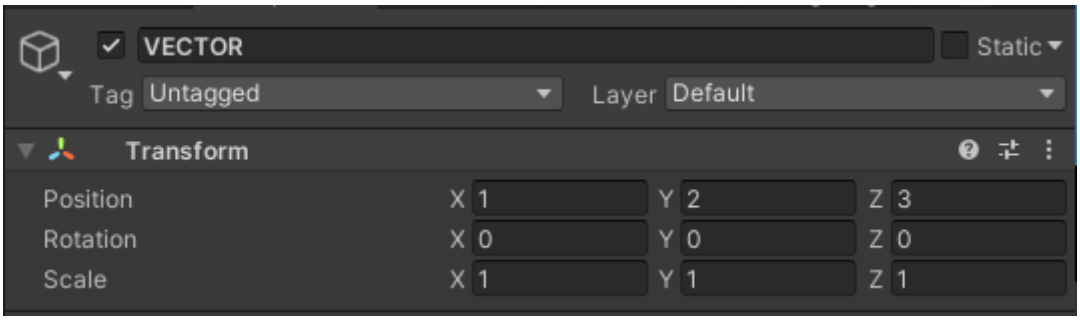
Chapter 1: Basic Operations

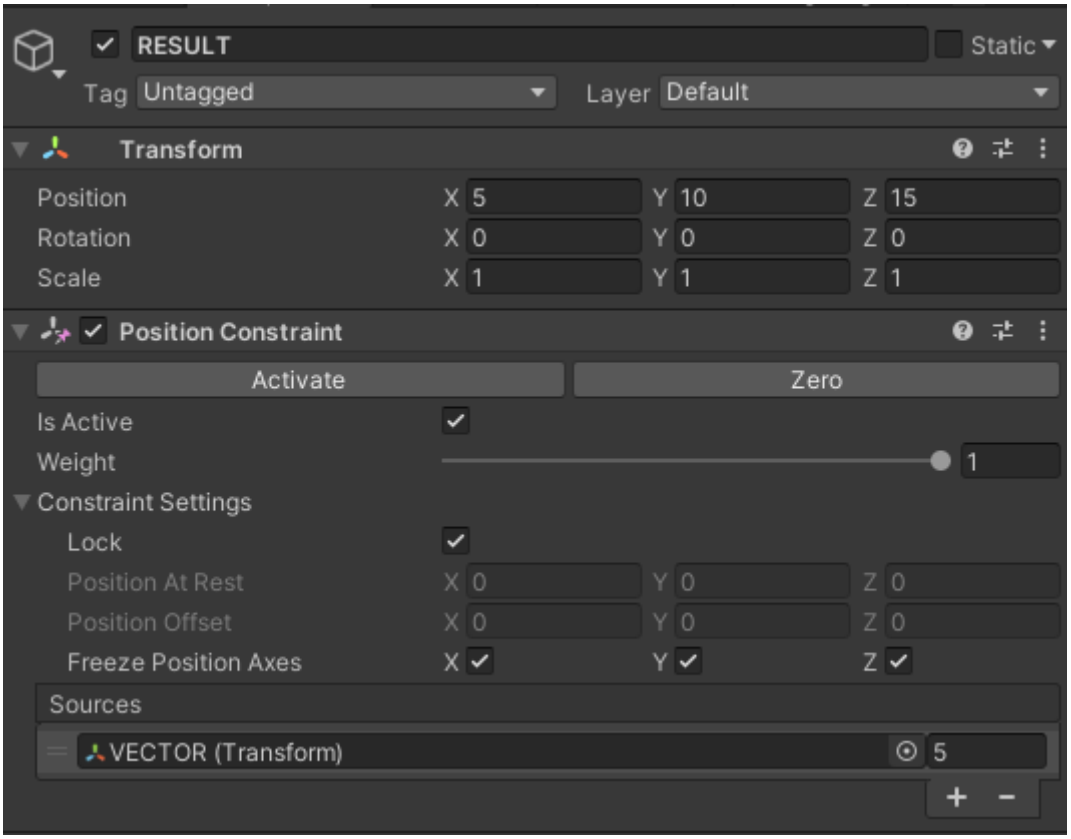
- Multiplying a Vector by a Scalar
- Inverting a Vector
- Adding two Vectors
- Subtracting two Vectors
- Normalizing a Vector
- What about dividing a vector by a scalar?
- Projecting a Vector (in World-Space)

Multiplying a Vector by a Scalar

To multiply a vector by a scalar we can use the fact that when a position constraint is given a single target, the source weight wont be normalized.

This means we can simply use a position constraint, with a single target (which contains the vector as its position) and give it a source weight of the scalar. For example this could look like this:





Here we multiply the vector (1,2,3) by 5 to get (5,10,15).

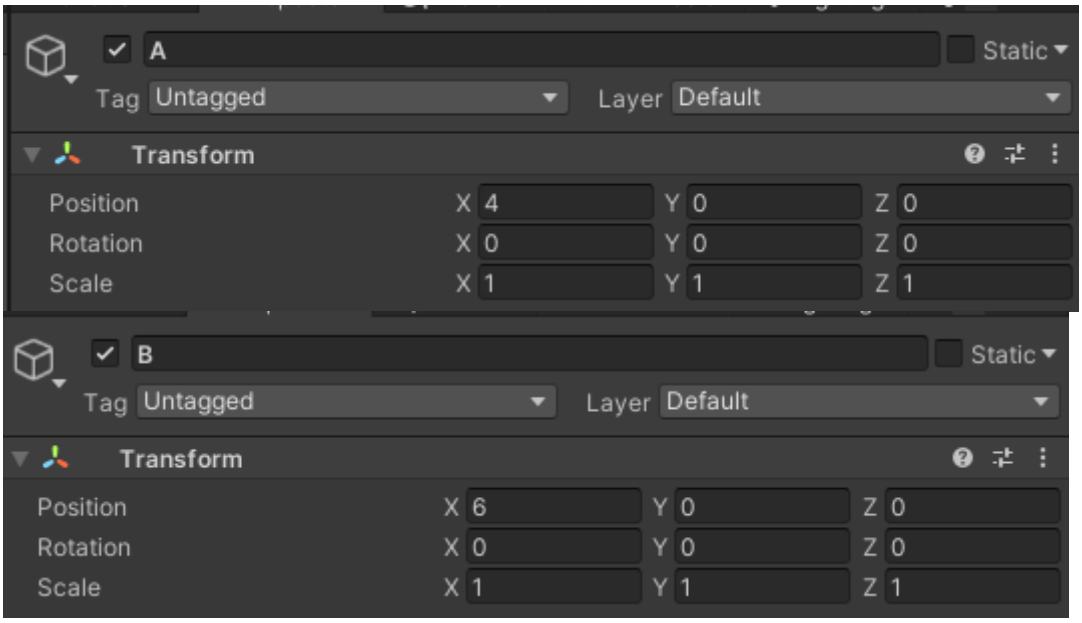
Inverting a Vector

This can be done with the same method as Multiplying a Vector by a Scalar by simply using -1 as the scalar.

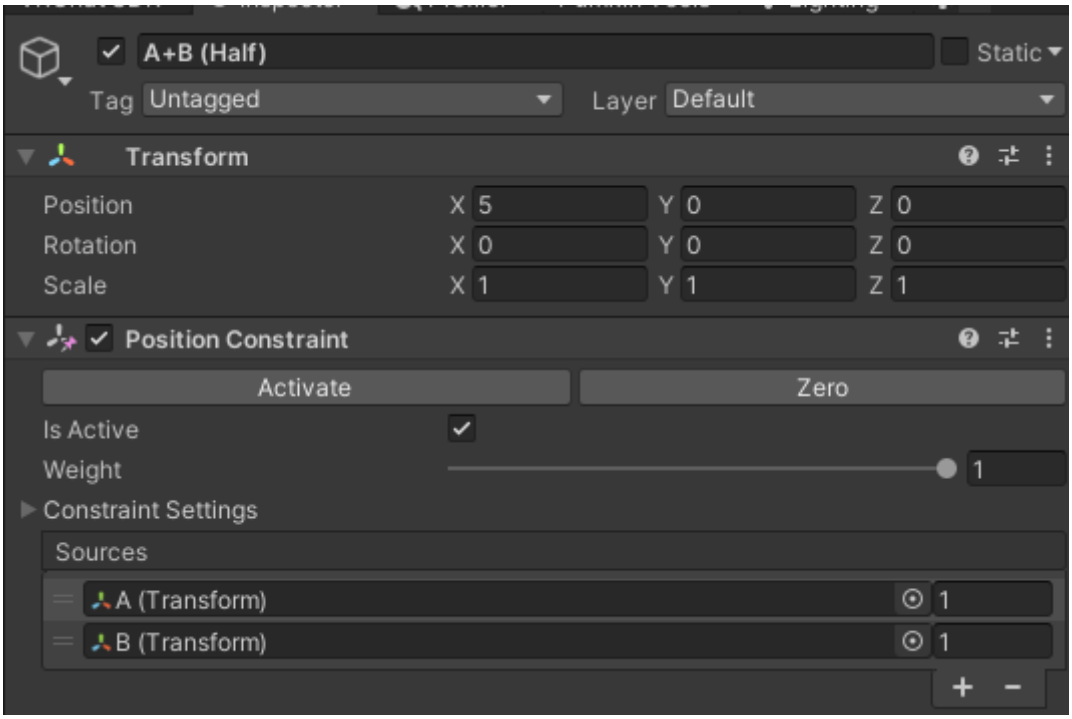
Adding two Vectors

To add two vectors we have to get a little creative. I wont explain why this trick works mathematically, but to add two numbers, we can average them and then multiply that by two.

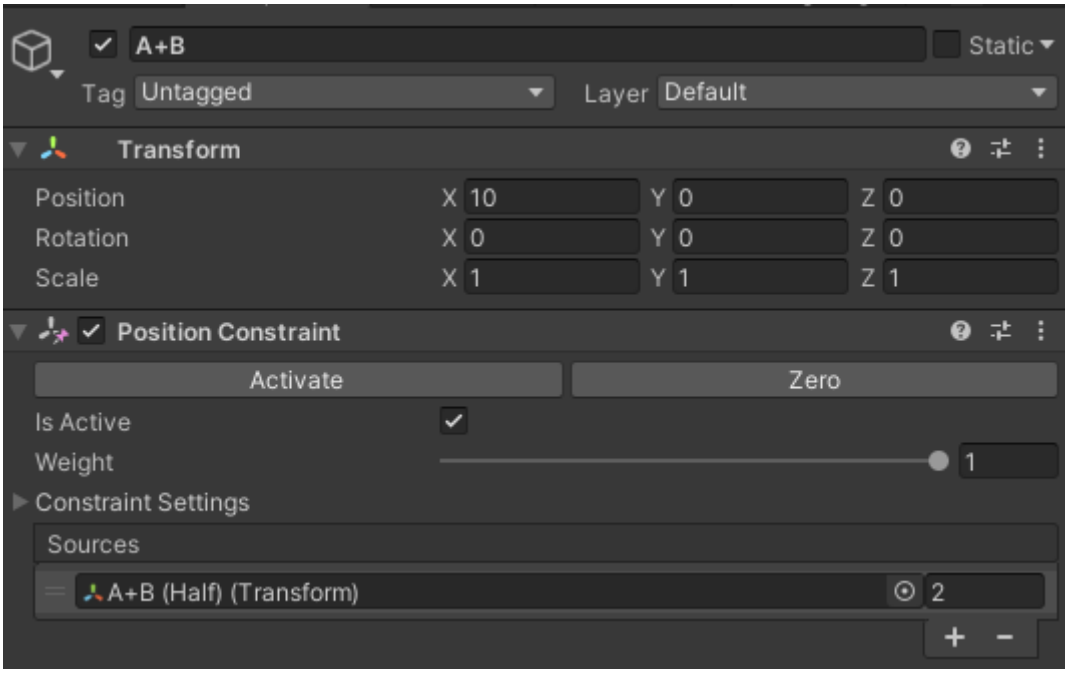
Since PositionConstraints can blend i.e. average between two transforms i.e. vectors and we already know how to multiply a vector by a scalar we can do this in two steps:



Step 1: Average



Step 2: Multiply by 2



You might wonder why we cant do this in one step, by simply using 2 as the weight for both A and B in Step 1, however remember that the source weights get normalized when you have more than one weight. So unfortunately we have to do this in two steps.

Subtracting two Vectors

To subtract two vectors you can use the same method as Adding two Vectors but inverting either vector first by inverting it first.

Which one you have to invert, depends on whether you want to:

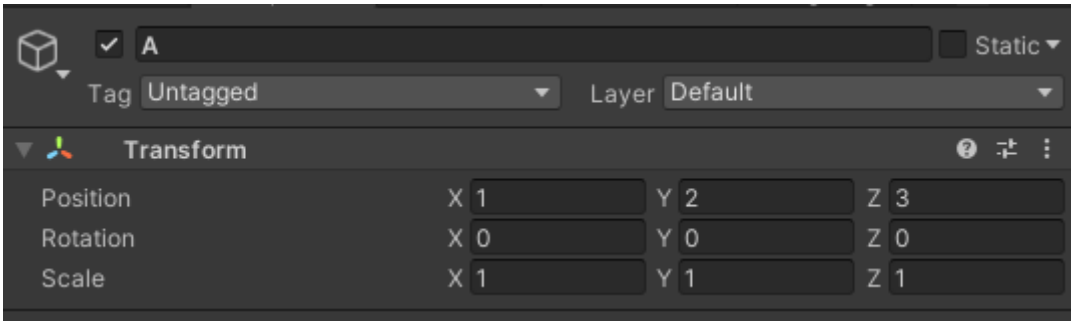
Subtract B from A (A - B) => Invert B

or

Subtract A from B (B - A) => Invert A

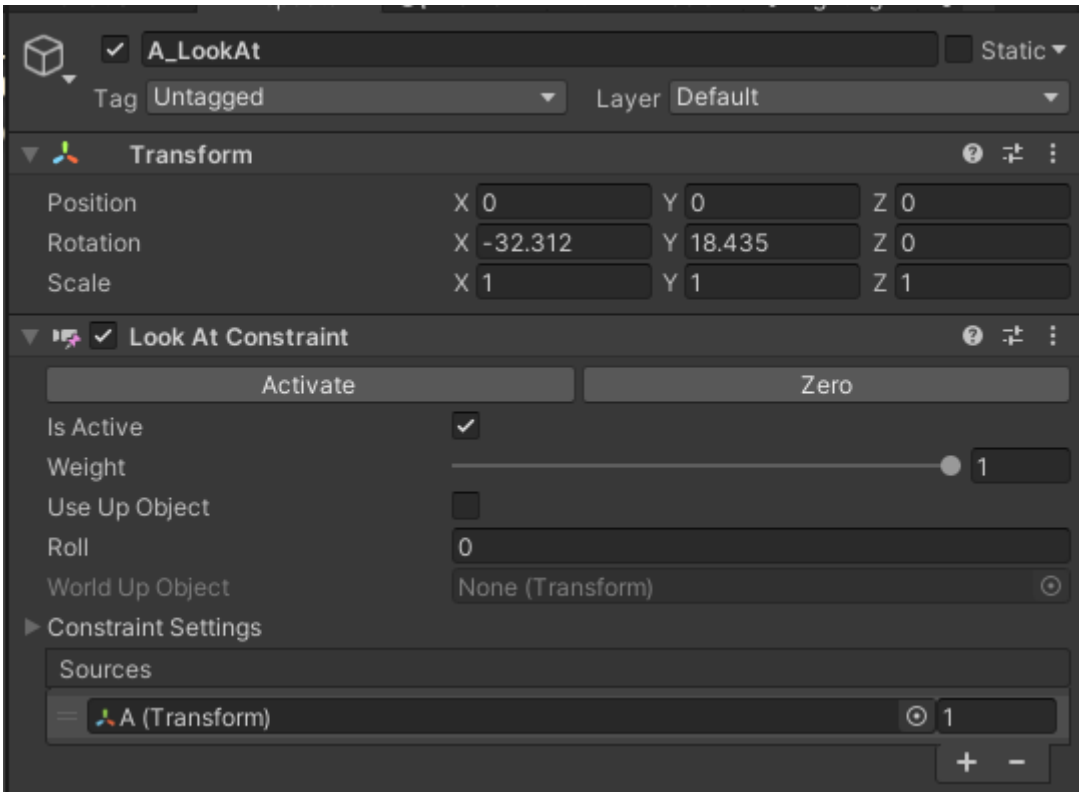
Normalizing a Vector

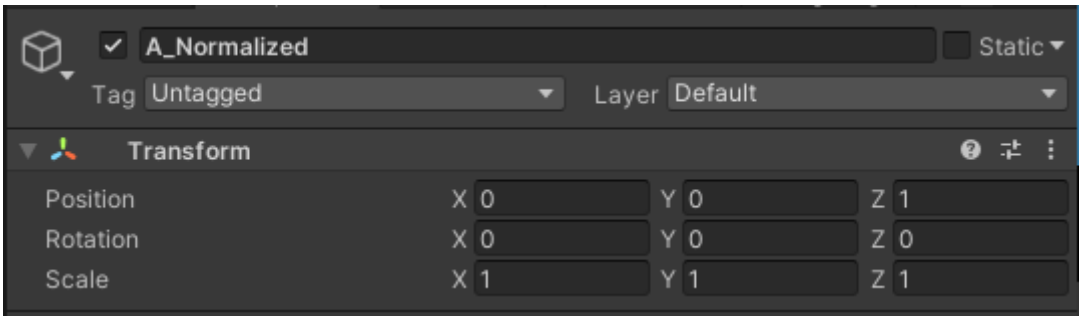
To normalize a vector we can use the LookAtConstraint. Make it look at the vector i.e. transform that you want to normalize. Give it a child with an offset of +1 on the Z-Axis. You have now normalized a vector.



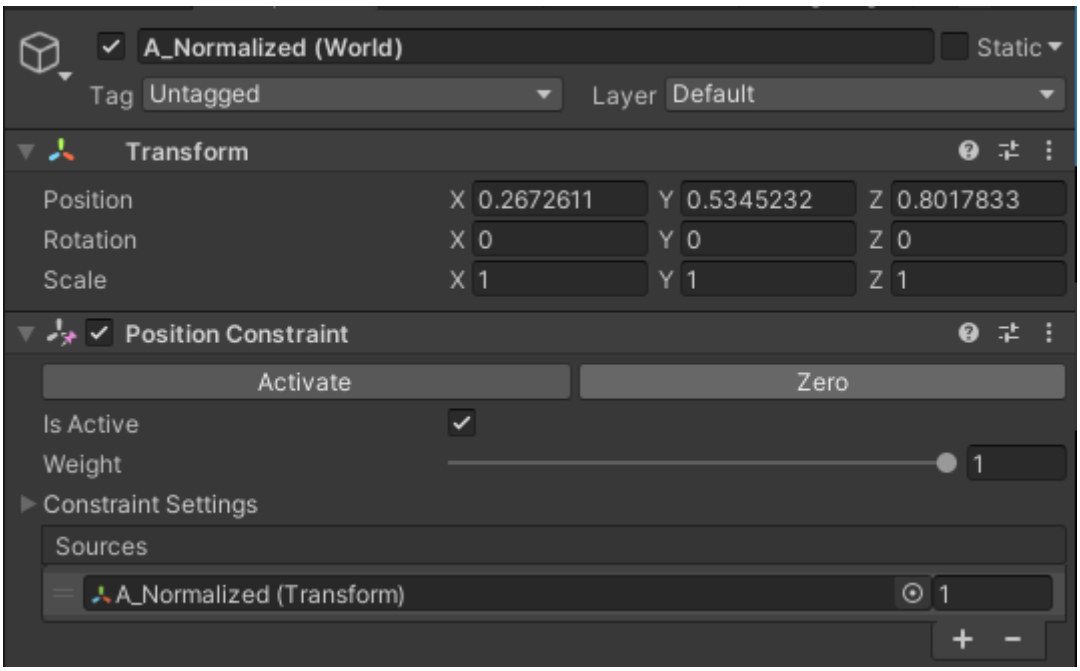
Hierarchy:







As previously mentioned "A_Normalized" shows its position in local-space. However whe can show its world-space position by using another position constraint. This is **not** nessecary for the math to work, but can be nice for debugging and visualization.



What about dividing a vector by a scalar?

Unfortunately I currently do not know of any method to find the multiplicative inverse of an arbitrary number.

If you want to divide a vector by a constant scalar however, you can simply just use the same method as Multiplying a Vector by a Scalar and use the multiplicative inverse (1/x).

Projecting a Vector (in World-Space)

To project a vector in world-space, you can simply use a position constraint, and only have one (to project the vector onto an basis axis) or two (to project the vector onto a basis plane) of the "Freeze Position Axes" toggles turned on.