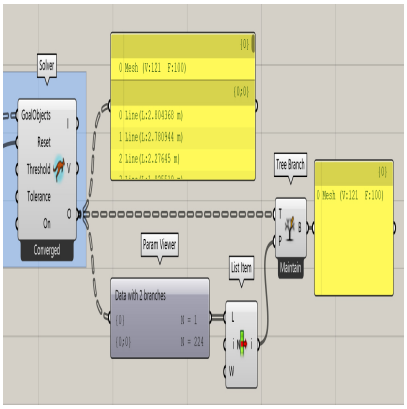


Rhino3D Grasshopper: Handle Paths with List / Cull

Description

You can extract Grasshopper *Data Tree Paths* just like simple *List Items*. It's slightly more work though – *Param Viewer* and *Tree Branch* are key.

For a good introduction on Grasshopper's Data Trees read this page from TU Delft.

New to Grasshopper? I suggest you read this article in the first place.

Need more learning resources? Check this out.

Disclaimer

As you will see in my screenshots most components carry a title above them displaying their name. This is because I use a plugin called *Bifocals*.

Also you'll see my Grasshopper tab titles abbreviated because there's not enough horizontal space. I will mention the titles in full in my script though.

As with everything Grasshopper, the technique I show you is no way my own discovery. I basically got the idea from Arturo Tedeschi who mentioned it in a webinar.

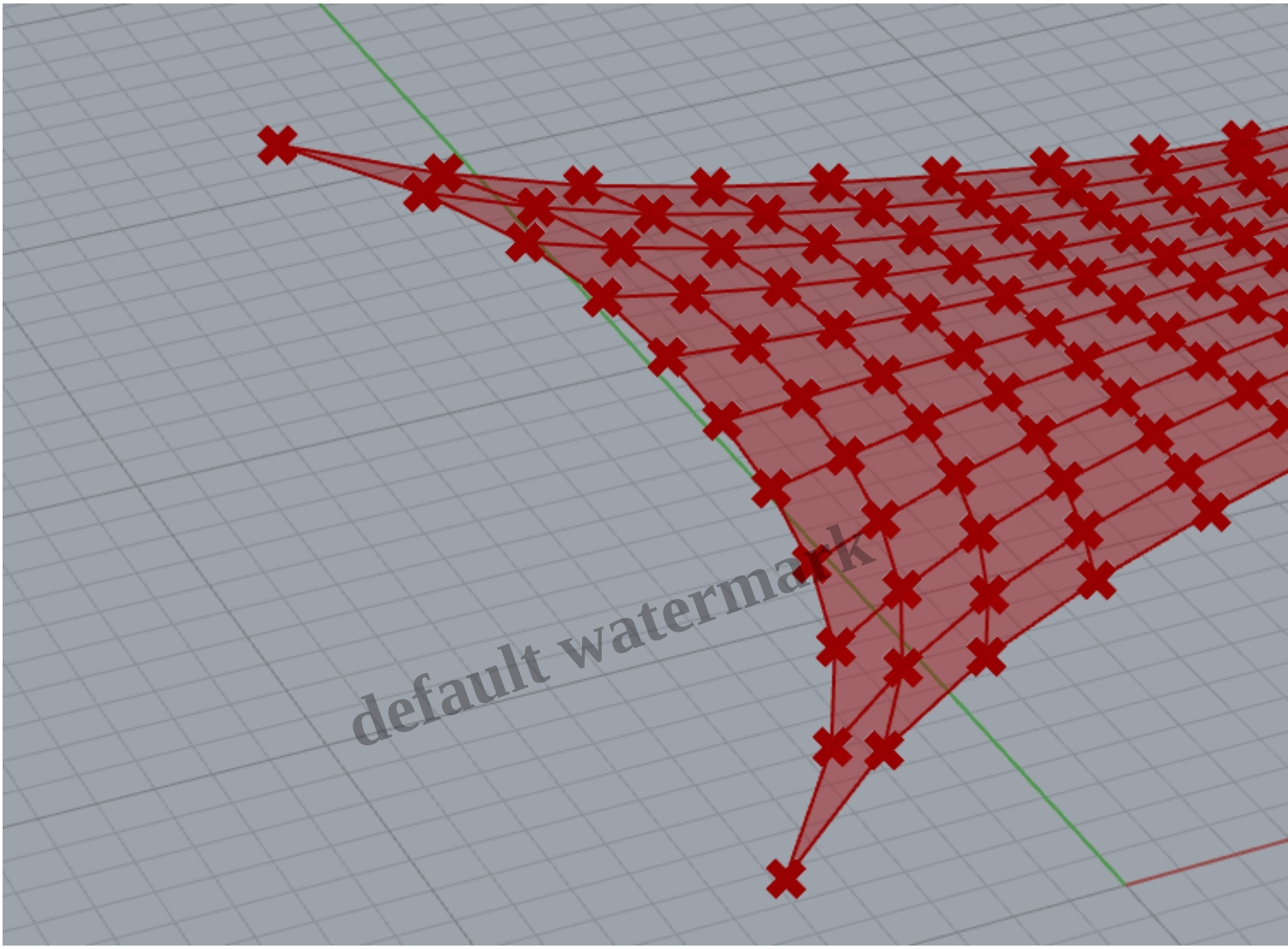
Let's Start

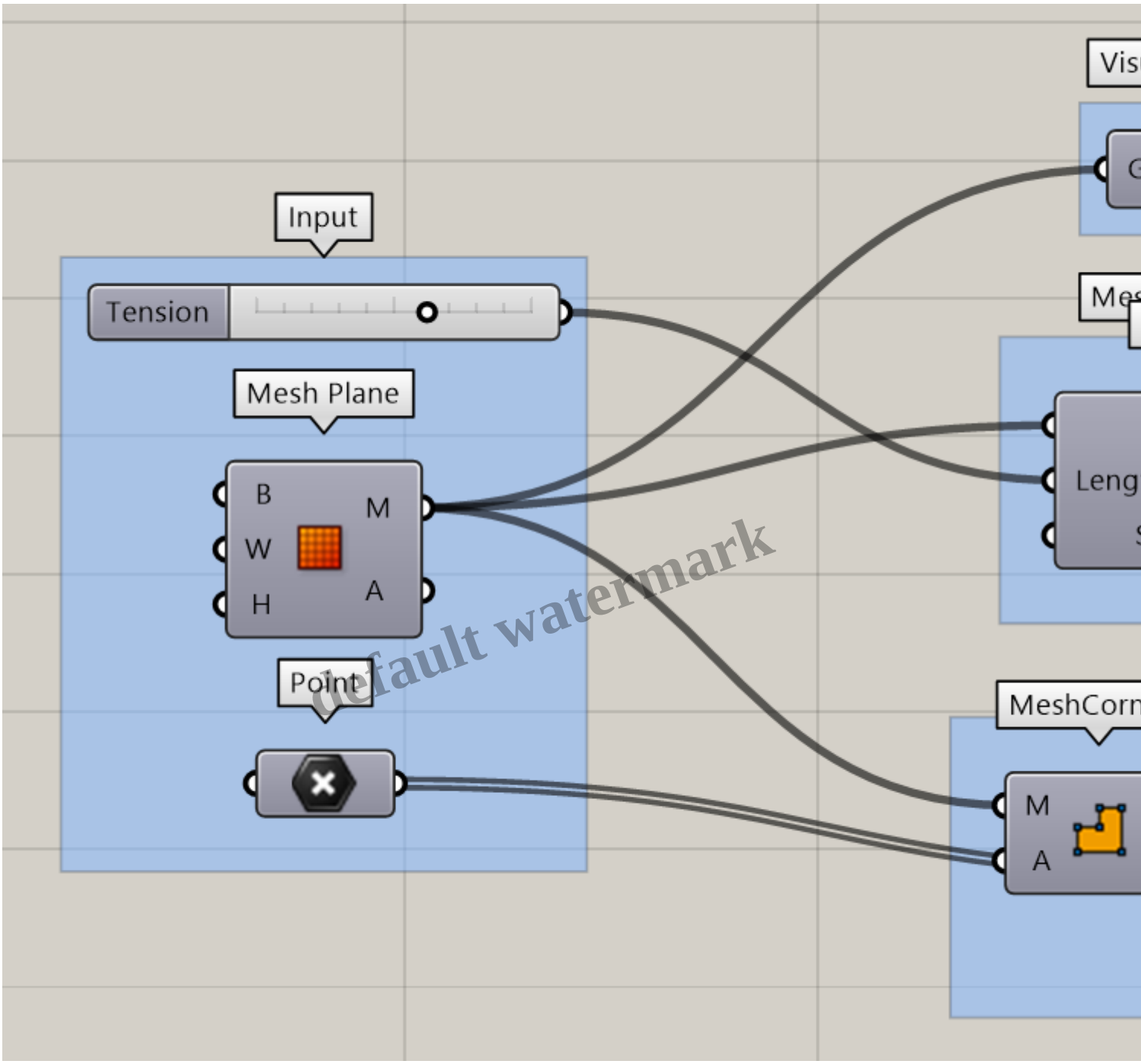
As shown in this article you can pull paths using *Tree Branch* – which seems simple enough.

Of course, this works only with a fixed tree structure because you have to feed specific path names into the *Tree Branch* input.

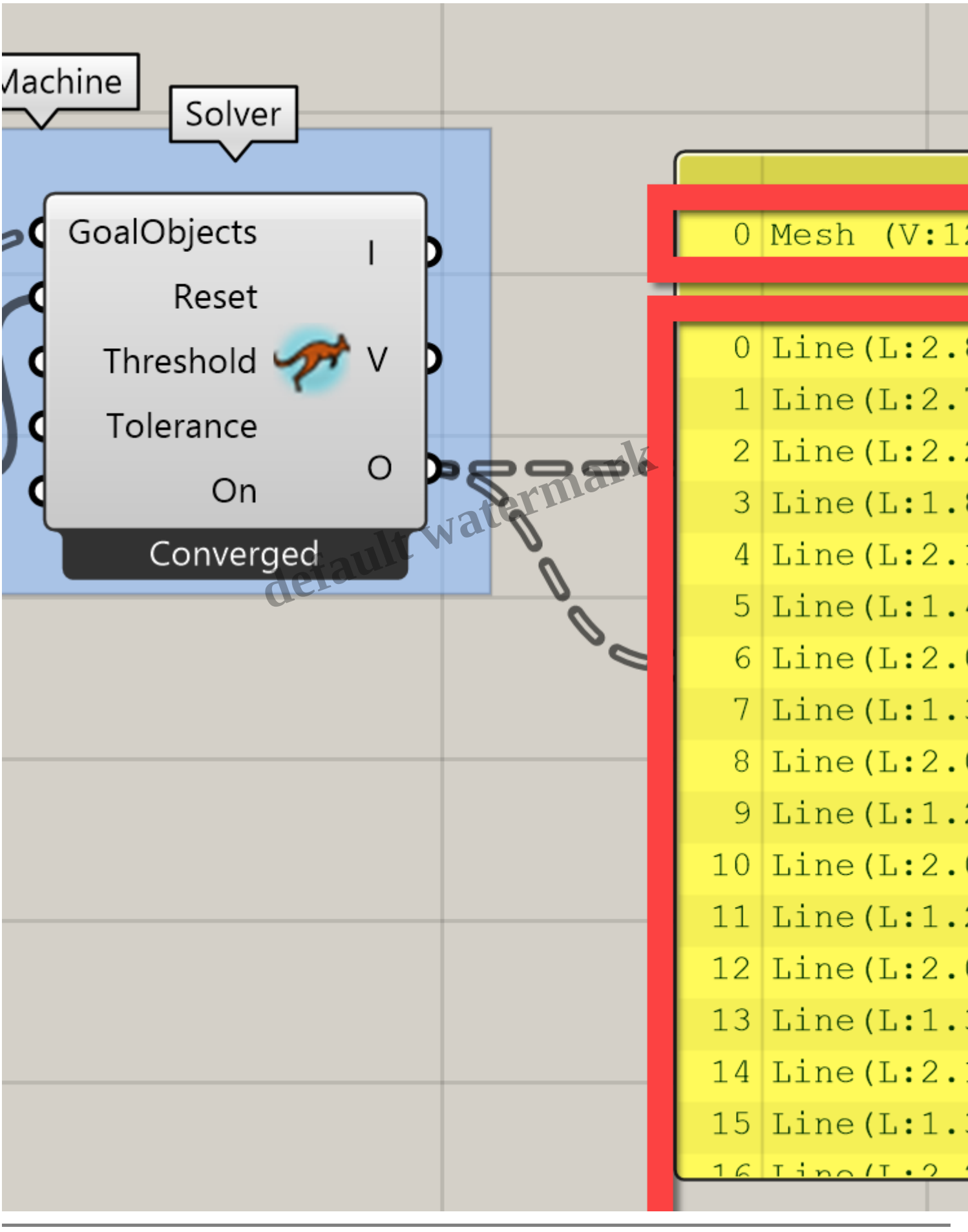
If you need to extract or hide paths with a given *index* – like you do with ordinary list items – you have to put some more components together to get the job done.

Let’s look at a situation that comes up with *Kangaroo:*





Let’s say we want to retrieve the resulting mesh for further work. A panel shows that we have mixed output:

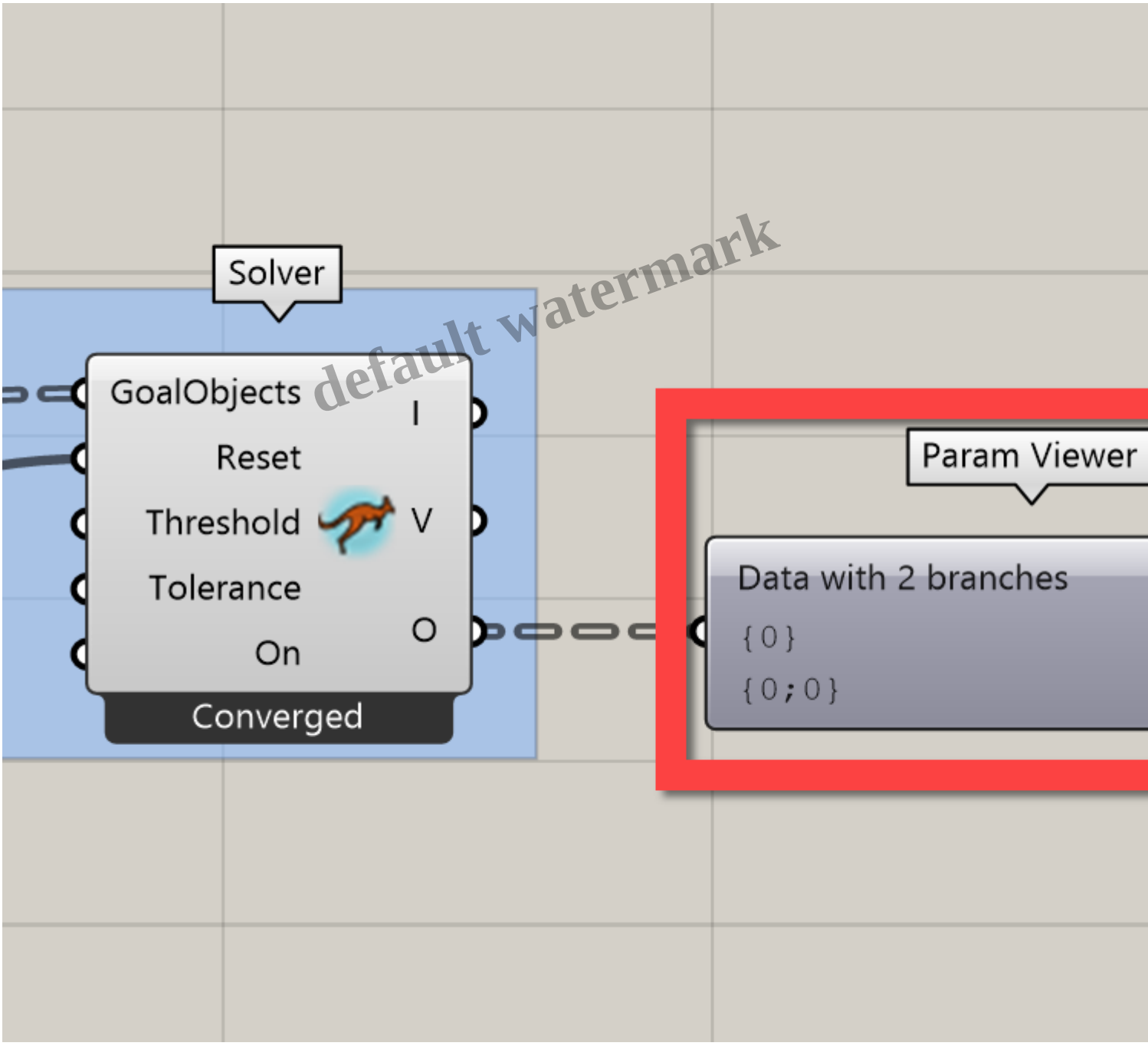


A *Param Viewer* component tells us exactly how many paths and items we've got here. OK – but how do we get hold of the first path {0}?

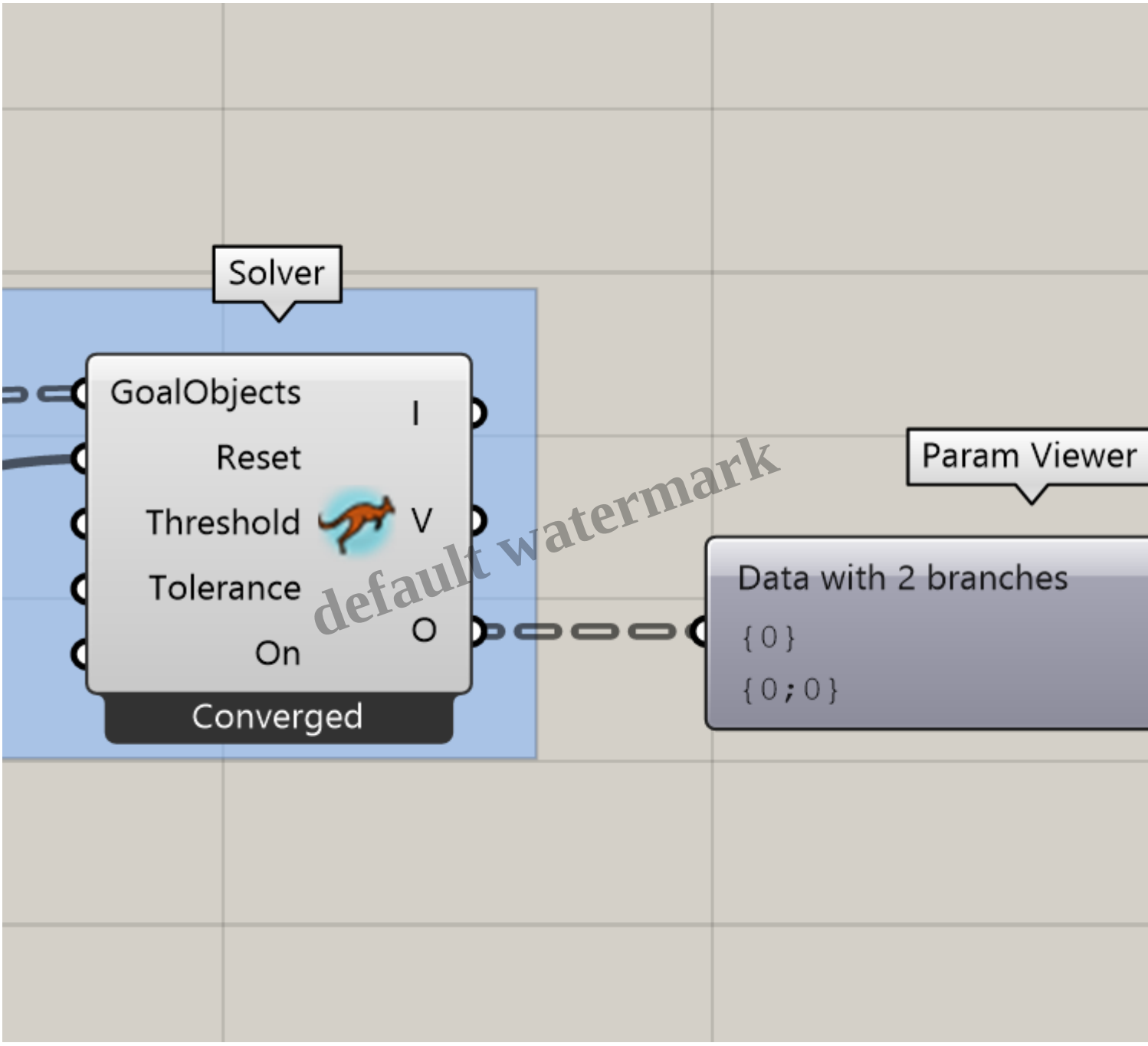
Again: A *Tree Branch* fed with a panelled *{0}* input would do – but this time we want to use a *List Item* component.

But of course, *List* and *Cull* components are made for lists, not for trees. So we have to use a kind of trick.

As you see a *Param Viewer* has been connected to our mixed output:

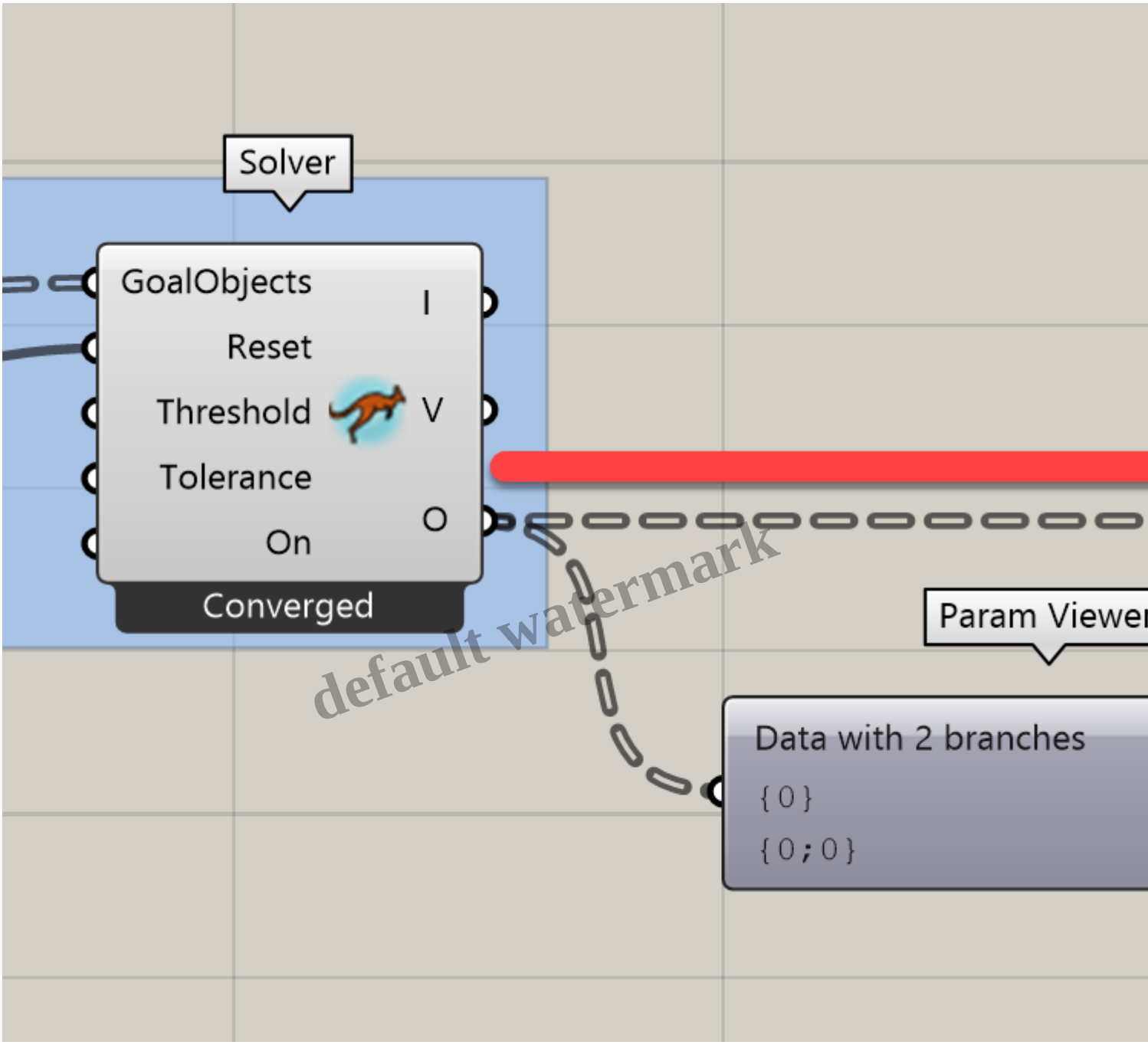


Since our plan is to extract the mesh – which is in *{0}* – get a *List Item* and connect it to *Param Viewer's* output:



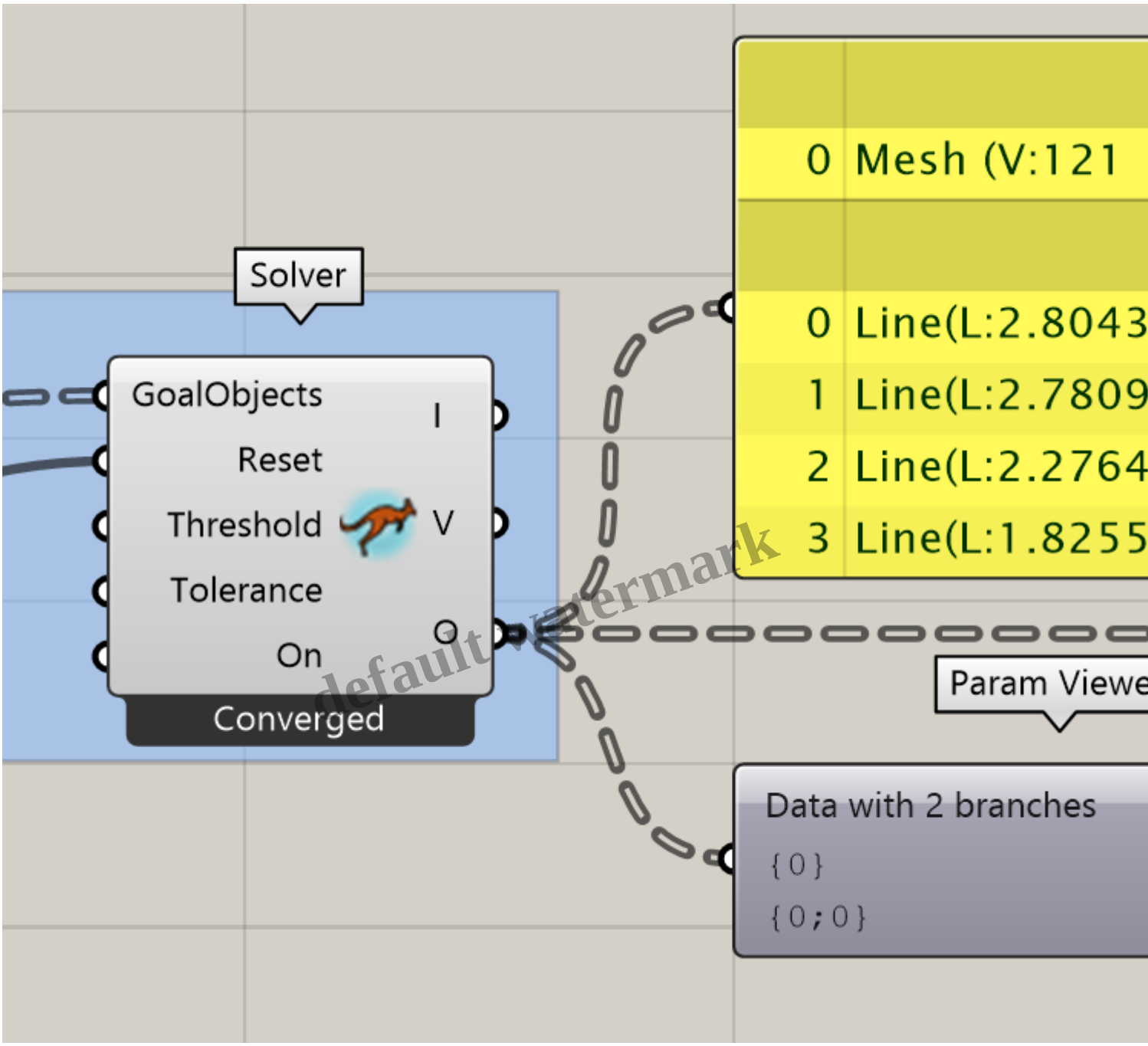
List Item's default output is *0,* meaning that – if our plan succeeds – it will pull our mesh. That's fine, but there's some more to do.

Pick a *Tree Branch* component and connect it to *Kangaroo Solver's* output:



Tree Branch wants to know which path it's supposed to extract for you *(P?)* – this is the place where we put in a path *name,* described in this article.

In this revised and more universal approach we connect *P* with our *List Item's* output *i:*



That's it – Using *Param Viewer* before and *Tree Branch* after *List Item* basically helped to extract a data tree path just like a normal list item.

I haven't tested it altogether – but it seems that this way you can apply all of Grasshopper's *list* an *cull* items on paths.

For a good introduction on Grasshopper's Data Trees read this page from TU Delft.

New to Grasshopper? I suggest you read this article in the first place.

Need more learning resources? Check this out.

Category

1. Rhino/Grasshoppper

Tags

1. Data Tree
2. Parametric Modeling

Date Created
November 2019
Author
hsondermanncom