but complete VisiRule model that can be used for the EMDS Introductory Tutorial. EMDS is a program created by and supported by Mountain View Business Group and the EMDS Consortium.

Background on this tutorial

This decision tree is designed to assess the feasibility of a streamlined NEPA process in support of potential timber salvage projects following wildfire. Such salvage projects are expected to serve one of two purposes: 1) fiber recovery, or 2) ecological restoration. The intent for the fiber recovery branch of the decision tree is to identify conditions under which salvage poses minimal ecological risk (i.e., low probability of short-term and no long-term harm to system structure or functionality) and thus can proceed with a streamlined NEPA process. We sought to identify conditions about which little uncertainty exists in the scientific literature. Most nodes in the decision tree are the same under the fiber recovery and the ecological restoration branches. Where the ecological restoration branch relies on salvage to financially underwrite restoration under a streamlined NEPA process, it requires demonstration of projected long-term ecological benefits and assumes a greater willingness to accept short-term negative consequences than the fiber recovery branch.

Create a new VisiRule Model

Initially, let's cover some terminology which we need to use.

Terminology

Chart: A VisiRule chart is a structured diagram which captures expert knowledge and generates executable code in terms of the Flex Expert System toolkit.

In this tutorial we will be using VisiRule Author, which is a Windows desktop application, to build and test the model interactively.

Node: A node is any VisiRule object that can be connected to another VisiRule object. These are namely, questions, expressions, statements or code boxes, start and end nodes.

Questions: The basic VisiRule question node has 3 fields: a question name, a prompt and an explanation.

Question Types: VisiRule and Flex supports as distinct question types: single choice, multiple choice, number input, integer input, set input and name input. Each of these has its own icon on the VisiRule toolbar.

Expressions: An evaluable expression is attached to each exit point from a question or statement box and is used to control the execution flow. It is also used to build the set of available question answers as these boxes are harvested by VisiRule to populate the various questions generated at run-time.

VisiRule IDE is based on the underlying WinProlog IDE. This is a set of menus and any number of program files. Program files can be VisiRule charts, Flex KSL programs or Prolog programs. Each has its own syntax analyzer and compiler. VisiRule charts are 'compiled' using the 'Show Code' option which is on the Options menu. This generates executable code in the form of Flex KSL code.

In addition, VisiRule has a floating toolbar, which contains all the question types, expressions, start and end nodes, statement boxes and code boxes (Figure 1).



Figure 1. The floating toolbar of VisiRule.

Tutorial Task – Build a simple VisiRule decision tree

Launch VisiRule Author. Note that we will normally just refer to VisiRule for brevity.

There should be a VisiRule short-cut installed on your Windows desktop. Upon launching, you will see a blank canvass (Figure 2).

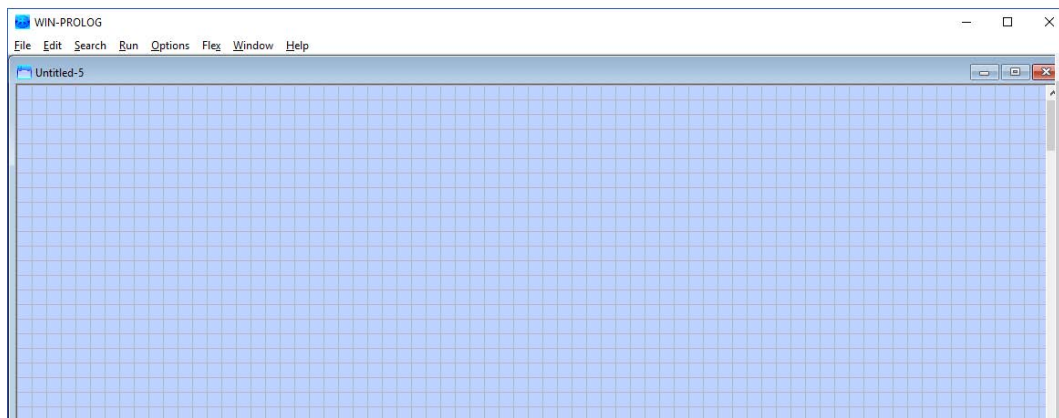


Figure 2. The VisiRule canvas is initially blank when VisiRule is launched.

We will start building our VisiRule Chart to assess whether the nature of the project and conditions on the site support a streamlined NEPA process.

The order in which we create the various nodes in a decision tree is not regulated, but it makes sense to start at the top (conceptually) and work downwards.

Every chart has at least one start node. So we select the start node box (Row 1 Item 2) from the floating VisiRule toolbar - this is green with a large arrow pointing to the right (Figure 3).

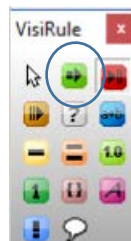


Figure 3. The start node is circled in the floating toolbar.

We click and drag on the empty canvass holding down the shift key. This creates a green start node prefilled with the word 'start1' (Figure 4). This is fine.

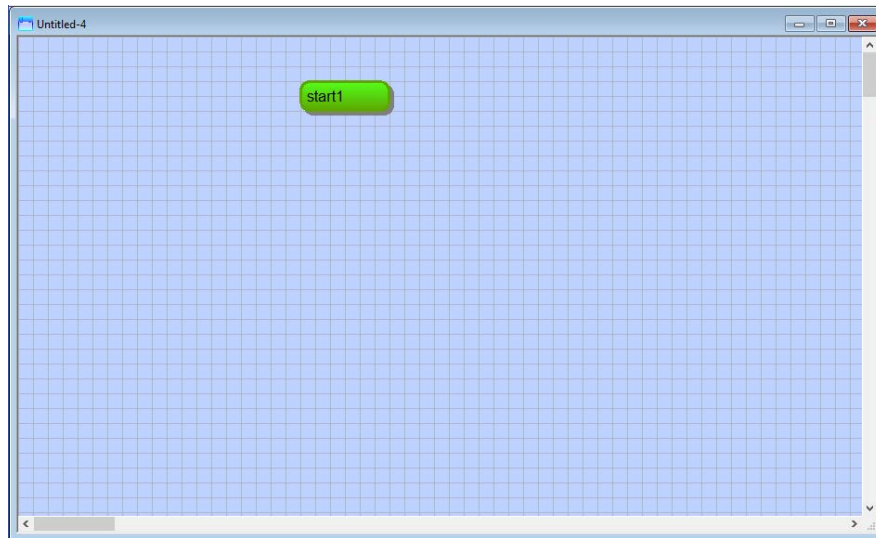


Figure 4. A start node has been added to the VisiRule canvas.

Next, we select the single choice question (Row 2, Item 1 VisiRule toolbox) - this is yellow with a single black horizontal line (Figure 5) - and click on the canvass just under the start box.



Figure 5. The VisiRule floating toolbar, with the single choice question item circled.

This creates a question box with 3 fields, each of which is prefilled with something like: Question1, Prompt, Explanation (Figure 6).

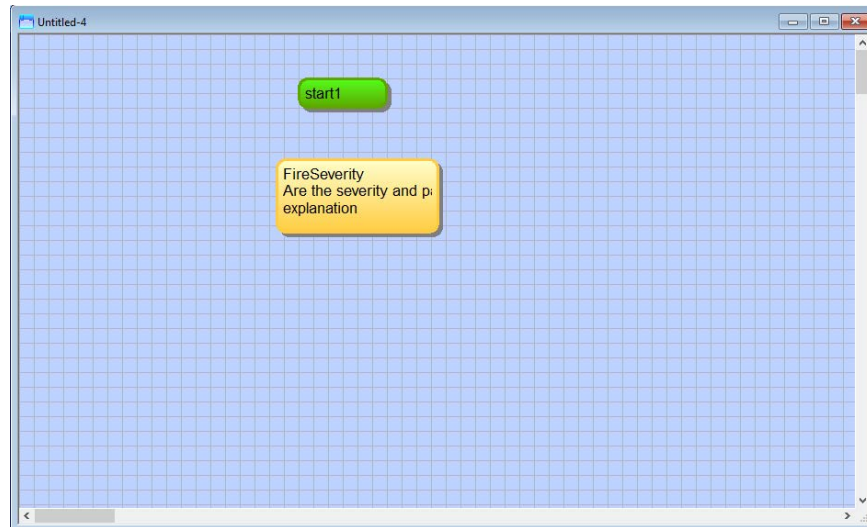


Figure 6. A single question node has been added to the VisiRule canvas.

Now we can edit the fields. To edit a field, simply click on the word, say question1, and then double click to select the text (Figure 7).

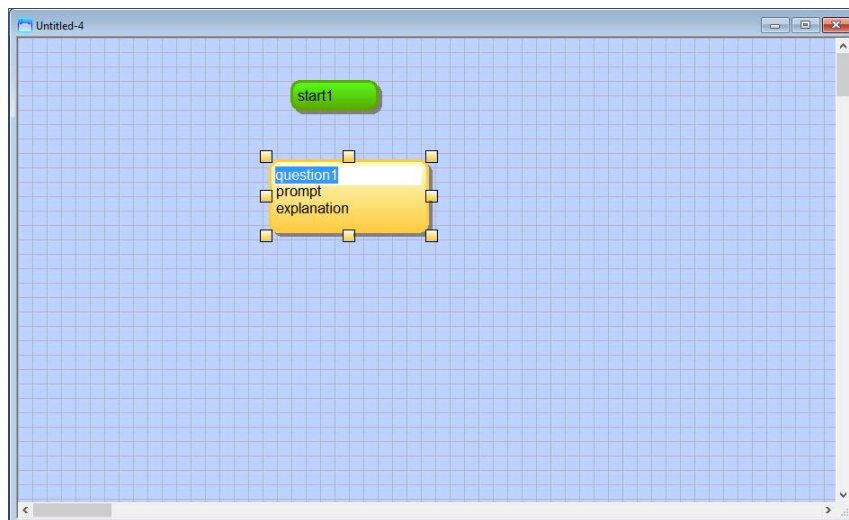


Figure 7. The single question node with the text “question1” currently selected for editing.

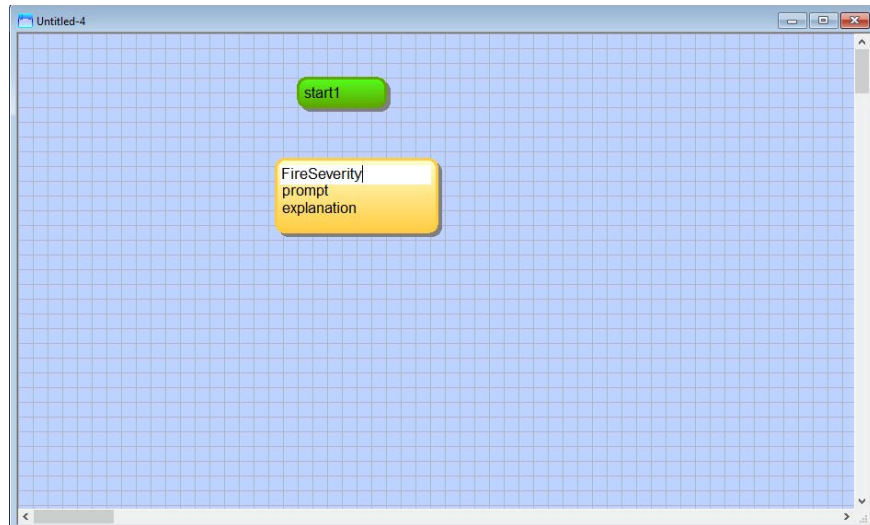


Figure 8. Single question node with its name field now edited.

Now start typing, say FireSeverity. This is the name we wish to allocate to the first question.

Now click on the Prompt field and type 'Are the severity and patch size typical?' The single question node now should like Figure 9.

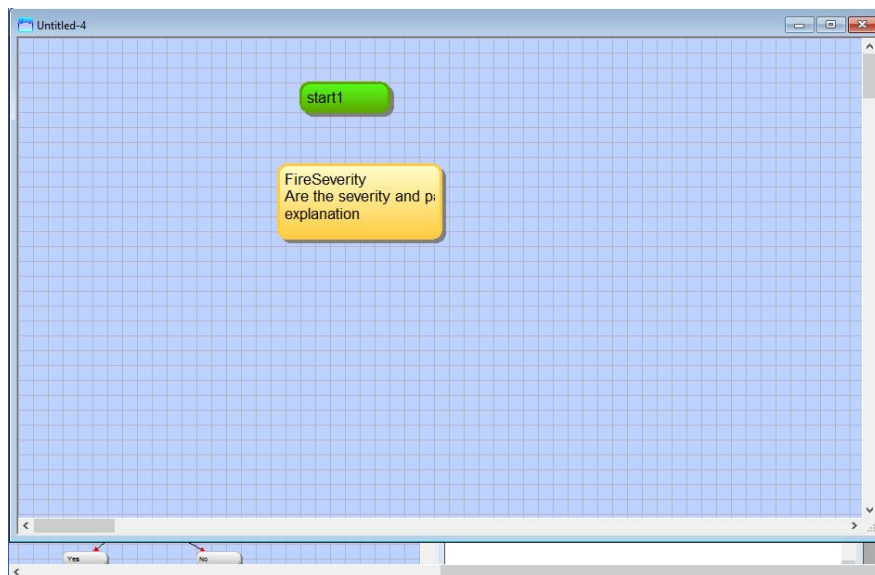


Figure 9. The single question node, with the question name and prompt edited.

You can tab to the next field which is the explanation. For now, we can leave the explanation field as is.

Next we will connect the start node to the question. We do this by simply clicking in the start node and dragging and releasing on the question node. This will create a directed link between the two (Figure 10). You can also resize the question node box using the resize handles (Figure 10).

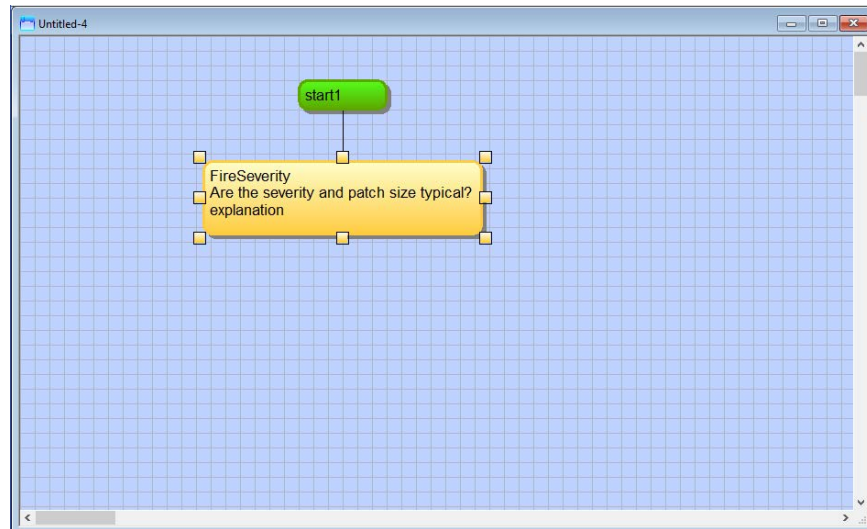


Figure 10. The start node is now connected to the first question node by a directed link.

We wish to offer 2 answers: Yes and No. So now we select the expression node (Row 2, Item 2 VisiRule toolbox) - this is white with a question mark - and click on the canvass just under the start box (Figure 11).

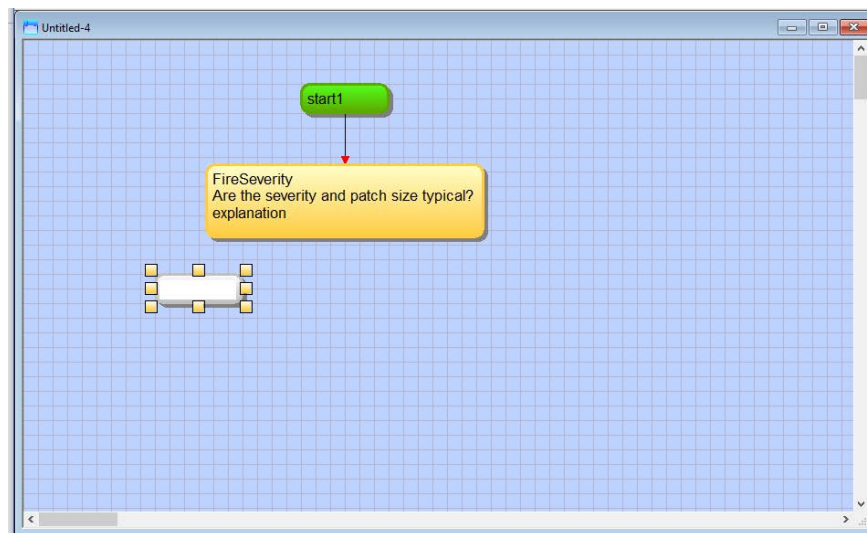


Figure 11. A blank expression node is added to the first question node.

We click on the canvass, down and to the left of our question and type 'Yes' (without the quotes).

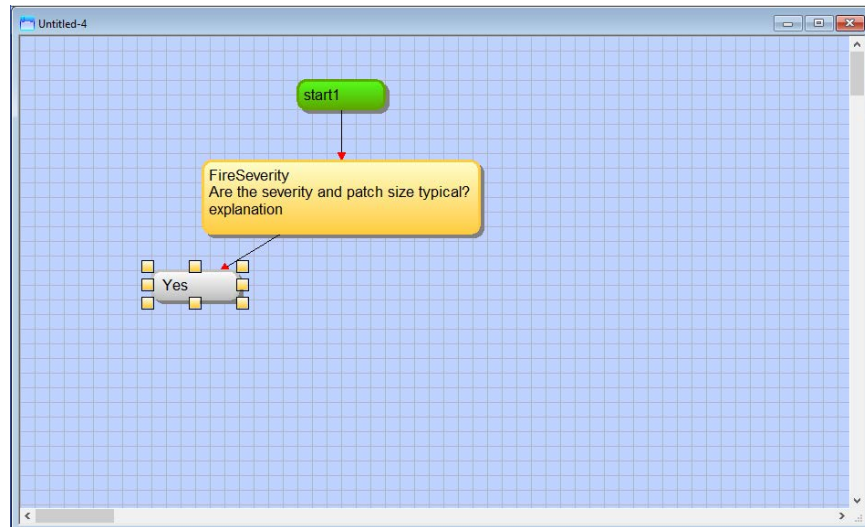


Figure 12. The first expression node now has the value 'Yes.'

To link these, we click in the question and drag a link to the Yes expression (Figure 12).

We then click on the canvas, down and to the right of our question, to add the next expression node, and type 'No' (without the quotes). Then we want to link these, so we click in the question and drag a link to the No expression (Figure 13).

Notice that we did not need to go back to the floating toolbar to select the expression node again in order to add the No expression. This is because, having selected the expression node to create the Yes expression, the expression node is still the active node type in the toolbar.

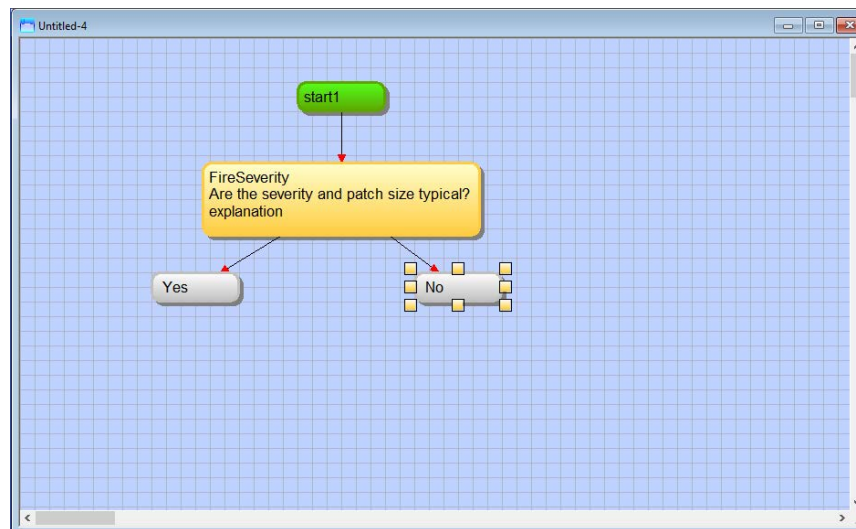


Figure 13. Our first question node concerning FireSeverity now has two expressions

Next, we will add two more questions, one on each branch, and then terminate part 1 of the VisiRule decision tree tutorial.

First, select the single choice question node in the toolbar, and click under the Yes branch. The name for the question will be FibreRecovery, and edit the prompt to ask, 'Is fibre recovery in small and medium site trees feasible?' Notice that we can force this prompt text onto two lines by inserting a hard return (<SHFT><ENTER>) in the appropriate place.

Next, with the single choice question node type still active, click to the right and below the No expression under the FireSeverity question. Name the new question node EcoRestoration, and give it the prompt, 'Is ecological restoration needed?'

Next, we will link these new nodes into the tree:

1. Click on the Yes node under FireSeverity, and drag it onto FibreRecovery.
2. Click on the No node under FireSeverity and drag in onto EcoRestoration.

Your decision tree to this point should now look like Figure 14.

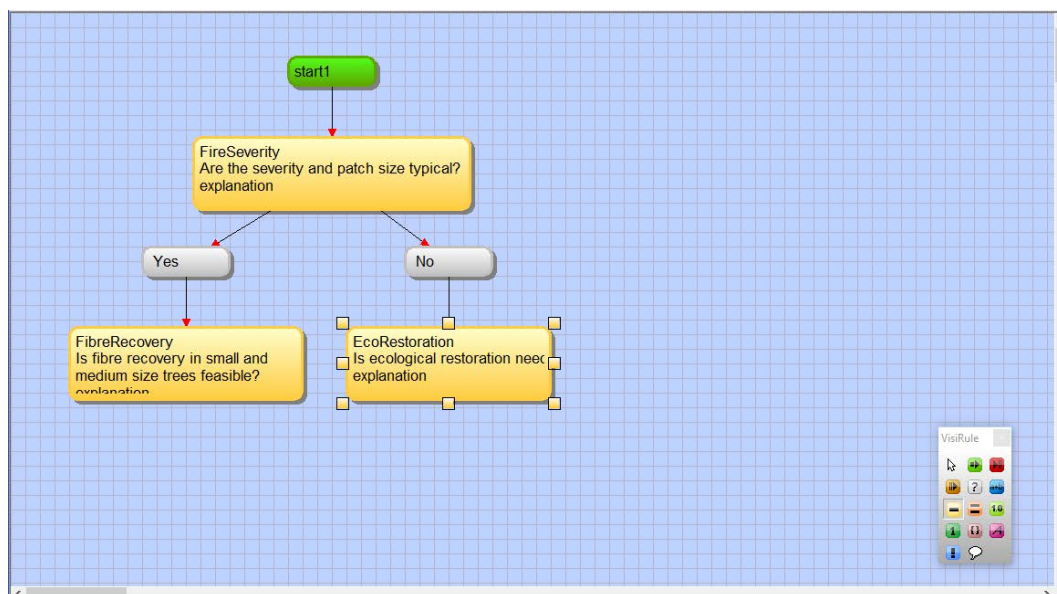


Figure 14. Decision tree with two new single question nodes for FibreRecovery and EcoRestoration.

Now let's add two expressions (No and Yes) **under** the FibreRecovery question node. Remember, though, to select the expression node in the toolbar first.

1. Click on FibreRecovery, and drag it onto the new Yes node **under** FibreRecovery.
2. Click on FibreRecovery again, and drag it onto the new No node **under** FibreRecovery

Now add two expressions under EcoRestoration (Yes and No), and link EcoRestoration to these two expression nodes.

Your decision tree should now look like Figure 15.

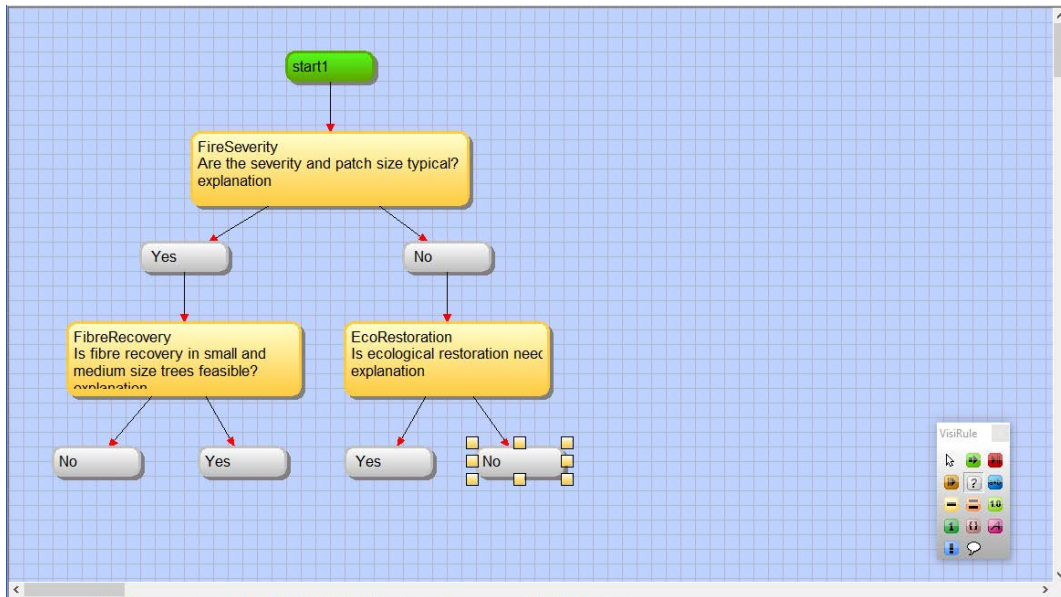


Figure 15. Decision tree with three single question nodes completed.

Now we have 4 'unfinished' nodes - which we will terminate for now with end nodes so as to give us a completed chart, which we can then compile and run. Later in we can always replace those end nodes with say additional questions or continue nodes so as to be able to continue the chart.

So now we select the End node (Row 1, Item 3 VisiRule toolbox) - this is red with an arrow head - and click on the canvas 4 times once underneath each expression. Replace the default text in the four new end nodes with the text shown in Figure 16.

Next we link from each expression to each end box.

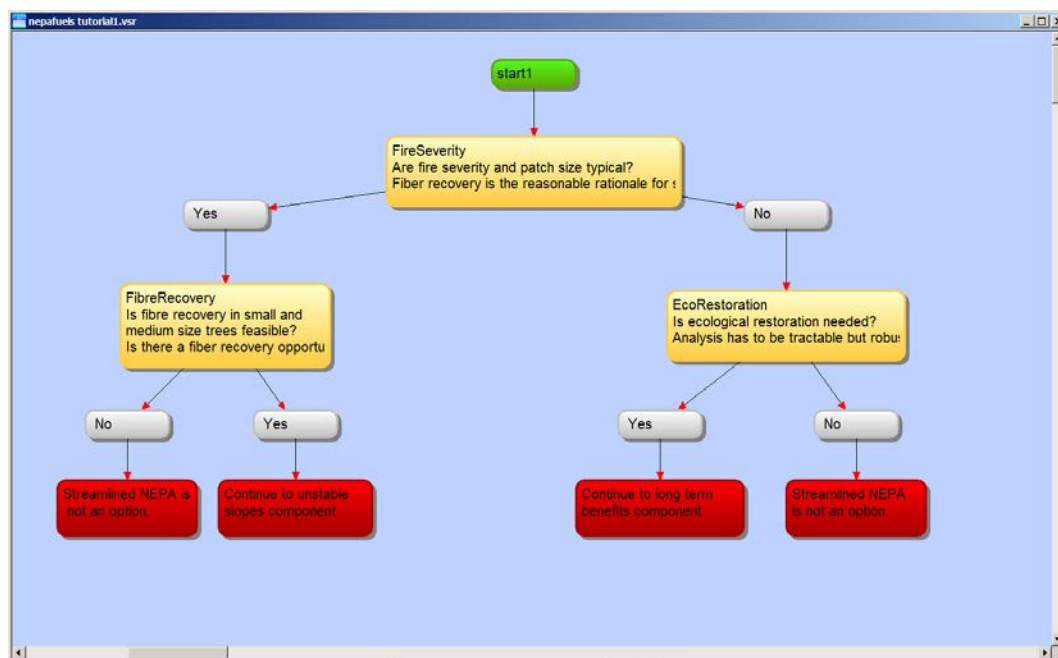


Figure 16. A truncated version of the decision tree.

At this point, you should save the decision tree, selecting Save on the File menu, and using a suitable name - say Nepafuels.

Now we are ready to generate the code and run the chart. You can invoke the Flex code generator in one of 2 ways - either by choosing the Run Code item from the Options menu (Figure 17),

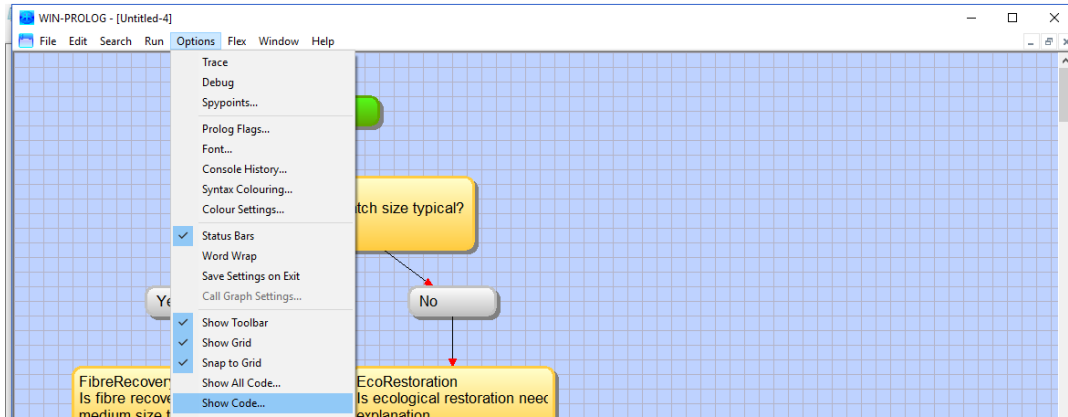


Figure 17. Selecting 'Show Code' from the Options menu.

or by simply right clicking anywhere in the canvas backdrop and choosing the item from the small pop-up menu.

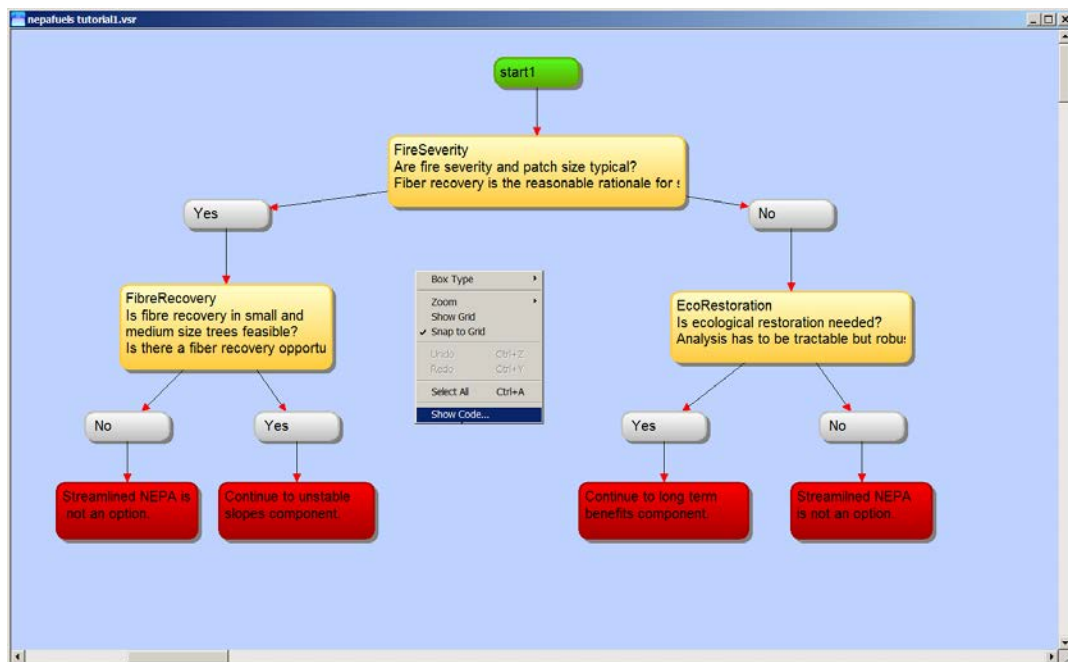


Figure 18. Right click on the canvas to get the popup dialog, and select 'Show Code.'

This will open the 'Generated code' dialog (Figure 19).

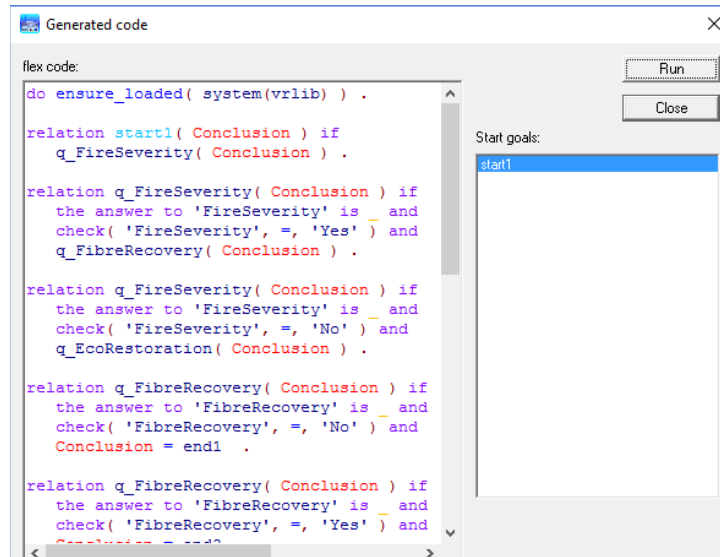


Figure 19. Generated code dialog window.

The generated code dialog contains the Flex KSL code which is implied and generated from your VisiRule chart. If you click on run, it will execute the associated KSL code and run the start goal -- this will lead to our first question (Figure 20).

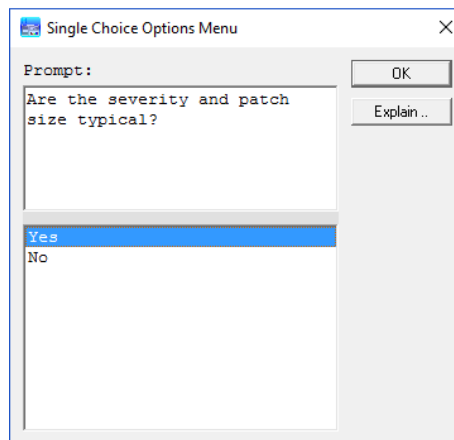


Figure 20. The VisiRule chart prompts for an answer to the first question.

If we select Yes, it will generate our second question (Figure 21).

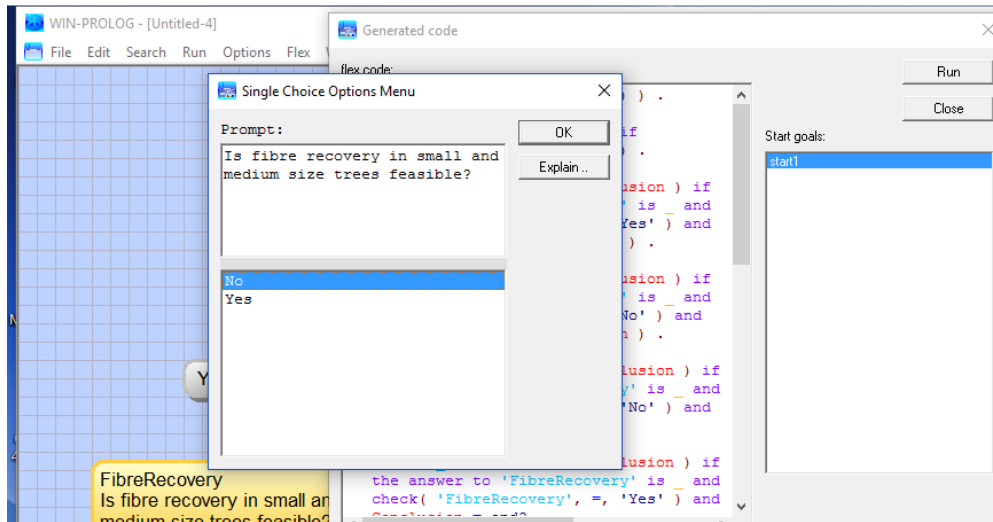


Figure 21. VisiRule chart prompts for answer to the second question.

If we select, Yes, we will reach an end node (Figure 22).

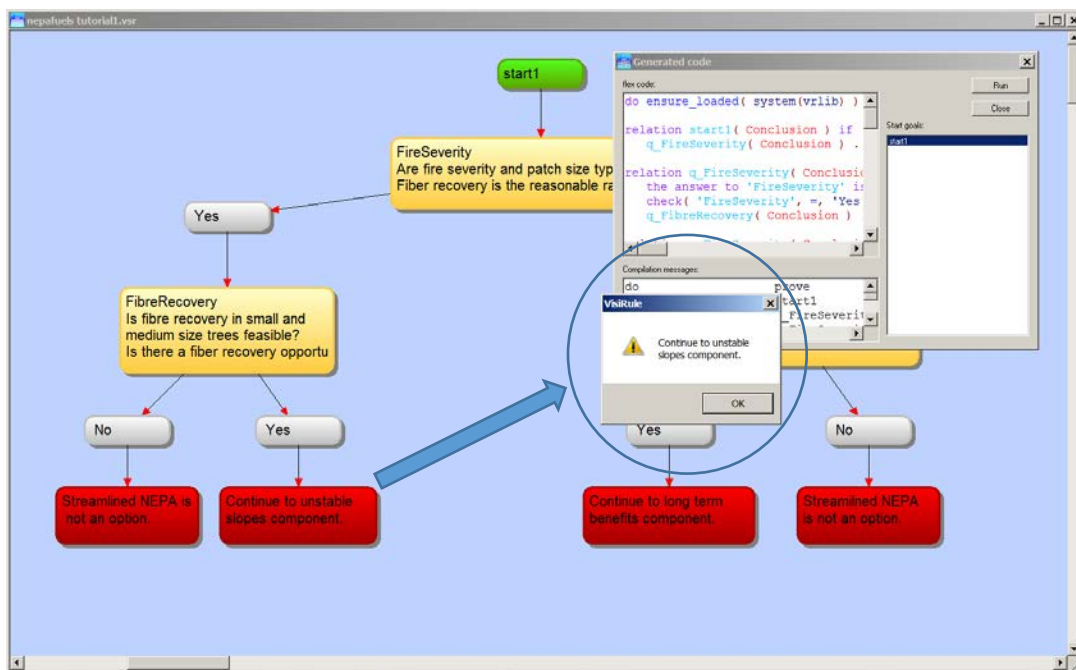


Figure 22. After answering Yes twice, VisiRule displays a solution (circled area).

If you click OK in the solution dialog (Figure 22), VisiRule will ask if you want to search for additional solutions. In our very simple example, there is only one possible solution, given that we answered Yes twice in this example, so you can simply answer No. However, more complex decision trees can, in fact, generate multiple alternative solutions.

If you want to explore the solutions further (e.g., different combinations of Yes and No), simply click on Run again in the generated code dialog window (Figure 19).

Wrapping up

In this tutorial, we have presented a very small and simple example of a VisiRule decision tree to introduce you to VisiRule, but VisiRule charts can handle far larger and much more complex types of decision problems. To continue learning the more advanced features of VisiRule, we suggest, as a start, downloading

- The VisiRule Tutorial (vsr_tut.pdf), which presents a number of examples of decision trees, and
- The VisiRule Reference (vsr_ref.pdf), which is a comprehensive reference for VisiRule.

Both of the above can be found at http://www.lpa.co.uk/dow_doc.htm, the documents download page of Logic Programming Associates, the makers of VisiRule. For the adventurous, the documents download page covers a wide variety of advanced products such as WIN-PROLOG, flex, and the WebFlex Toolkit.

In addition to the above resources, there is a VisiRule resources page at <http://www.visirule.co.uk/resources>, which has a collection of papers, videos, and several interactive web demos of VisiRule.