

TreeSnatcher Plus

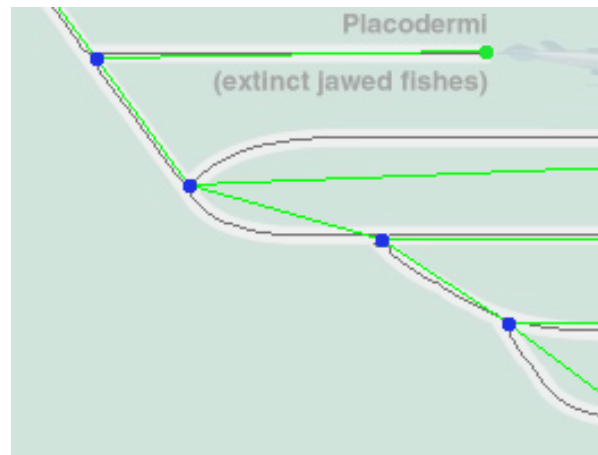
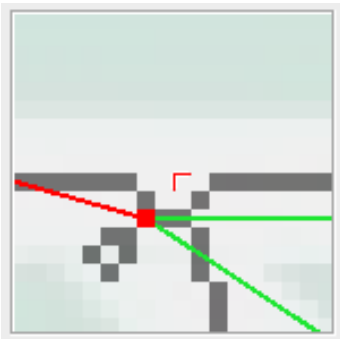


Tutorial 4 **Manipulating trees**

What we want

Please load the snapshot for tutorial 4.

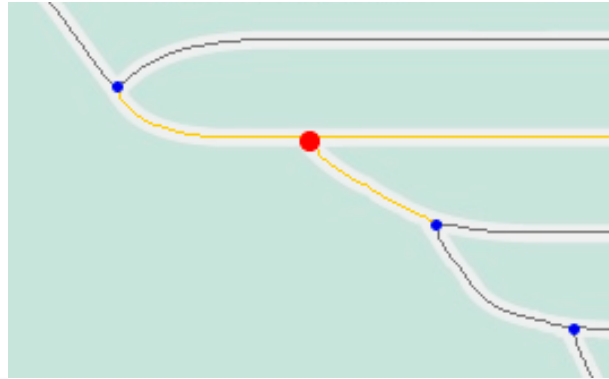
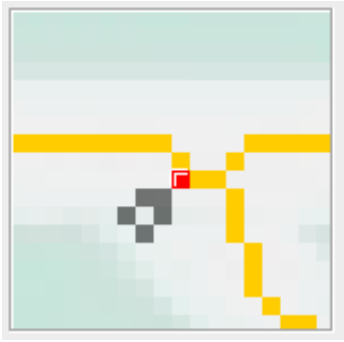
We would like to remove a part from the tree and append a new branch.



Locate the position shown in the magnifier:

Some foreground pixels are redundant but were ignored by the path finding algorithm. For the determination of branches and branch lengths TreeSnatcher Plus must identify the shortest paths between each pair of nodes.

Select **Check Linked Branches**. In this view, you can watch the foreground paths that emerge from the selected node. In the example, the paths around the suspicious node visible in the magnifier are short and without detours. If you encounter a path with detours, correct the pixels before you reuse **Calculate Branches**.



Return to the normal view by clicking **Show All Branches**.

Click **Whole Image**, then select **Remove Nodes in Selection** from the pop-up menu. This deletes all branches and erases all nodes within the selection box. As an intended side effect, the topology the program has constructed up to now is discarded. You need to place all nodes again later.

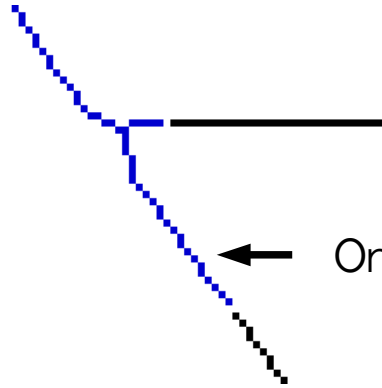
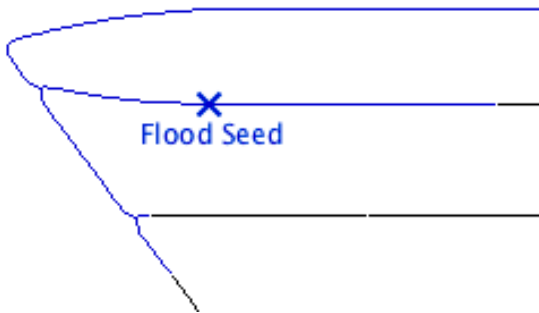
Select **Flood User Drawings**.

Right-click somewhere into the thinned tree, then choose **Flood Foreground** from the pop-up menu.

Mark the flood seed using **Mark TreeFlood Seed**.

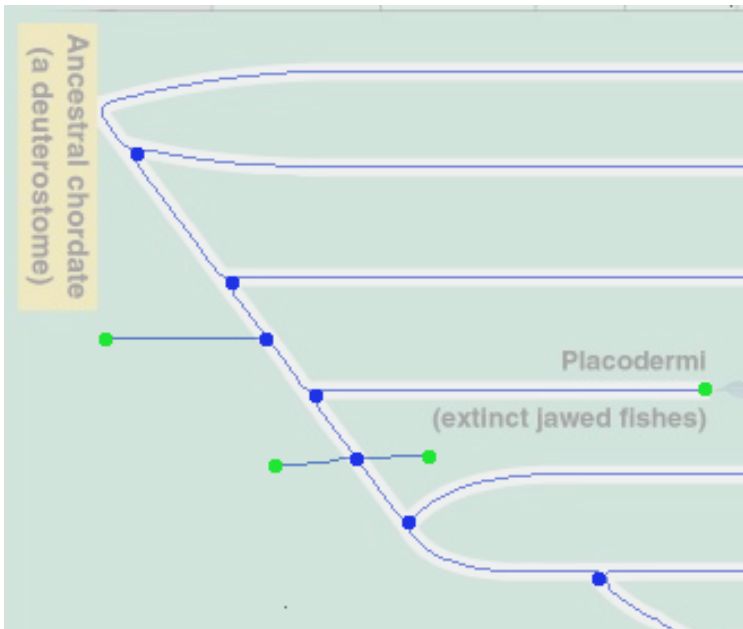
Now disconnect a path in the tree: Use the **Rubber** to delete one foreground pixel. By this you decouple a part of the tree from the rest. In subsequent steps the program will ignore the parts of the image that are not flooded now.

This could be the result:

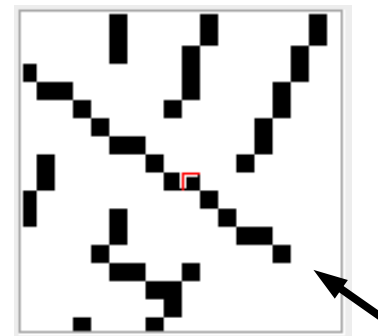
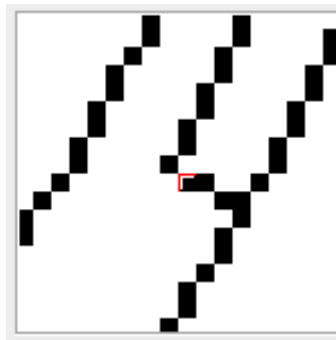


← Only this part belongs to the tree.

Unselect **Mask User Drawings** and select **Thin Freehand Drawings**.
Observe what happens if you draw into the image with the **Pencil**.
Select **Black Line** and append a line to the tree.



Select **Mask User Drawings** and draw a line in a crowded area: the line is isolated from other foreground elements by a white border.



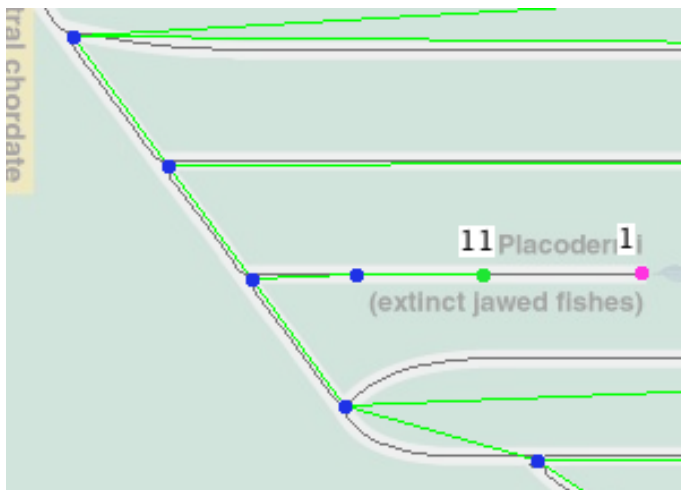
Use **Undo** to return to a previous state of work.

Click **Locate Nodes**.

Erase redundant nodes. Place at least one node at an improper position, e.g. an inner node in the middle of a path.

Click **Calculate Branches**.

Example:



Tip 11 gets connected to an inner node. But tip 1 is isolated which leads to an inconsistent topology. TreeSnatcher Plus displays an error message in the Newick view and highlights the tip in pink. Delete tip 1.

If the inner node in the middle of the path is yellow, the program has selected it as the node from where the Newick tree representation is constructed.

Why should the program select this node?

It is the only inner node in the image with exactly two emerging branches.

TreeSnatcher Plus assumes that such a node serves as the tree origin.

You can declare any inner node as origin with **Use Node as Origin**.

How a TreeSnatcher Plus **Snapshot** is built:

If you save the current state of work as a Snapshot, TreeSnatcher Plus uses the specified file name, i.e. „snap“, and stores the following files on harddisk:

snap.ser	All internal data structures and the tree topology
snap_b.png	Binarized image (if there)
snap_c.png	Current state of the image
snap_o.png	Source image

You need all of them in order to restore your work state.

On the TreeSnatcher Plus tutorials website you will find a selection of snapshots. Each snapshot comes as a ZIP-file. You can unzip all files into one directory.