

# Assembly

The nodes on this page report on SimLab's **VR assembly** system. They are **states**: each one keeps an eye on a single part and continuously reports a simple true/false answer about where that part is in the assembly — for example “is it this part's turn to be put on?” or “is this part fully assembled?” A state is not an action you run once and finish; it is a live condition the scene keeps checking the whole time it is playing.

In a VR assembly, parts go together in a set order, and each part has a state of its own — think of a screw: all the way *out* (taken apart), *seated in place but not tightened* (partway together), or *tightened* (fully together). The nodes below let your scene react to where a part is in that process — whether it is this part's turn to be added or removed, and whether a part is completely assembled or completely apart.

Every one of these nodes gives you the same three things to work with:

- **OnTrue** — a trigger that fires the moment the answer becomes true.
- **OnFalse** — a trigger that fires the moment the answer becomes false.
- **Output Boolean** — the current true/false value, ready to read at any time or to send into a **Branch** node.

The two triggers fire only when the answer *changes* — not over and over while it stays the same. These nodes only watch and report; they never move, assemble, or change anything in your scene.

## What's on this page

- **Node Can Assemble** — is it this part's turn to be put on next?
- **Node Can Disassemble** — is it this part's turn to be taken off next?
- **Node Fully Assembled** — is this part completely assembled (in place and secured)?
- **Node Fully Disassembled** — is this part completely taken apart (all the way out)?

Looking for other state nodes — checking whether two objects overlap, or whether the viewer is overlapping an object? See the **States** page.

# Node Can Assemble

Reports whether a part in a VR assembly is currently allowed to be put on next — that is, whether it is this part’s turn in the assembly order.

## What it does

In a VR assembly, every part has to go on in a set order, and some parts can only be added once the parts before them are already in place. This node watches one part and keeps answering a simple yes/no question: is it this part’s turn yet? The answer is `true` as long as everything that must come before this part has already been assembled, so it is now ready to be put on. While some earlier part is still missing, the answer is `false`.

It only watches and reports — it never assembles the part or changes anything in the scene. You connect what should happen on the “ready” side and the “not yet” side, and read the current answer whenever you like.

## Inputs

Port	Type	What to connect
<b>SceneNode</b>	Scene node	The part you want to check — for example a <code>Bolt_01</code> object from your assembly. The node reports whether this part is currently allowed to be assembled next.

## Outputs

Port	Type	What you get
<b>OnTrue</b>	Trigger	Fires once at the moment the part becomes ready to assemble — that is, when the last part that had to come before it is finally in place. Wire this to whatever should happen when it is this part’s turn, such as highlighting it or showing a hint.

Port	Type	What you get
<b>OnFalse</b>	Trigger	Fires once at the moment the part is no longer ready — for example if an earlier part is removed again, so it stops being this part's turn. Wire this to whatever should happen when the part is not yet ready.
<b>Output Boolean</b>	True / false	The current answer, ready to read at any time: <code>true</code> while the part is allowed to be assembled next, <code>false</code> while it is not yet its turn. Send this into a Branch node, or use it anywhere you need the current yes/no value.

## Example

<b>SceneNode</b> input	<code>Bolt_01</code>
<b>Output Boolean</b> output	<code>true</code> — the <code>Housing</code> and <code>Gear</code> that come before <code>Bolt_01</code> are already assembled, so it is now the bolt's turn
<b>OnTrue</b> output	Fires the instant <code>Gear</code> is seated — the bolt is now ready, so you could highlight it to guide the trainee to fit it next

## Tips

- Use the **Output Boolean** when you just need the current answer somewhere else (for example in a Branch); use **OnTrue** / **OnFalse** when you want something to happen the moment a part becomes ready or stops being ready.
- The triggers fire only on a change. If the part is already ready when your scene starts, **OnTrue** won't fire until the answer leaves and returns to true.
- This node only reports readiness — it never assembles the part. Combine it with your assembly steps to guide the trainee through the parts in the right order.

## Node Can Disassemble

Reports whether a part is currently allowed to be taken off next in a VR assembly — that is, whether every part that has to come off before it has already been removed, so it's now this part's turn.

# What it does

In a VR assembly, parts come apart in a set order — the pieces on top have to be removed before the ones underneath. This node watches one part and answers a simple live question: “Is it this part’s turn to be taken off?” The answer is true when all the parts that must be removed ahead of it are already off, and false while something on top of it is still attached.

This node only watches and reports — it never takes the part off or changes anything in the scene. The answer updates on its own as the assembly is taken apart, so you can use it to know exactly when a part becomes ready to remove.

## Inputs

Port	Type	What to connect
<b>SceneNode</b>	Scene node	The part you want to watch — for example a bolt or a panel. The node reports whether this part is currently allowed to be taken off next.

## Outputs

Port	Type	What you get
<b>OnTrue</b>	Trigger	Fires the moment the part becomes ready to be taken off — the instant it becomes this part’s turn. Wire it to whatever should happen then, such as highlighting the part or showing a “remove this” hint. It fires once at that moment, not continuously.
<b>OnFalse</b>	Trigger	Fires the moment the part is no longer the one allowed to come off next. Wire it to whatever should happen then, such as turning off a highlight. It fires once at that moment.
<b>Output Boolean</b>	True / false	The current answer as a true/false value — true while it’s this part’s turn to be removed, false otherwise. You can read it at any time or send it into a Branch node.

# Example

<b>SceneNode</b> input	<code>Bolt_01</code>
<b>OnTrue</b> output	fires when the cover above <code>Bolt_01</code> has been removed and it's now the bolt's turn to come off
<b>OnFalse</b> output	fires if <code>Bolt_01</code> is no longer the next part allowed to be removed
<b>Output Boolean</b> output	<code>true</code> once the bolt is ready to be taken off

## Tips

- Use **OnTrue** to guide the user — for example, light up the next part the moment it becomes removable so they always know what to take off next.
- This node reports readiness only; it does not take the part off. Pair it with your normal grab-and-remove interaction to act on the part itself.
- Send the **Output Boolean** into a Branch node when you want a check elsewhere in your logic to react to whether a part is ready to be removed.

# Node Fully Assembled

Watches one object and reports whether it is completely assembled — fully in place *and* secured.

## What it does

This node keeps an eye on the object you connect and answers a simple yes/no question while your scene runs: is this object fully assembled? In SimLab's assembly system a part moves through three stages — think of a screw. With the screw all the way out, the part is disassembled. Once it is dropped into place but not yet tightened, it is only partially assembled. When it is finally tightened down, it is fully assembled. This node reports **true** only at that last stage, when the object is completely in place and secured. As long as the object is still loose, only seated, or not yet positioned, the answer stays **false**.

It only watches and reports — it never moves, tightens, or changes the object in any way. You use its answer to drive the rest of your scene: play a confirmation sound when a part is fully assembled, show a warning while it still isn't, or feed the current answer into a Branch node.

## Inputs

Port	Type	What to connect
<b>SceneNode</b>	Scene node	The object you want to watch — the part whose assembly you care about, such as <code>Bolt_01</code> . The node reports on this object's own assembly state.

## Outputs

Port	Type	What you get
<b>OnTrue</b>	Trigger	Fires once the moment the object becomes fully assembled. Wire it to whatever should happen when the part is finished — a success chime, a checkmark, the next step in your guide.
<b>OnFalse</b>	Trigger	Fires once the moment the object stops being fully assembled — for example if it is loosened or taken back apart. Wire it to whatever should happen when the part is no longer complete.
<b>Output Boolean</b>	True / false	The current answer — <code>true</code> while the object is fully assembled, <code>false</code> otherwise. You can read it at any time or send it straight into a Branch node.

## Example

<b>SceneNode</b> input	<code>Bolt_01</code>
<b>OnTrue</b> output	Fires when <code>Bolt_01</code> is tightened all the way down — play a confirmation sound and mark the step complete.
<b>OnFalse</b> output	Fires if <code>Bolt_01</code> is later loosened or removed.
<b>Output Boolean</b> output	<code>true</code> once the bolt is fully assembled, <code>false</code> while it is still loose or only seated.

## Tips

- A part that is only dropped into place but not tightened still counts as **false** here — this node reports true only when the part is both in place and secured.

- **OnTrue** and **OnFalse** each fire just once, right at the moment the answer changes. If you instead need to react to the answer continuously, read **Output Boolean** — it always holds the current value.

# Node Fully Disassembled

Reports whether the object you connect is completely taken apart — all the way out of place, not just loosened.

## What it does

Connect one object and this node keeps a live yes/no answer to the question “is this part fully disassembled?” In an assembly scene a part can be in three states — think of a screw: all the way out (**fully disassembled**), dropped into place but not tightened (**partially assembled**), or tightened down (**fully assembled**). The answer here is **true** only when the object is all the way out, and **false** while it is partially assembled or fully assembled.

The node only watches and reports — it never moves the object or changes anything in the scene. The answer can flip at any time as the user works: each time it flips, one of the two triggers fires, and a true/false value is always available to read.

## Inputs

Port	Type	What to connect
<b>SceneNode</b>	Scene node	The object you want to watch — the part whose assembly state you care about, for example a <code>Bolt_01</code> . The node reports on this object’s own state.

## Outputs

Port	Type	What you get
<b>OnTrue</b>	Trigger	Fires once at the moment the object becomes fully disassembled — the instant it comes all the way out of place. Wire this to whatever should happen when the part is completely removed.

Port	Type	What you get
<b>OnFalse</b>	Trigger	Fires once at the moment the answer turns false — that is, the instant the object is no longer fully out (it has been seated back into place, even loosely). Wire this to whatever should happen when the part is back in.
<b>Output Boolean</b>	True / false	The current answer, ready to read at any time: <b>true</b> while the object is fully disassembled, <b>false</b> otherwise. Send it into a Branch node, or use it anywhere a true/false value is needed.

## Example

<b>SceneNode</b> input	<code>Bolt_01</code>
<b>OnTrue</b> output	Fires the instant <code>Bolt_01</code> is backed all the way out — you could use it to mark the disassembly step complete.
<b>OnFalse</b> output	Fires the instant <code>Bolt_01</code> is seated back into place — even just dropped in, before tightening.
<b>Output Boolean</b> output	<code>true</code> while the bolt is fully out, <code>false</code> once it is seated or tightened.

## Tips

- Use **Output Boolean** when you just need the current answer somewhere else; use **OnTrue** / **OnFalse** when you want something to happen the moment the part comes out or goes back in.
- The triggers fire only on a change. If the part is already fully out when your scene starts, **OnTrue** won't fire until it is put back and then removed again.
- “Partially assembled” (in place but not tightened) still reports **false** here — this node turns true only when the part is all the way out. To react to the tightened, fully-in state instead, use the matching fully-assembled state node.

Revision #2

Created 9 June 2026 12:47:47 by Rafat

Updated 9 June 2026 12:51:53 by Rafat