EVOKE

ORIGIN

GETTING STARTED WITH VICON EVOKE

What's inside

Vicon Motion Systems is an Oxford Metrics plc company.
Email: support@vicon.com Web: http://www.vicon.com

A RANGE OF LOCATION BASED VR PRODUCTS BY VICON

About Vicon Evoke

Evoke is the Vicon software used to control Vicon's Origin system, designed for use in the Location-Based VR market. Origin system hardware components include:

- Vicon Beacon wireless synchronization unit. The primary hardware synchronization unit for both the cameras (wired) and the Pulsar active marker clusters (wireless).
- Vicon Pulsar active marker clusters. Synchronized active tracking objects with eight infrared LEDs that can be configured in unique patterns to enable tracking by the Evoke software.
- Vicon Viper cameras. Provide powerful and flexible motion tracking, specifically designed for LBVR applications.

For the relevant safety and regulatory information, see the Origin system safety and regulatory information, on docs.vicon.com.

About this guide

This guide briefly describes how to set up and use Evoke for real-time tracking and character solving.

Prepare your Origin system

It is assumed that your Origin system hardware components (including Viper cameras, Beacon(s), and Pulsar active marker clusters) have been placed in your volume and connected to a power supply, and that Evoke is installed and licensed.

If you're installing your Origin system yourself, see any Vicon documentation that was supplied with your hardware and Installing and licensing Vicon Evoke. If you need further help, please contact Vicon Support[1].

1 mailto:support@vicon.com

To set up your Origin system, complete the following procedures:

You may also want to complete the following optional preparations:

⚠ Before you begin:

- Ensure the Beacon(s) that you want to use are connected to a powered PoE+ switch and that this switch is connected to the host PC that runs Evoke.
- To benefit from the latest enhancements and bug fixes for your Vicon system, make sure your firmware is up-to-date (see Update camera firmware, page 28).

Supported devices

Evoke checks for supported Vicon devices before allowing connection. Currently supported devices include:

- Viper, ViperX, Vero, Vantage and Vertex motion capture cameras
- Vue video cameras (to provide full-color reference video, synchronized with the rest of the Vicon system)
- Beacon and Pulsar
- Lock

Ensure Beacon is free of interference

Beacon channels are in the 2.4 GHz ISM band and numbered 11 to 26. To ensure smooth running of your system, make sure that there is no interference from other 2.4 GHz radio/Wi-Fi sources and that nothing blocks the signal between the Beacon and the volume.

To prevent interference and signal-blocking, follow these guidelines:

- Place the Beacon as close to the middle of the volume as possible.
- Place the Beacon high up, to minimize masking of the RF signal by participants.
- Do not re-use a channel number within a site or a co-located group of sites.
- Do not place multiple Beacons closer together than four meters.
- If channel n and channel n+1 are used in a site, do not use channels n-1 or n+2 in that site or adjacent sites.
- If channel n and channel n+2 are used in a site, do not use channels n-2 or n+4 in that site or adjacent sites.

> **Note**
> If your system includes multiple switches, make sure the Beacon is connected to the switch that is directly connected to the PC.

Prepare the Pulsars

Turn on the Pulsars (press the power button once).

Check that the Pulsar firmware is up-to-date and that all the Pulsars are fully charged.

- Check Pulsar firmware, page 8
- Ensure Pulsars are fully charged, page 10
- Keep marker LEDs enabled while charging, page 10

Check Pulsar firmware

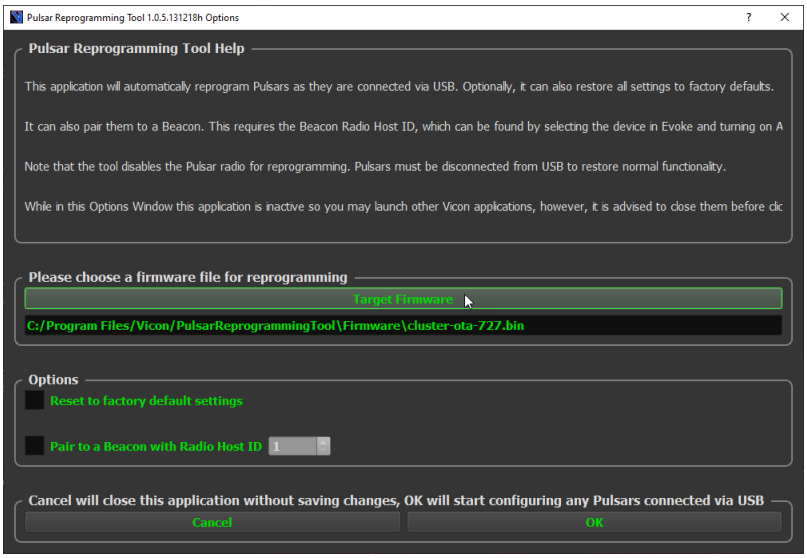
Check the firmware for your Vicon hardware when you first set up your Origin system and periodically afterwards.

> **Important**
> When updating, to avoid connectivity issues, check that all hardware is updated simultaneously to compatible firmware versions. For information on which versions to use, see the release notes.

To update your Pulsar firmware:

1. Connect the Pulsars to the PC via USB, either by using the supplied recharging dock or directly, using USB cable(s).
2. From the Windows Start menu, click **Vicon**, then **Vicon Pulsar Reprogramming Tool**.

3. In the Pulsar Reprogramming Tool window, click Target Firmware and select the firmware file to update to.



4. Before you update, ensure you have closed any other Vicon applications that you may have opened.
5. To update to the selected firmware version, click OK.
6. When the firmware has been updated, reboot the Pulsars, either by unplugging them or by turning off the recharging dock.

> **Tip**
> If at any time, the Pulsars' settings become corrupted (for example, if the Pulsars become unresponsive and don't connect to the system), in the **Options** section, select **Reset to factory default settings** to reset them.

For information on updating firmware for cameras and Beacons, see Update camera firmware, page 28.

Ensure Pulsars are fully charged

If a Pulsar's status light appears pink or red, the battery is less than 20% charged.

To recharge Pulsars:

- Place them in the supplied recharging dock and connect to your power supply; or
- Recharge them using a standard micro-USB to USB cable

The average battery run-time for a Pulsar used at 100% brightness is around nine hours of continuous use. You can extend battery life by disabling Pulsars when not required for tracking, which will give up to 18 hours of typical use.

Note the following times to recharge Pulsar batteries:

- Recharge time to 80%: 1 hr (max)
- Recharge time to 100%: 3 hr (max)

Keep marker LEDs enabled while charging

From Evoke 1.2.2 and later, Evoke can configure Pulsars to keep their IR marker LEDs on while charging. This may be useful if you're using a backpack PC as a power source. The default behavior is still to turn off marker LEDs while charging.

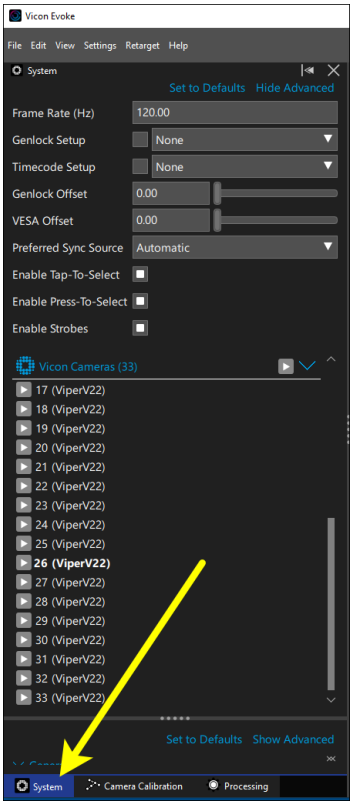
This parameter is available both for Pulsar devices (on the **System** tab) and for Smart Objects (on the **Tracking** tab). When a Pulsar is linked to a Smart Object, it inherits the Smart Object value for **Markers On While Charging**.

Start Evoke

When you install Evoke, a Vicon Evoke desktop shortcut appears on your desktop (and an entry is added to the Windows Start menu).

(Depending on the options selected during installation, you may also see icons for Vicon Retarget, which lets you create retarget setup files (*.vsr), used by Evoke for retargeting; and the Vicon Firmware Update Utility.)

- Double-click the Vicon Evoke icon to start Evoke and ensure the **System** tab is visible.

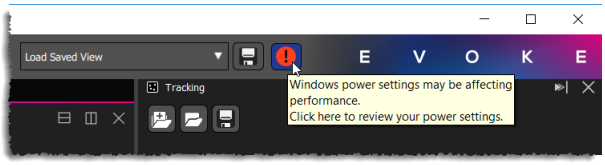


From this tab, you monitor and control devices connected to the software.

> **Note**
> When you have finished setting up your system, you can save your current settings by clicking the Save button at the top right of the **System** tab.
> The next time you need to reload your settings, you can select the saved configuration by clicking the **Load Saved System Configuration** list.

Windows power options monitoring

When you first start Evoke, you may see a warning icon on the right side of the menu bar, like this:



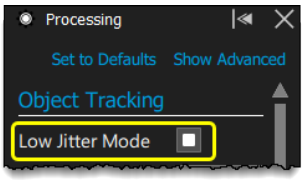
This icon indicates that the current power plan is set to favor power savings over performance. The power-saving features of Windows can significantly reduce Evoke's performance and increase output latency, depending on the power plan that is chosen and the processor support for power-saving features.

For more information and access to the **Power Options** in the Windows Control Panel, click the icon.

For best performance in VR, choose the **High performance** plan (or a plan created from the high performance plan).

Ensure optimal latency

To ensure optimal latency for all scenarios, on the **Processing** panel, ensure **Low Jitter Mode** is selected.



> **Caution**
> Using cameras with a resolution above 5 megapixels may result in sub-optimal latency performance. When using a virtual reality headset, this is not recommended.

Turn on the Beacon and enable pairing

Note that if the Pulsars are already paired to the correct Beacon, you can skip these pairing steps.

To enable pairing:

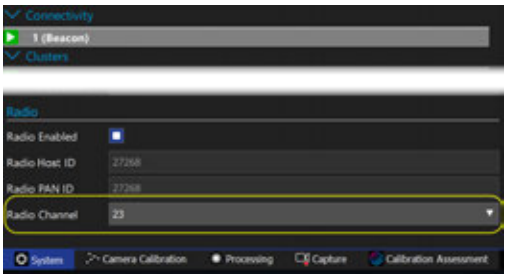
1. Ensure that the Beacon is connected to a powered PoE switch and that this switch is connected to the host PC running Evoke.
2. Turn on the Beacon.
   In Evoke, on the **System** tab, a Beacon is displayed in the **Connectivity** section.
   Check that the Beacon connection status is green.

3. To perform a radio scan to check which Beacon channels are clear from interference from other radio devices and Wi-Fi, right-click the Beacon and select **Radio Channels**.

The **Radio Channels** dialog box shows the signal strength of the currently available channels.

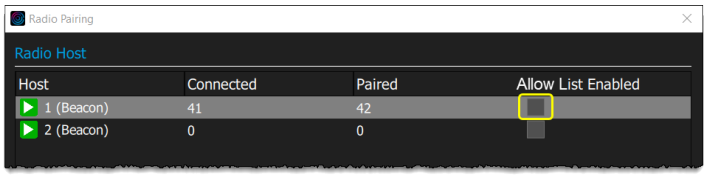
The numbers on the left of the chart represent the signal strength in decibels.
The channel numbers along the bottom of the chart are buttons that enable you to select the channel.
The yellow warning triangles alert you that the channel above is in use (for more information, hover the mouse pointer over the icon).
The light blue bars indicate the average power for each channel and the dark blue bars indicate the maximum power.
In the bars for each channel, the white line represents the last sample taken.
Note that all Pulsars connected to the Beacon are temporarily disconnected while scanning is in progress as the Beacon can't scan channels and broadcast sync at the same time.

4. Select a channel in one of the following ways:
   - In the **Radio Channels** dialog box, click a channel number at the bottom of the chart to select the relevant channel; or
   - On the **System** tab, ensure the Beacon is still selected and in the **Radio** section below, click in the **Radio Channel** field and select an appropriate channel.



   If you choose a channel that is already in use, the Beacon does not turn on sync broadcasts and a warning is displayed.

5. With the Beacon still selected, right-click on it and then click **Radio Pairing**.
6. In the **Radio Pairing** dialog box, click the **Start Pairing** button for the selected Beacon.
   Pairing mode is enabled, which means that the Beacon is available for pairing and any unpaired Pulsar that is turned on (see Pair Pulsars with a Beacon, page 20) will try to connect to it.
7. To specify which pairing requests are accepted, in the **Radio Host** section at the top of the dialog box, select or clear **Allow List Enabled** for the required Beacon:



   - If the allow list is not enabled, all pairing requests are accepted.
   - If the allow list is enabled, only Pulsars on the allow list are accepted.

Other Pulsars are refused, but are displayed in the **Radio Pairing** dialog box so you can choose to accept the request.
To add one or more selected Pulsars to the selected radio host allow list, right-click the Pulsar(s) and then click **Add to <host name> allow list**. The Pulsars on the allow list are now accepted.
In the case of pairing requests, pairing must also be started to enable the Pulsar to connect.

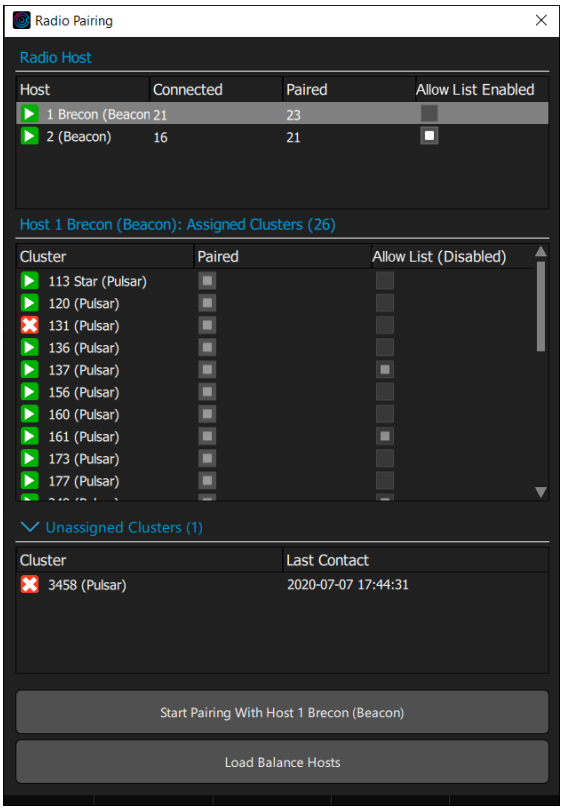
> **Tip**
> Pairing/connection requests persist until you exit Evoke. If a Pulsar fails to connect, check its physical status. Evoke only knows about the last communicated status of the Pulsar, which may now be out-of-date.

For more information about the symbols displayed in the **Radio Pairing** dialog box and how to move Pulsars to and from radio hosts and their allow lists, see About the Radio Pairing dialog box, page 17.

8. On the **System** tab, check that no warnings are displayed next to the Beacon.

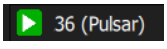
> **Note**
> When you start pairing or change an allow list configuration, the Beacon's sync broadcast changes. Any Pulsars that are in scan mode may try to pair or connect when they detect the change, but this can take a several seconds (longer if there are a lot of Pulsars or a lot of interference).

About the Radio Pairing dialog box

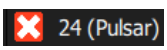


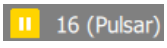
At the top of the **Radio Pairing** dialog box, the **Radio Host** list shows the number of clusters connected to and paired with the radio host, and enables you to choose whether the allow list for the selected Beacon is enabled.

Below the **Radio Host** section, the **Assigned Clusters** section lists Pulsars that are paired to the selected Beacon or are on its allow list.

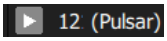
- 

A green 'Play' symbol indicates a Pulsar that is paired and connected to the selected host's allow list.
- 

A blue 'i' indicates a Pulsar that is paired to the system, but unable to connect because it's not on the host's allow list.
- 

A red 'x' indicates a Pulsar that is unpaired. It cannot connect unless the Beacon is put into pairing mode (and the Pulsar is on the allow list, or the allow list is disabled).
- 

A yellow 'Pause' symbol indicates a Pulsar that is connected but disabled.

- 

A gray 'Play' symbol indicates a Pulsar that is connected but not linked to a Smart Object.
- 

A magenta 'i' indicates a pairing request.
- 

No icon indicates a Pulsar is missing (ie, it hasn't connected during this session).
- 
A partially selected check box in the **Paired** column indicates a Pulsar that is paired to a Beacon that is different from the one that is currently selected.

To perform the following operations on Pulsars in the **Assigned Clusters** list, right-click on one or more selected Pulsars and select the required option:

- **Add to** or **Remove** Add or remove the selected Pulsar(s) from the selected radio host allow list
- **Clean the allow list**. Remove any Pulsars from the selected radio host allow list that are paired to another radio host. This option is useful after load-balancing (see Load balancing between hosts, page 19).
- **Unpair** Unpair the selected Pulsar(s) from the selected radio host
- **Transfer** Pair the selected Pulsar(s) to another connected, enabled radio host

At the bottom of the **Radio Pairing** dialog box, the **Unassigned Clusters** section lists connection or pairing requests, and unpaired Pulsars. This list persists for the lifetime of the application session. To add Pulsars in this list to to the selected radio host's allow list, right-click on the Pulsar(s) and then click **Add to Host <hostname> allow list**.

Load balancing between hosts

The number of clusters connected to each Beacon must be balanced. If too many clusters are connected to one host then connection stability is reduced, and the time required to send commands to all clusters increases.

> **Important**
> To avoid loss of connection stability and an unacceptable delay in sending commands to all clusters, we recommend that no more than 80 Pulsars are connected to a single Beacon.

To automatically balance the number of Pulsars between Beacons, click the **Load Balance Hosts** button at the bottom of the dialog box. This transfers the connected clusters so that an equal number is connected to each host.

After load balancing, you may find a number of clusters on one host's allow list are paired to a different host. To remove these clusters from the allow list, right-click on the cluster(s) and then click **Clean Host <host name> allow list**.

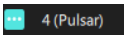
Pair Pulsars with a Beacon

After you have turned on a Beacon and made sure it is available for pairing, you can pair Pulsars to it.

1. On a Pulsar, press the power button to turn it on.

> **Tip**
> If you transfer a Pulsar between Beacons, you must un-pair the Pulsar so that it can connect to the new Beacon. To do this, during the 10-second period after booting, double-press the Pulsar button to un-pair it.

The Pulsar automatically searches radio channels to find the Beacon it is paired with, or if unpaired, a Beacon that has pairing enabled (see Turn on Beacon and enable pairing, page 13).
In Evoke, on the **System** tab, in the **Clusters** section, a Pulsar is displayed.

2. Ensure that the Pulsar connection status is cyan (connected but not assigned a marker pattern).


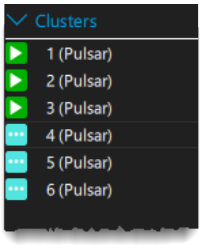
If the Pulsar's battery level is low, a status icon indicates this.
If any other icon is displayed, see Pulsar status icons in Evoke, page 21 to troubleshoot possible issues

3. When all the Pulsars have been paired, turn off pairing mode. To do this, either click the **Stop Pairing** button or close the **Radio Pairing** dialog box.

> **Tip**
> To deactivate a Pulsar, on the **System** tab, select the Pulsar and in the **General** section below, clear the **Enabled** check box.
> In the **Clusters** section, the Pulsar connection status turns yellow. The marker LEDs turn off, to reduce power consumption.

Pulsar status icons in Evoke

On the **System** tab, in the **Clusters** part of the System tree, Pulsars that are paired with a connected Beacon are displayed. Adjacent icons indicate the status of the Pulsars:



The following table describes each of the Pulsar status icons that are displayed on the **System** tab in Evoke.

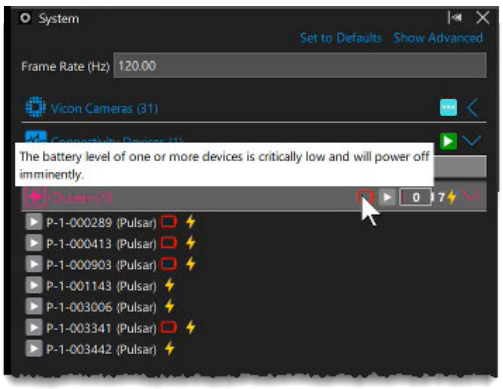
Icon	Meaning
Cyan	Connected but not assigned a marker pattern
Yellow	Connected but disabled
Gray	Automatically disabled as not linked to a Smart Object
Green	Enabled and assigned a valid marker pattern
Red	Disconnected
Yellow battery	Battery has low charge (less than 20%)
Red battery	Battery has very low charge (less than 10%)
Lightning bolt	Plugged in for recharging
Update	Evoke has queued or sent new settings to the Pulsar but has not yet received a response
No icon	Loaded from a previous Evoke session but not yet connected to this instance of Evoke

Note that the overall system charge is determined by the device with the lowest charge (unless recharging – see the following note), which is shown at the top of the **Clusters** section on the **System** tab.

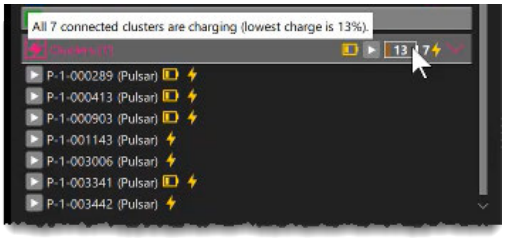
The number of devices on charge is also displayed here.

> **Note**
> The information displayed by the **Lowest Battery Charge** indicator depends on the number of devices that are currently charging:
> - If some connected devices are currently charging, they are ignored, as they aren't at risk of running out of charge.
> - If all of the connected devices are recharging (indicated by a lightning bolt icon on the charge bar), they are included, so that you know when they are ready for use.

In the following image, the red battery icons to the right of the Pulsar names indicate that the batteries of the relevant Pulsars are low. The **Lowest Battery Charge** indicator shows that the Pulsar with the lowest charge is almost at 0%, ie, almost fully discharged (a Pulsar whose battery is fully discharged can no longer communicate with Evoke and disconnects, and the **Lowest Battery Charge** indicator then displays the Pulsar with the next lowest battery).



In the following image, the yellow lightning icons to the right of the Pulsar names indicate that the batteries of the relevant Pulsars are charging (and the yellow batterty icons indicate that the charge is still low). The **Lowest Battery Charge** indicator shows that the Pulsar with the lowest charge is now at 13%.



Enable Evoke to track Pulsars and props

Both Smart Objects and basic objects define a pattern of markers that Evoke can locate from camera centroid detections.

Create Smart Objects

A Smart Object is the representation in Evoke of a type of object that has programmable active markers (eg, a Pulsar). Pulsars are smart in that they communicate with the Evoke PC via a Beacon. This enables a Pulsar to set its marker pattern, sync to the Viper cameras and report its battery level back to the PC.

To enable tracking of Pulsars, you must create a Smart Object in Evoke for each Pulsar that you want to track.

For more information, see Work with Smart Objects, page 39 or watch the Vicon video:

Evoke 1.2 - Smart Objects[2]

Create basic objects

A basic object lacks the two-way communication of a Smart Object (eg, a Nova active strand). A Nova is a basic object as its active markers can be either on or off and it has to be controlled manually.

To enable tracking of props, you must create basic objects in Evoke for each prop that you want to track.

For more information, see Work with basic objects, page 59 or watch the Vicon video:

Evoke 1.2 - Basic Objects[3]

2 https://youtu.be/LFwQvQh0RfA
3 https://youtu.be/ENT5df3Yh8E

Manage your tracking configuration

When you have finished setting up your Smart Objects and basic objects, you can save the whole tracking configuration (including all basic and Smart Objects) in MCP format for future use.

To do this, at the top of the **Tracking** pane, click the **Save tracking configuration** button:

The default location for tracking configuration files is:

C:\Users\Public\Documents\Vicon\Evoke1.x\Tracking

When you want to re-use your tracking configuration, click the **Load tracking configuration** button to re-load the file.

Position cameras and Pulsars

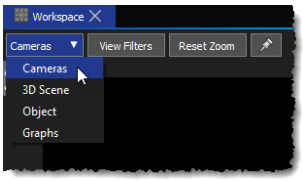
With your Origin system installed and licensed, and after you have created Smart Objects, you can position the cameras and markers.

1. Position your cameras around the volume, ensuring that two or more cameras can see every point in the volume in which you intend to track motion.
2. Place Pulsars around the floor to outline your volume.

Check volume coverage

After you have positioned cameras and Pulsars in the volume, ensure that the cameras can see the whole of the volume.

1. Ensure your cameras are physically connected to the system.
2. Ensure your Origin system hardware is switched on, and that Evoke is running.
3. On the **System** tab (by default located on the left of the Evoke window), SHIFT+click to select all the cameras or, for a large number of cameras, right-click and click the relevant **Select All** option.
4. In the **Workspace** ensure that the **Cameras** view is selected.



5. Using the default lens settings on each camera, ensure that:
   - You can see Pulsar images from each connected camera.
   - Two or more cameras can see every point in the volume in which you intend to track motion.

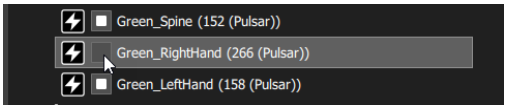
Deactivate unused objects

It's important to ensure that any objects that aren't currently in use are deactivated.

Evoke is constantly trying to find all enabled objects. Objects that are enabled but not visible in the volume (eg, Pulsars that are charging, powered off or out of the volume) may cause spurious object tracks, or impede the tracking of the objects that are visible in the volume.

To deactivate an object:

- On the **Tracking** pane, select it and in the **General** section below, clear the **Enabled** check box.
  Or
- In the **Tracking** tree, clear the relevant check box:



Update camera firmware

Vicon hardware is programmed with firmware to control its operation. Periodically, Vicon supplies firmware updates to correct or improve device functionality. You apply these firmware updates to your Vicon cameras and Beacon(s) via the Vicon Ethernet network using the Vicon Firmware Update Utility, as described below.

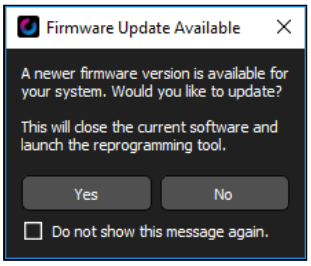
You are automatically notified when any component of your Vicon system is running out-of-date firmware, and given the opportunity to update to the latest version.

> **Important**
> To ensure optimum performance and access to all the latest functionality, Vicon recommends that you upgrade to the latest firmware whenever it becomes available.

To monitor and/or upgrade system firmware:

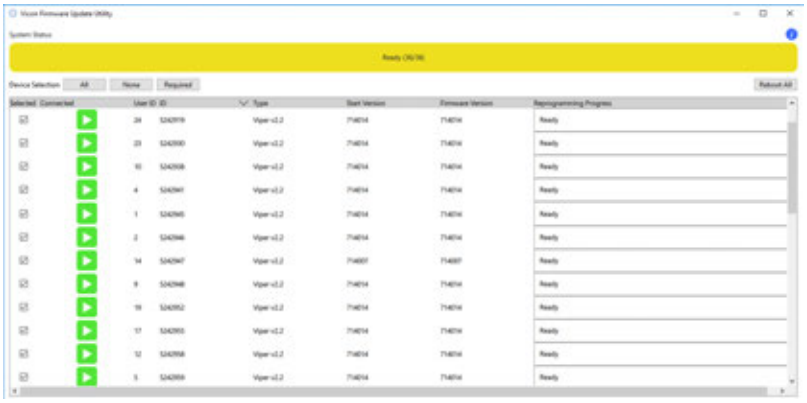
1. When you start Evoke or connect any Vicon devices into your system, Evoke checks to see whether the firmware for your cameras and Beacon(s) is up-to-date.
   If your devices aren't using the latest firmware, Evoke displays an icon in the toolbar to let you know that a more up-to-date version of the firmware is available:

2. Click the icon to display more information.
   Evoke displays a prompt that enables you to open the Vicon Firmware Update Utility (reprogramming tool).

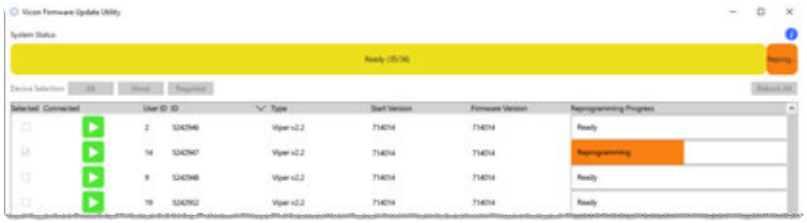


3. Click Yes to open the Vicon Firmware Update Utility. Note that you can also open the Vicon Firmware Update Utility from the **Start** menu (select **Vicon** > **Vicon Firmware Update Utility**).

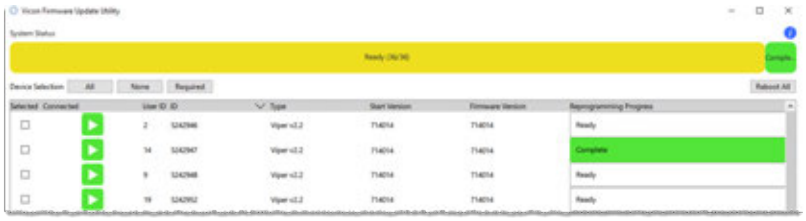
Evoke closes and the Vicon Firmware Update Utility is displayed, showing all the connected devices and their current firmware version. By default, all devices are selected.



4. If you don't want to update any of the devices, clear the relevant check box(es).
   Note that if required, you can select devices to be updated that are already using the latest version.
5. At the bottom of the Vicon Firmware Update Utility window, in the **Choose Firmware version** list, select or browse to the required firmware version.
6. Click **Reprogram** to update the firmware for the selected device(s).



When updating is complete, the **Firmware Version** column displays the updated firmware version and the **System Status** line and the **Reprogramming Status** column display **Complete** on a green background.



> **Note**
> If you do not have continual internet access, Evoke is unable to notify you when a new version of the system firmware is available. In this case, install the Vicon Firmware Update Utility on an internet-connected machine to detect and download the latest version of the firmware. You can then transfer this download to the local machine and use the Vicon Firmware Update Utility to update to the latest version of the firmware.

To downgrade to an earlier firmware version

To downgrade to a firmware version that was previously downloaded, open the Vicon Firmware Update Utility (from the **Start** menu click **Vicon** > **Vicon Firmware Update Utility**) and select the required firmware version.

Make any optional changes required

In addition to the preparations described at the beginning of Prepare your Origin system, page 4, depending on your requirements, you may also want to make the following changes:

- Change camera settings, page 33
- Customize views and layouts, page 33
- Change the auto-save interval, page 37
- Reduce processing rate for large numbers of objects, page 38

Change camera settings

The default settings for Viper cameras are suitable for many typical usage scenarios. Viper cameras are sensitive to the infrared light emitted by the Pulsar marker LEDs. They are factory-configured with the aperture set to F8 and back-focused to a depth of field 0.8 m–∞. No additional hardware setup is required on site, however, in some circumstances (eg, if you have a small volume with little IR noise) you may benefit from adjusting the **Threshold** setting on the **Processing** panel in Evoke.

> **Note**
> When using the Viper cameras' tap-to-select feature, note that taps that are also registered by other cameras mounted nearby are ignored. To minimize the effects of vibrations on the rigging, tap lightly.

Customize views and layouts

Evoke provides you with a number of ways to customize both what you view in the Evoke Workspace and the layout of the windows. You can change and customize the view to suit your way of working.

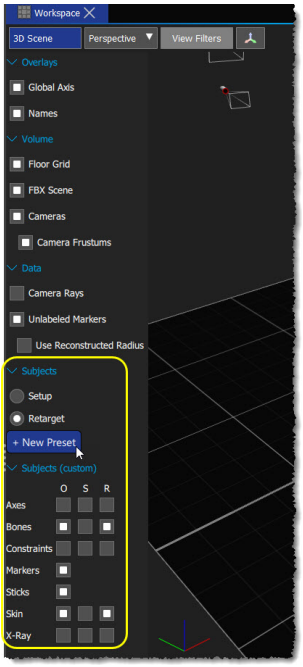
- Set appropriate View Filters, page 34
- Change camera orientation detection, page 35
- Display the number of selected markers, page 35
- Save your window layouts, page 36

Set appropriate View Filters

While you're working with Evoke, you can set the View Filters to give the most useful view of your data.

The default view filter sets are for setup and retargeting. The **View Filters** for the **3D Scene** view include options for solving and retargeting.

To add your own filter sets, in the 3D Scene view, click **View Filters** and in the **Subjects** section, click the **New Preset** button.



The options are displayed in a matrix, with columns for **O**bject, **S**olving and **R**etargeting, enabling you to choose a view configuration that suits your current task.

Your settings, including any custom presets, are automatically saved, so that you can easily re-use them for different workflows.

For details, see the Vicon video:

Evoke 1.2 - View Filters[4]

4 https://youtu.be/6xBon9Q1a-0

The view filter presets are stored in the view presets file, by default:

C:\Users\Public\Documents\Vicon\Evoke1.x\<username>\SubjectViewPresets.xml

The current view filter preset is stored in the view file, by default:

C:\Users\Public\Documents\Vicon\Evoke1.#\LastRun\<username>\LastRun.View

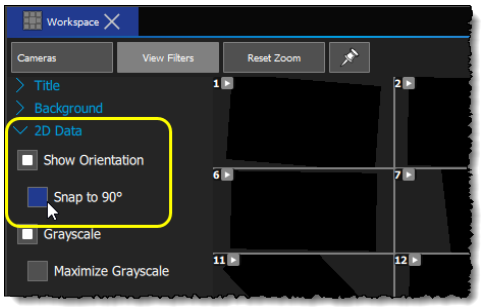
Change camera orientation detection

View frames in the Cameras view can be rotated to match their physical orientation, so that up in the image matches up in the real world. The orientation is determined from the camera calibration if available, otherwise the accelerometer on board the camera is used.

This option is enabled by default.

To turn on or off camera orientation detection:

1. In the **Cameras** view, click **View Filters**.
2. In the **2D Data** section, select or clear **Show Orientation**.
3. To choose whether to snap to 90° increments, ignoring smaller rotations, select or clear **Snap to 90°**.



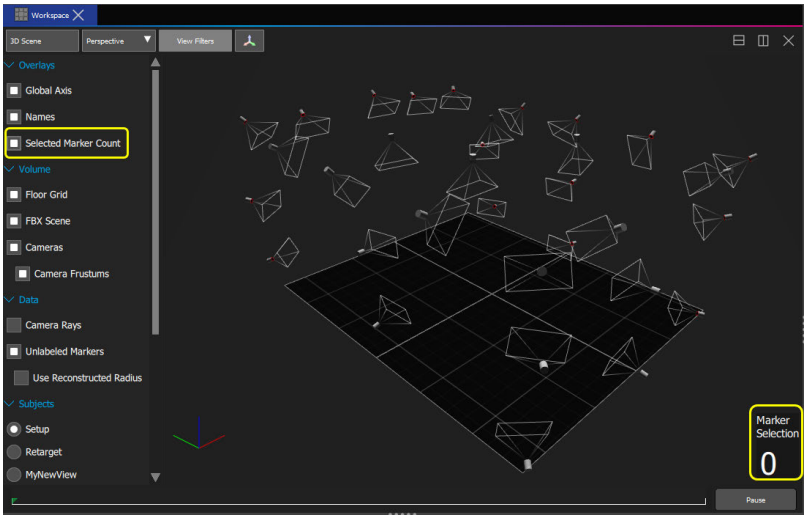
Display the number of selected markers

To quickly check that all markers on a particular target are visible, you can select an option in the **3D Scene** view to show the number of markers currently selected.

To display the number of selected markers:

1. In the **3D Scene** view, click **View Filters**.

2. In the **Overlays** section, select or clear **Selected Marker Count**.
   At the bottom right of the view pane, the **Marker Selection** count is displayed. If no markers are currently selected, a zero is displayed.

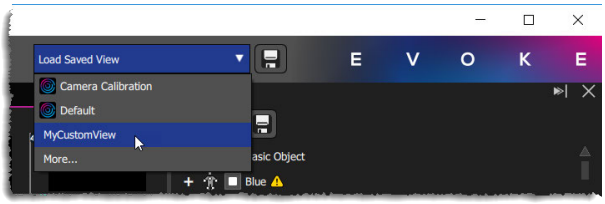


   As you select markers, the Marker Selection counter changes to display the number of selected markers.

Save your window layouts

In addition to using the supplied Default tracking and Camera Calibration layouts, you can save your own customized layouts of the workspace and panels.

This enables you to quickly switch between layouts when either setting up the system or running experiences.



Custom layouts are saved by default to:

C:\Users\Public\Documents\Vicon\Evoke1.x\<username>\Views

Change the auto-save interval

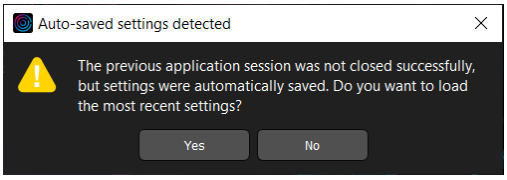
Evoke settings are automatically saved while you're using Evoke, as well as when you exit. The auto-saved files are located in the usual settings folder, which defaults to:

C:\Users\Public\Documents\Vicon\Evoke1.#\LastRun<username>

These files are replaced by the normal last run settings files when Evoke closes:

Temporary filename	Normal filename
AutoSave.System	LastRun.System
AutoSaveSubjects.mcp	Subjects.mcp
AutoSave.View	LastRun.View
AutoSave.HotKeys	LastRun.HotKeys

If any of the AutoSave files is detected when Evoke starts, you're prompted to confirm which files to load:



- To load the most recent (auto-saved) settings, click **Yes**.
  or
- To load settings from the last successfully closed session, click **No**.

This feature prevents you from losing your settings if Evoke closes abnormally, for example, due to process termination, machine reboot, etc.

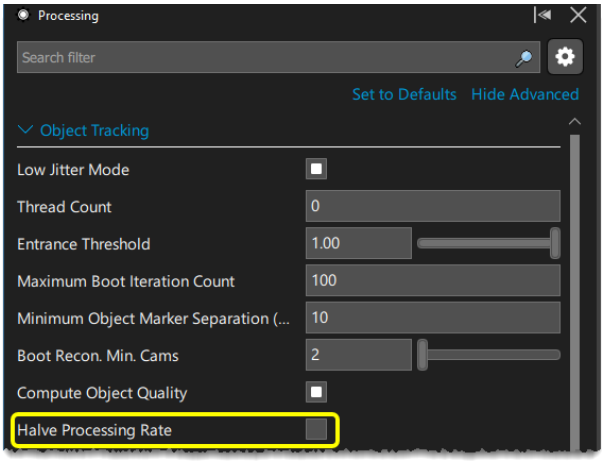
Note that the auto-save files are only updated if there have been changes since the last auto-save.

To change the interval at which files are auto-saved:

1. On the **Settings** menu, click **Preferences** (or press Shift-P), then in the **Preferences** dialog box, click the **User Preferences for** *<username>* tab.
2. In the **Interface** section, change the value in the **Autosave interval** field from the default (60 seconds).

Reduce processing rate for large numbers of objects

Normally, when the system is heavily loaded, to avoid dropping frames, you must reduce the system frame rate. When the (Advanced) processing option **Halve Processing Rate** is selected, you can keep the system frame rate high, but reduce the rate of subsequent processing (eg, object and character tracking). A high frame rate may improve tracking quality (particularly for fast movements) while subsequent processing is reduced, lowering processing load requirements on the host PC.



When using the **Halve Processing Rate** option, note the following points:

- 2D centroid detection always happens at the system frame rate
- Subsequent processing depends on whether **Halve Processing Rate** is selected or not:
  - When cleared (the default), every frame is processed as normal
  - When selected, every other frame is dropped (giving a tracking rate of 1:2)
- When using the Datastream, if **Halve Processing Rate** is selected, dropped frames occur at every other frame.

Work with Smart Objects

To enable the Pulsars to be tracked, you create Smart Objects in Evoke for each Pulsar in your system.

You manage objects in the **Tracking** pane, which lists all objects tracked by Evoke. You can visually distinguish Smart Objects from basic objects by their icons.

 Smart Object

 Composite Smart Object

 Basic object

For information on working with Smart Objects see the following topics:

- Display the required items in the Tracking panel, page 40
- Create Smart Objects, page 40
- Confirm tracking and labeling, page 47
- Reassign Smart Object patterns, page 48
- Merge Smart Objects, page 51
- Split a merged Smart Object, page 54
- Use Smart Object templates, page 54
- Change a Smart Object's origin, page 57
- Swap clusters, page 58

See also the Vicon video:

 Evoke 1.2 - Smart Objects[5]

5 https://youtu.be/LFwQvQh0RfA

Display the required items in the Tracking panel

By default, all items are displayed in the Tracking panel. To display only the items needed for the current task, you can display or hide items in the Tracking panel, depending on their type.

To toggle the display of tracking items:

At the top right of the **Tracking** panel, click the relevant button(s):

The buttons toggle the display of:

- Basic objects (non-assigned)
- Characters (including assigned objects)
- Smart Objects (non-assigned)

Create Smart Objects

To enable the Pulsars to be tracked, you must create Smart Objects in Evoke for them.

You can create Smart Objects for Pulsars in any of the following ways:

- Create Smart Objects automatically, page 42
- Create Smart Objects for selected Pulsars, page 44
- Create Smart Objects manually, page 45

Each Smart Object is assigned a marker pattern, which is sent to the linked Pulsar.

When you select a Smart Object, its linked Pulsar device is also selected and its status lights display a selection sequence. The object is also selected in the **3D Scene**.

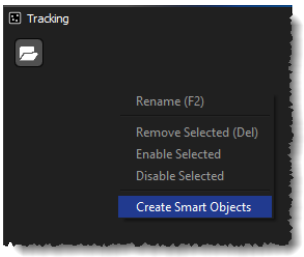
On the **System** tab, the Pulsar connection icons on the linked Pulsars turn green as they are assigned patterns by the Smart Objects.

> **Note**
> When Smart Objects are created, some cluster parameters, such as **Enabled** and **Marker Pattern** become read only, as these are managed by Smart Objects.

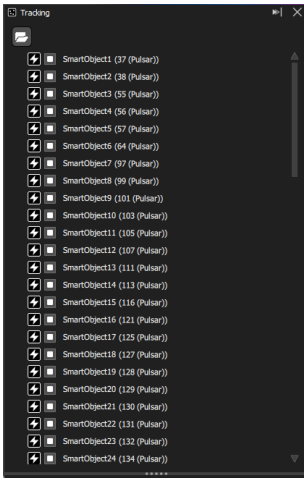
Create Smart Objects automatically

The following steps let you quickly create Smart Objects for all connected Pulsars (except Pulsars that are already linked to a Smart Object).

1. In the **Tracking** pane, right-click and then click **Create Smart Objects**.



Smart Objects are created for all connected Pulsars that are not already linked to a Smart Object, up to a maximum of 70 with patterns.



The Smart Objects are automatically linked to each Pulsar and assigned a unique name and pattern.

Rename Smart Objects

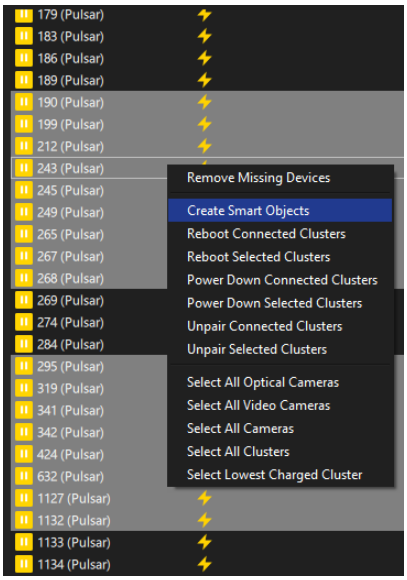
When you create Smart Objects automatically, they are allocated names. It is normally useful to rename them to match their intended location (eg, `Green_LeftHand` or `Red_RightFoot`).

To rename Smart Objects:

1. In the tree at the top of the Tracking pane, do one of the following:
   - Double-click the Smart Object.
   - Right-click the Smart Object and then select **Rename**.
   - Select the Smart Object and press F2.
   - Select the Smart Object and in the **Properties** pane below, click the **Name** field.
2. Enter a unique new name for the Smart Object.

Create Smart Objects for selected Pulsars

1. In the System tree, select the required Pulsars (click, Shift+click, Ctrl+click, drag-select, etc).
2. Right-click and then click **Create Smart Objects**.



Smart Objects are created for the selected Pulsars (unless they are already linked to a Smart Object), linked to each one and automatically assigned a unique pattern.

3. Rename the Smart Objects appropriately (see Rename Smart Objects, page 43).

Create a Smart Object manually

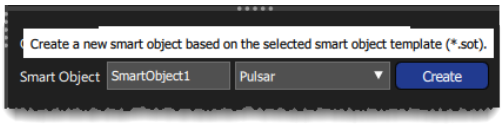
When you create a Smart Object manually, you can select the appropriate template for your Smart Object and you must link the Smart Object to the relevant Pulsar(s) to enable tracking.

In addition to the Pulsar template, templates are supplied for the Vicon headset mount for supported headsets: the Oculus Rift, HP Reverb and HTC Vive. The templates are pre-configured with sections corresponding to the Pulsar slots at the top and front of the headset, and the Smart Object origin set to the optical origin of the headset. Templates for the Vicon hat and backstrap are also supplied.

If required, you can create additional templates (see Use Smart Object templates, page 54).

To create a Smart Object manually:

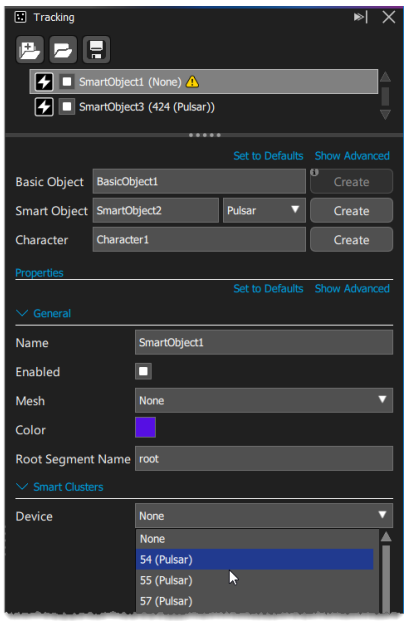
1. In the **Tracking** pane, in the **Smart Object** field, enter a unique name for the new Smart Object, ensure that the required template is selected and then click the **Create** button.



The new Smart Object is displayed at the top of the Tracking pane.

It has the name you entered and is automatically assigned a unique pattern.
Notice that no device is associated with it yet. To enable tracking, you must link the smart object to a Pulsar.

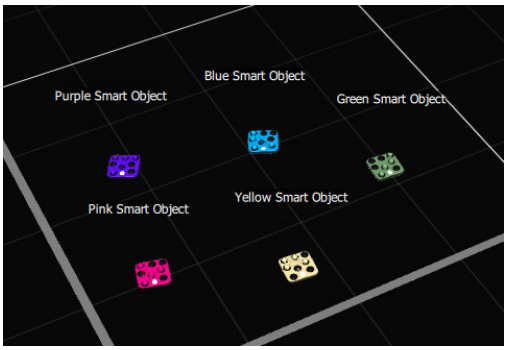
2. In the **Properties** pane, go to the **Smart Clusters** section and in the **Device** field, select the Pulsar that you want to associate with this Smart Object.
   If you are using the template supplied for the Vicon headset mount for a supported headset, you must select a Device for both Pulsars.



3. Enter the remaining details for the Smart Object (color, mesh, etc).
   The mesh is an FBX file with the mesh skinned to a Root bone. You can use the installed meshes that are displayed when you click on the **Mesh** list, or your own FBX files, saved to *C:\Users\Public\Documents\Vicon\PropMeshes*.

Confirm tracking and labeling

After you have created the required Smart Objects, make sure they're displayed in the **3D Scene**, with labels matching the Smart Object names.



Cluster and Smart Object settings are stored in the *System* file. Its default location is:

C:
\Users\Public\Documents\Vicon\Evoke1.4\LastRun<username>
\LastRun.System

Object settings are stored in the *Subjects* file. Its default location is:

C:
\Users\Public\Documents\Vicon\Evoke1.4\LastRun<username>
\Subjects.mcp

Settings are auto-saved periodically when changes are made as well as when you exit Evoke.

Reassign Smart Object patterns

The number of unique patterns available for tracking without using proximity grouping is limited to:

- 56 patterns when using 5 from 8 markers; or
- 70 patterns when using 4 from 8 markers

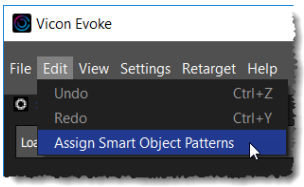
To use more than 70 patterns, see Use proximity-based tracking, page 148.

If more Smart Objects are created than available patterns, the excess smart objects are assigned pattern 0 (no markers on), and show a warning in the **Tracking** pane. The linked Pulsars also show a cyan status icon in the **System** tab.

The pattern set is chosen automatically, depending on the number of Pulsar clusters selected when creating Smart Objects.

If you later change the number of Smart Objects, you can reassign patterns to all loaded smart objects and if necessary change the pattern set. To do this:

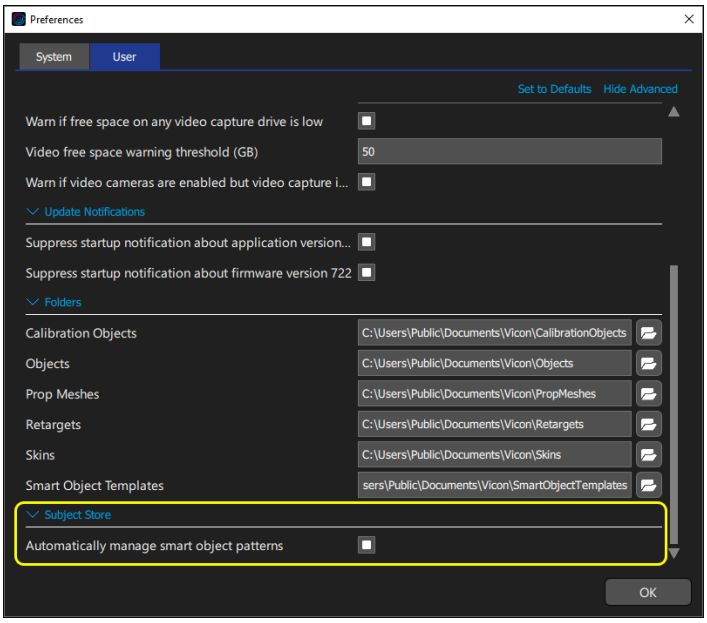
- On the **Edit** menu, click **Assign Smart Object Patterns**.



Smart Object pattern management

You can assign patterns to Smart Objects in either of these ways:

- Enable Evoke to automatically assign a unique pattern to each Smart Object. This is the default option.
  To check that Evoke is managing Smart Object patterns:
  - Open the Preferences dialog box (**Settings** > **Preferences** or press Shift-P) and at the bottom of the **User** tab, ensure that **Automatically manage smart object patterns** is selected.



When this option is selected, Evoke automatically manages the patterns for Smart Objects whenever a Smart Object is created. Note that:

- The new Smart Object is assigned a unique pattern from the current pattern set, if one is available.
- The initial pattern set is 5-from-8, meaning that 5 of the Pulsar marker LEDs are turned on. There are 56 unique patterns in this set.
- If you require more than 56 patterns, Evoke switches to the 4-from-8 pattern set. There are 70 unique patterns in this set.
- Any Smart Objects that aren't using the correct pattern set are assigned a new pattern.
- If you require more than 70 patterns, excess Smart Objects are assigned pattern 0, meaning they will not turn on any markers and can't be tracked.

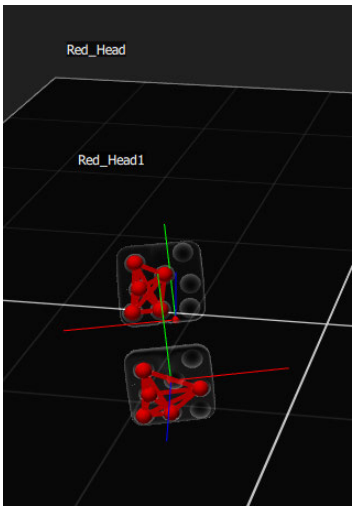
- Assign patterns using the Evoke API (see Vicon Evoke API & automation). In this case, if you don't want Evoke to make changes to the patterns, in the **Preferences** dialog box, clear the **Automatically manage smart object patterns** option.

Merge Smart Objects

To get good tracking performance from different viewpoints or under occlusion, you may need to attach multiple Pulsar clusters or extra markers to a single rigid object. In particular, if you are using VR headsets and/or guns or similar props, you may benefit from the better rotational stability and/or occlusion resistance offered by combining multiple Pulsar clusters.

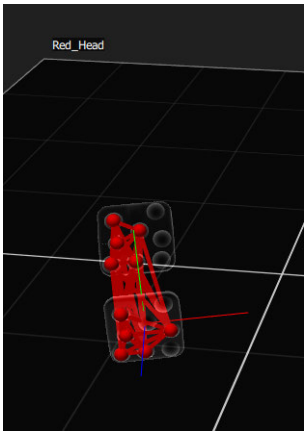
To merge Smart Objects:

1. Create Smart Objects as normal (see Create Smart Objects, page 40).

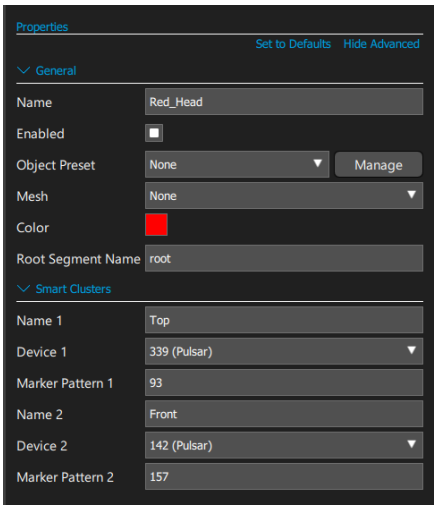


2. In the **Tracking** pane, select the Smart Objects that will contribute to the merged object, right-click and then click **Merge Smart Objects**. In the **Tracking** pane, the merged Smart Object is displayed with a different icon .

3. Enter a name for the merged Smart Object.
   The selected Smart Objects are replaced with a single Smart Object, centered on the centroid of the previously selected objects.



In the **Properties** pane for the merged Smart Object, the properties of the combined objects are displayed (the Smart Object that was selected last appears first in the Smart Clusters section). Note that in the following example, the Advanced properties are displayed.

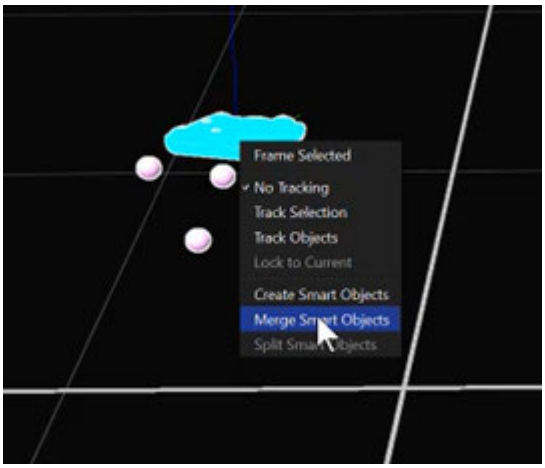


Merge additional markers into Smart Objects

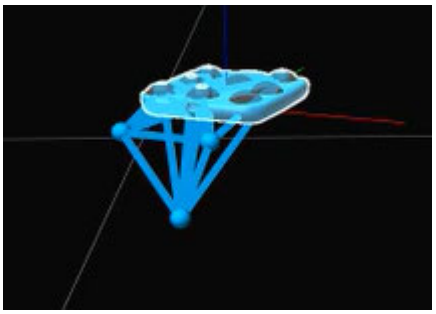
In addition to merging Smart Objects, page 51, you can combine them with additional markers to create more complex objects while maintaining Smart Object functionality (for example, adjustable patterns). This enables Smart Objects to benefit from more widely spaced LEDs, enhancing tracking performance and visibility.

To merge additional markers into a Smart Object:

- In the Evoke **3D Scene** view, select the required Smart Object and the additional markers, right click, and select **Merge Smart Objects**.



In the 3D Scene view, the merged object is displayed.



You can save this composite object as a template and use it as a family of objects, each with a unique Smart Object pattern for identification. Note that because these additional markers are not controlled by a Smart Object, they cannot be enabled or disabled with the rest of the composite object.

Split a merged Smart Object

If you need to split Smart Objects that you have previously merged, you can separate them again.

To split apart a merged Smart Object:

1. In the **Tracking** pane, select the merged **Smart Object**.
2. Right-click the merged object and then click **Split Smart Objects**. The separated objects are given the name of their device slot (e.g. Top, Front), appended with a number if that name is already in use.

Use Smart Object templates

Smart Object templates let you easily create Smart Objects for Pulsars and supported accessories. Evoke comes with templates for the Vicon-supplied accessories Backstrap, Hat, and supported headsets: Oculus Rift, HP Reverb and HTC Vive (the templates for the clip for the headset, with two Pulsar mounts), as well as for a single Pulsar.

You can also create custom templates, for example, for a backpack, to speed up the creation of merged Smart Objects.

The template stores information about each device and its relative orientation, together with the merged object properties (object preset, mesh, color and root segment name). Smart Objects that are created from this template are automatically assigned patterns, so you only need to select the required Pulsar from the **Device** field to enable tracking.

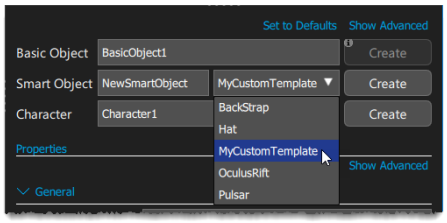
You can assign to any objects a mesh that is displayed in the 3D Scene.

To create a template for a merged Smart Object

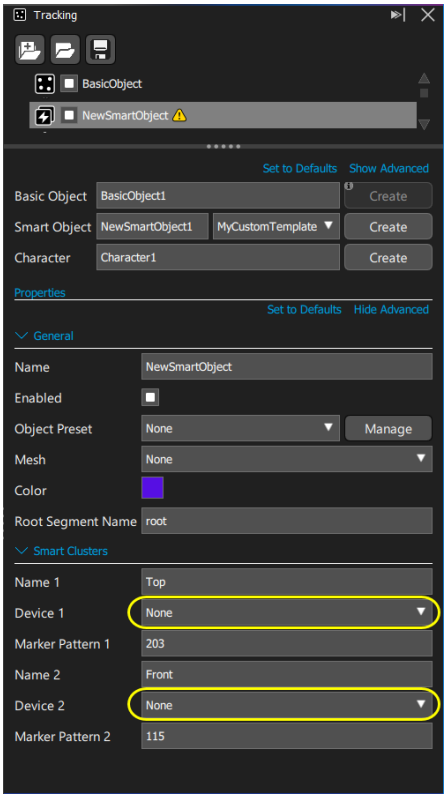
1. Create the merged smart object (see Merge Smart Objects, page 51) with the required mesh and other attributes.
2. In the **Tracking** pane, right-click the composite smart object and then click **Export Template**.
3. In the **Export Smart Object Template** dialog box, enter a name for the template and then click **Save**.
   The template is saved into the default location:
   C:\Users\Public\Documents\Vicon\SmartObjectTemplates
   You can now use your template to create further smart objects.

To apply a custom template:

1. Create a Smart Object as described in Create a Smart Object manually, page 45 , but instead of Pulsar, select your custom template:



2. In the **Smart Clusters** section of the **Properties** panel, click the **Device** lists and select the required Pulsars.



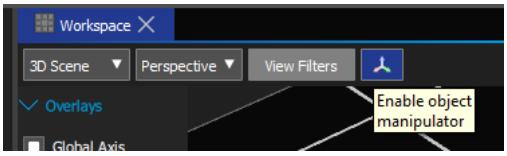
When you have entered these details, the object is displayed in the 3D view with the mesh and correct orientation.

Change a Smart Object's origin

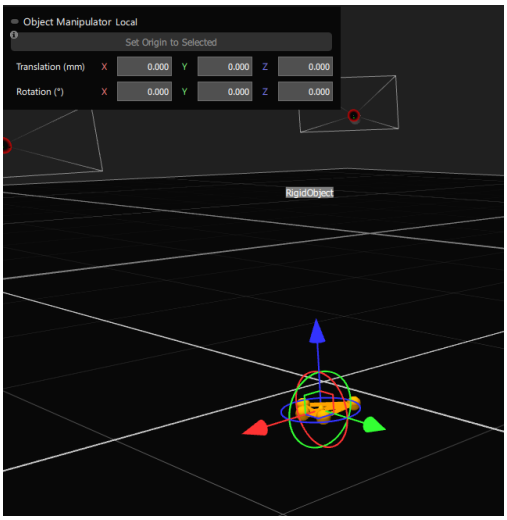
The object manipulator enables you to specify a Smart Object's origin (ie, the center of the physical object in relation to the marker pattern that is tracked by Evoke).

Note that you only need to change the origin of Smart Objects that you create manually or from custom templates: the built-in Smart Object templates already have the correct origin set.

1. To activate the object manipulator:
   a. Click the **Pause** button at the bottom right of the **Workspace** to pause Evoke.
   b. Select the Smart Object.
   c. In the **3D Scene** view, click the object manipulator button.



2. In the Workspace either:
   - Drag the manipulator to move or rotate the Smart Object to the required location; or
   - Enter the required values in the **Translation** and/or **Rotation** fields;



> **Tip**
> To scale the Manipulator, on the numeric keypad, press + (scale up) or - (scale down).

Note that the manipulator operates in local or global space (corresponding to the object coordinate system or the world coordinate system respectively). To switch between the two, click the icon to the left of the **Object Manipulator** text.

Swap clusters

Use the **Swap Clusters** option when you need to replace a device, for example, if you need to change one of your Pulsars because its battery charge is low or if you have created a Smart Object using one of the supplied templates (OculusRift, etc) and need to link a physical device to its Smart Object.

1. Ensure the replacement Pulsar is in the volume.
2. Create a Smart Object, page 40 for it if it doesn't have one already.
3. With the new Smart Object selected, hold down the Ctrl key and select the Smart Object for the Pulsar that you want to replace. (Note that selecting the device in the **System** tree also selects the Smart Object if one is present.)
4. If if either of the Smart Objects is a composite Smart Object, from the sub-menu, select which device slot is to be affected by the swap. (The other selected Smart Object must have a single device slot.)
5. In the **Tracking** pane, right-click and select **Swap Cluster**.
   The new Pulsar is used by the existing Smart Object or slot.

Work with basic objects

To enable tracking of your props, you create basic objects in Evoke.

> **Tip**
> It is easiest to use the Vicon Nova kit to mount active markers on a variety of props and use them within Evoke.

You manage objects in the **Tracking** pane, which lists all objects tracked by Evoke. You can visually distinguish basic objects from Smart Objects by their icons.

Basic object

Smart Object

Composite Smart Object

Basic objects have a fixed pattern, so if a marker is moved you must re-create the object.

Note that you can import VSK files for basic rigid objects.

For more information, see:

- Display the required items in the Tracking panel, page 60
- Create basic objects, page 60
- Add a mesh to a basic object, page 63
- Change a basic object's origin, page 64
- Export a basic object as a VSK, page 65

See also the Vicon video:

Evoke 1.2 - Basic Objects[6]

6 https://youtu.be/ENT5df3Yh8E

Display the required items in the Tracking panel

By default, all items are displayed in the Tracking panel. To display only the items needed for the current task, you can display or hide items in the Tracking panel, depending on their type.

To toggle the display of tracking items:

At the top right of the **Tracking** panel, click the relevant button(s):

The buttons toggle the display of:

- Basic objects (non-assigned)
- Characters (including assigned objects)
- Smart Objects (non-assigned)

Create basic objects

To represent your props, you create basic objects in Evoke.

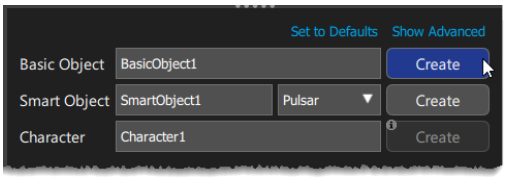
To create a basic object:

1. Place the prop with active markers attached in the volume, making sure that the markers are turned on.

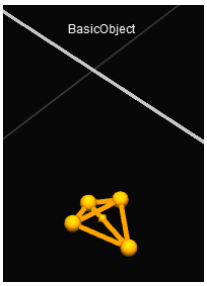
   > **Tip**
   > If you don't see anything in the 3D Scene, in the **View Filters**, ensure **Unlabeled Markers** is selected.

2. In Evoke, select at least four (five or more is recommended) unlabeled reconstructions.
   The orientation of the object depends on the order in which you select the reconstructions. The first is the root, the second is the axis it will point along, and the third is the Up vector for the object. If necessary, you can change this later (see Change a basic object's origin, page 64).

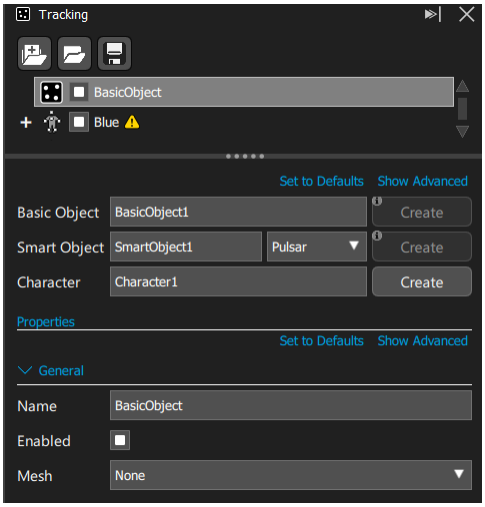
3. In the **Tracking** pane, click the **Create** button for a Basic Object.



A new basic object is created with labeled markers and sticks



In the **Tracking** pane, a new basic object (by default, called Basic Object #) is displayed.



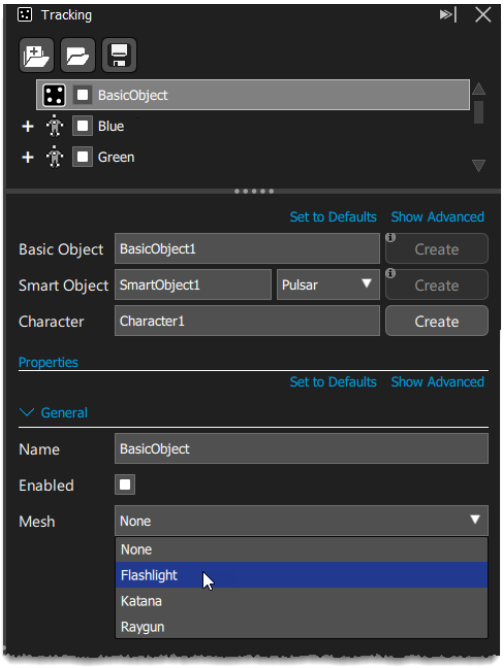
> **Tip**
> If your new object has a warning symbol (yellow triangle) next to it, to view information about the issue, hover the mouse pointer over the symbol. For more information, see Understand object evaluation.

4. To rename the basic object, in the tree at the top of the Tracking pane, either:
   - Double-click the basic object to edit the name; or
   - Right-click it and then select **Rename**; or
   - Select the basic object and press F2; or
   - Select the basic object and in the **Properties** pane below, click the **Name** field.
5. Enter a unique new name for the basic object.

Add a mesh to a basic object

Adding a mesh to basic objects enables you to better visualize them in the Workspace to check that they are correctly aligned and that they give the user experience that you want.

1. In the **Tracking** pane, select the basic object to which you want to add a mesh.
2. In the **Properties** pane, from the **Mesh** list, select the required mesh. The mesh is an FBX file with the mesh skinned to a Root bone. You can use the installed meshes, or your own FBX files, saved to *C:\Users\Public\Documents\Vicon\PropMeshes*.

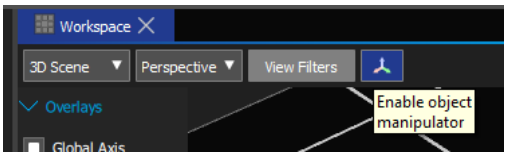


Your mesh is displayed in the 3D workspace.

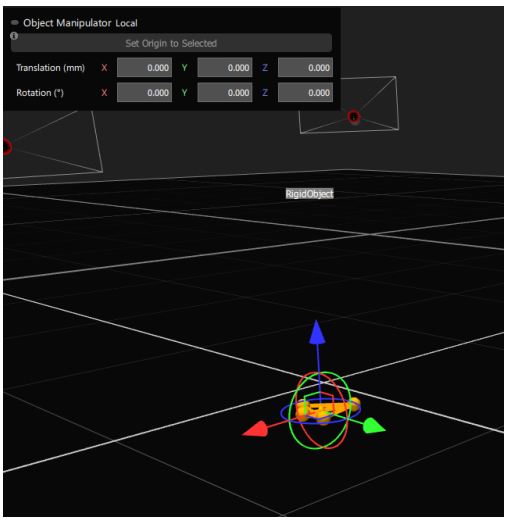
Change a basic object's origin

The object manipulator enables you to specify a basic object's origin (ie, the center of the physical object in relation to the marker pattern that is tracked by Evoke).

1. To activate the object manipulator:
   a. Press the space bar or click the **Pause** button at the bottom right of the **Workspace** to pause Evoke.
   b. Select the basic object.
   c. In the **3D Scene** view, click the object manipulator button.



2. In the Workspace either:
   - Drag the manipulator to move or rotate the object to the required location; or
   - Enter values in the **Translation** and/or **Rotation** fields;



When you resume streaming, the object's orientation is updated.

> **Tip**
> To scale the Manipulator, on the numeric keypad, press + (scale up) or - (scale down).

Note that the manipulator operates in local or global space (corresponding to the object coordinate system or the world coordinate system respectively). To switch between the two, click the icon to the left of the **Object Manipulator** text.

Export a basic object as a VSK

To export a basic object as a Vicon Skeleton (VSK) file:

1. On the **Tracking** pane, right-click on the basic object and then select **Export**.
2. In the **Export** dialog box, browse to or enter the location for the object.
3. Save the VSK.

Create a custom calibration object

You can use a Vicon Active Wand to set up your volume coordinate system quickly and easily (see Set the volume origin, page 83). However, using a larger calibration object (for example, markers embedded in the volume floor and wall) can improve calibration stability and consistency over time.

You can create and export a custom calibration object from any basic object as described in Create basic objects, page 60 and Export an object as a VSK, page 65.

When you export the VSK, save it to your calibration objects folder. The default location for this is:

C:\Users\Public\Documents\Vicon\CalibrationObjects

You can now use it to set the origin of your volume:

1. On the **Camera Calibration** tab, make sure the Advanced options are displayed (click **Show Advanced** at the top right if necessary).
2. Ensure the **Set Volume Origin** section is expanded.
3. From the **L-Frame** list, select the object that you created.
4. Ensure **Perform Rescale** is selected.
   This ensures that the marker distances in the L-Frame object are used for volume scaling.
5. Click **Start Set Origin**.
   The button displays **Set Origin** and is unavailable until Evoke has enough data to set the origin, when the button is enabled.
   If issues are detected, Evoke displays a message in the **Set Origin** button's tooltip to help you solve the problem. For more information, see About collecting frames, page 85.
6. Click **Set Origin**.
   The system scale is adjusted to provide improved calibration stability and consistency.

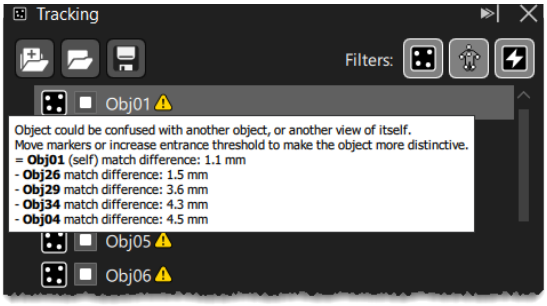
Understand object evaluation

Object evaluation is a new feature that runs automatically in the background to compare similarity in the pattern of markers for each object that is enabled in the Tracking panel with every other enabled object.

When you create, load or change objects, it is important to avoid placing markers in the same position on different similar objects, as this makes it difficult for Evoke to distinguish between the objects. This can result in tracking problems where objects may be confused with each other, leading to object swaps.

To help you to avoid this issue, Evoke evaluates the objects and if there is a high likelihood of confusion between objects, it flags them with a yellow warning triangle on the Tracking panel.

To view more information about the issue, hover the mouse pointer over the warning symbol.



For details about the way Evoke evaluates the similarity of objects and the actions you may need to take, see these topics:

- Why objects need to be evaluated, page 68
- How does object evaluation work?, page 68
- Control object evaluation, page 74

> **Important**
> If you have many objects or objects with many markers, evaluation can take a long time, so when you are running a live experience, we recommend that you turn off object evaluation (see Control object evaluation, page 74).

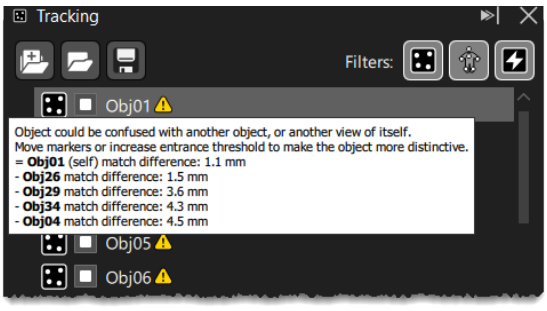
Why objects need to be evaluated

The Evoke tracking algorithm requires all objects to have a unique pattern of markers (with the exception of template objects in a proximity group). This enables Evoke to start tracking immediately, in the same frame that an object becomes visible, which is particularly helpful in cases of frequent occlusion, and results in more consistent tracking with fewer dropped frames. However, the requirement for uniqueness can be difficult to fulfill without an assessment when all possible orientations of the objects are considered, particularly when you have large numbers of similarly sized objects.

Smaller objects are also much more difficult to make distinctive, as the average distance between markers is proportionally smaller. The ability to resolve markers in distinct patterns depends on the camera resolution, and the typical distance of objects from the cameras. A volume with low camera density will generally require larger objects (with proportionally large marker patterns) to compensate.

How does object evaluation work?

Evoke's object evaluation feature compares each object that is enabled in the **Tracking** panel with every other enabled object, and displays a warning when a match between a pair of objects is detected. The warning tooltip indicates which other object (or objects) that the current object may be confused with, and the match distance (in millimeters). The match distance is the approximate average displacement between the object markers, when the two objects have been aligned to the closest matching pose.



A match distance of 0 indicates that the object pair is identical (or to be more precise, the fraction of markers specified by the object's Entrance Threshold, page 69 is identical to a pattern of markers on the other object). Conversely, a large match distance indicates the objects are easy to distinguish, and unlikely to be confused.

> **Evaluation of Smart Objects**
> Smart Objects with a single smart cluster are not evaluated against other Smart Objects with a single smart cluster. The smart cluster (e.g. Pulsar) markers have already been designed to offer the maximum match distance between different marker patterns, and there is instead a warning if the same marker pattern has been specified for two different smart clusters.
> Composite Smart Objects have special rules for starting tracking. In addition to the normal Entrance Threshold, all the markers on at least one of the smart clusters in the composite Smart Object must be detected. This is also accounted for in object evaluation.

Why is the Entrance Threshold important?

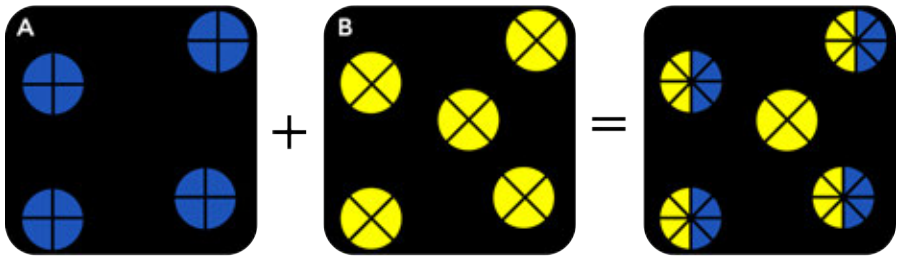
The **Entrance Threshold** setting is a value between 0 and 1, which represents the minimum proportion of markers that must be visible to the cameras before the object is booted, where 1 is 100% of the object's markers.

The Entrance Threshold has a significant effect on object evaluation. The Entrance Threshold specifies the fraction of object markers that must be detected in order to start tracking. If the Entrance Threshold is low, it is easier to start tracking the object, but more likely that the tracking can be initialized incorrectly. For example, if you have a 10-marker object with an Entrance Threshold of 0.3 or lower, tracking can start as soon as any 3 markers are detected in the correct pattern (3 is the minimum number required to track). This means that the object could be confused with any other object that has the same or similar pattern of 3 markers, regardless of how different they are as a whole. In general, we recommend that you set the Entrance Threshold to 1 (require all object markers), unless the object geometry makes it difficult for the system to detect all the markers simultaneously. This may be the case for an object resting on the ground, for example: any markers on the underside may not be visible to the cameras. For details on how to change the Entrance Threshold setting, see Resolve issues with object similarity, page 72.

Examples of object similarity

The Entrance Threshold for all objects in these examples is 1, unless otherwise specified.

Different number of markers

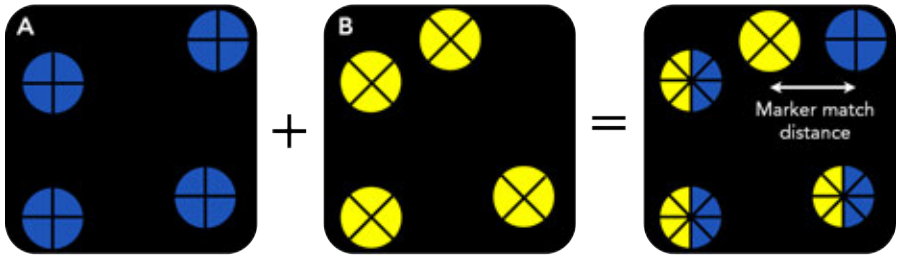


Object A has 4 markers, object B has 5, both have Entrance Threshold 1:

- Object A matches object B, match distance = 0 (The match distance is the approximate average displacement between the object markers, when the two objects have been aligned to the closest matching pose.)
- Object B doesn't match object A (which doesn't have enough markers to meet the Entrance Threshold for object B).

If we reduced the Entrance Threshold to 0.8 for object B, it would then match object A identically (with match distance 0).

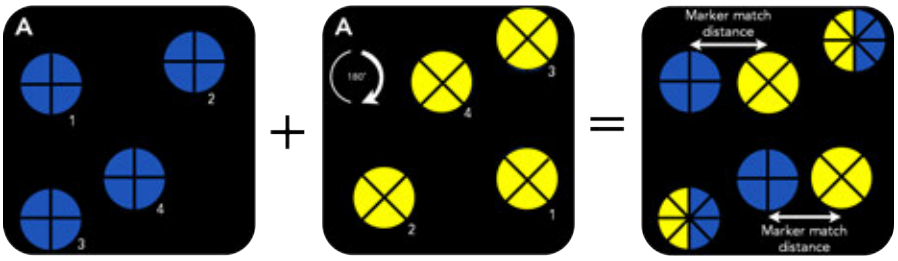
Moved marker



Object A and object B both have 4 markers, one of the markers on object B has been moved. The distance between the markers is the marker match distance, as indicated on the above diagram.

- Object A matches object B with an average match distance of the marker match distance/4 (3 markers match exactly, with distance 0).
- Object B matches object A with the same average match distance.

Self-similarity (rotational symmetry)

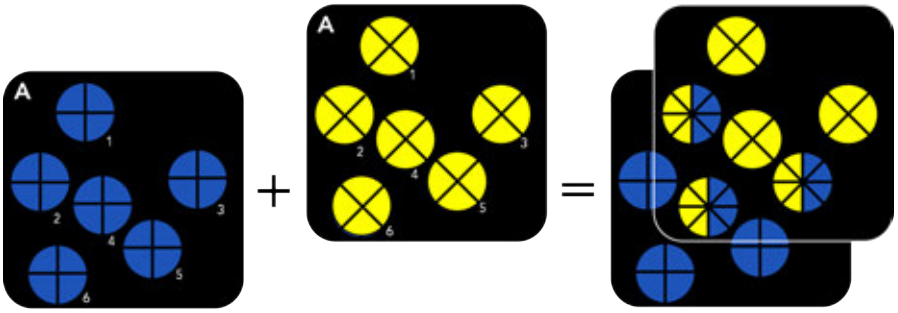


Here we have rotated object A by 180° clockwise.

- Object A matches itself with average match distance of the marker match distance/2 (2 markers match exactly, with distance 0).

Rotational symmetry is a problem because it means Evoke will have difficulty in determining the correct orientation of the object. You would notice this as unpredictable rotational errors or 'flipping' the object when starting tracking.

Self-similarity (repeated marker pattern)



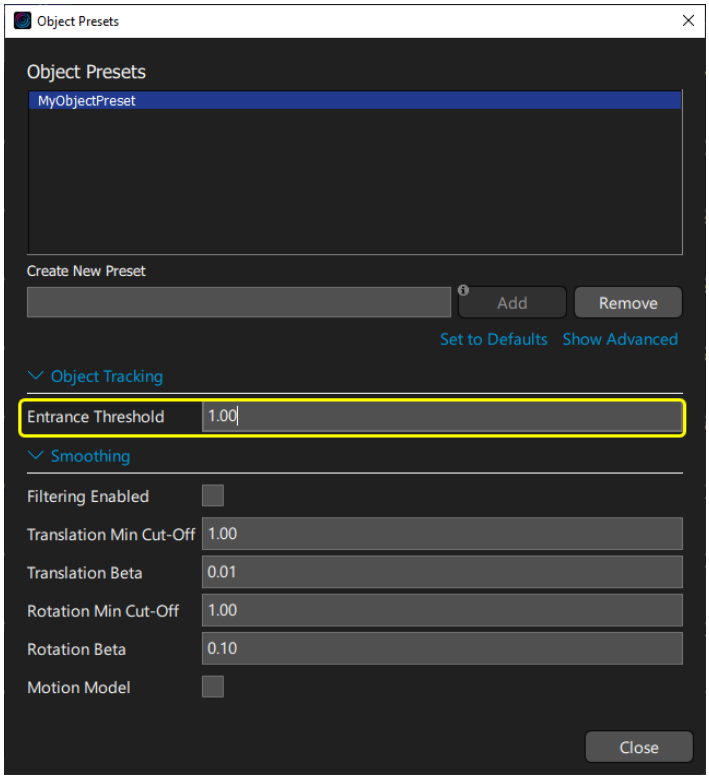
Here object A has 3 markers repeated in the same pattern - (1, 3, 4) matches (2, 5, 6) exactly.

This is not necessarily a problem if Entrance Threshold is set to 1, but if Entrance Threshold is set to <= 0.5 then Evoke could start tracking from either pattern subset, and cannot distinguish between the two. You would notice this as an unpredictable translational error when starting tracking.

Resolve issues with object similarity

To resolve issues with object similarity, try the following steps:

1. The likelihood of object confusion is significantly increased if objects have low entrance thresholds, so set the Entrance Threshold high, where possible. The Entrance Threshold is set in two places:
   - The overall entrance threshold (**Processing** tab > **Object Tracking** section > **Entrance Threshold** setting) sets the default entrance threshold for all objects. Normally, we recommend that you leave this at its default setting. You can override this setting for individual objects by using an object preset, as described next.
   - To change the Entrance Threshold for a selected object, either select an object preset with the required Entrance Threshold, or create a new object preset with the required Entrance Threshold and select it. To do this:
     i. On the **Tracking** panel, ensure the object is selected,
     ii. On the **Properties** tab below, in the **General** section, on the right of the **Object Preset** list, click **Manage**.
     iii. In the **Object Presets** dialog box, enter a name for the new preset, click **Add** and ensure the new preset is selected.
     iv. In the **Object Tracking** section, specify the required **Entrance Threshold** (a value between 0 and 1, which represents the minimum proportion of markers that must be visible to the cameras before the object is booted, where 1 is 100% of the object's markers).



v. Ensure that the other settings are as required and close the dialog box.

vi. In the **Tracking** panel, ensure that the object to which you want to apply the preset is selected.

vii. From the **Object Preset** list, select the preset that you created.

2. If objects may still be confused with one another, adjust the placement of one or more markers on the affected objects to make their patterns distinct.

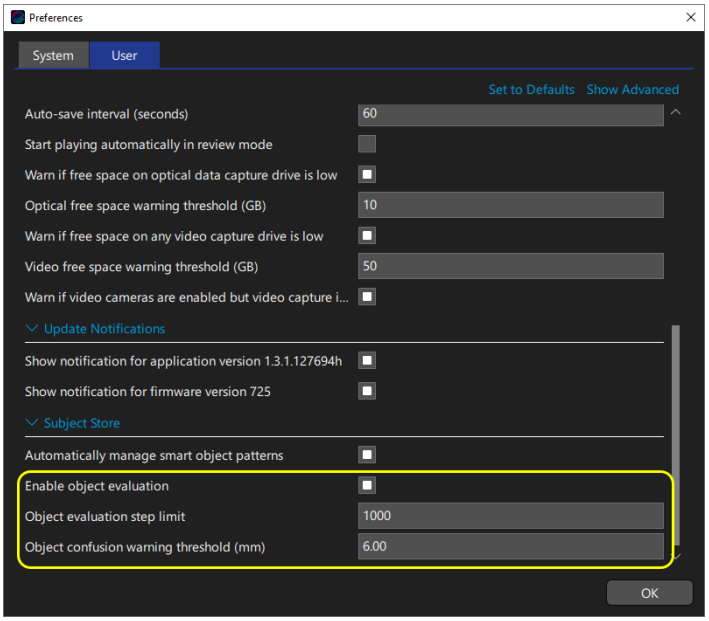
Control object evaluation

To turn object evaluation on and off or to change the sensitivity of the evaluation to object differences:

1. Open the **Preferences** dialog box (**Settings** > **Preferences** or Shift+P).
2. On the **User** tab, scroll to the **Subject Store** section.
3. You can change the following settings:
   - **Enable Object Evaluation** - Turn object evaluation on or off: the default is selected (on). When selected, each object is compared to all the others for possible confusion.

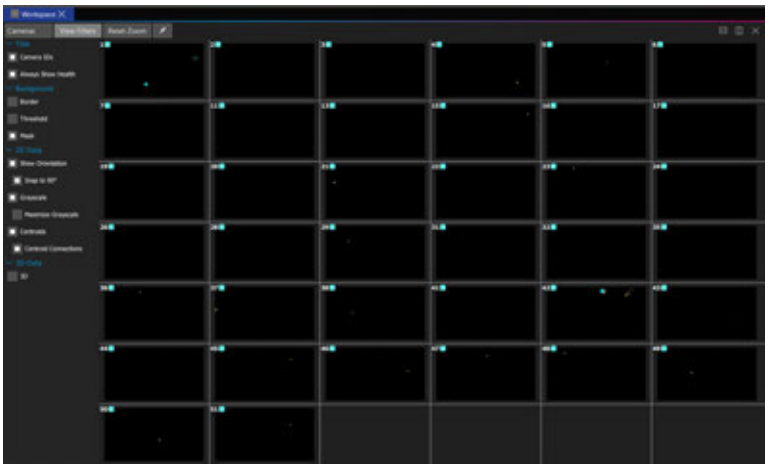
     > **Important**
     > If you have many objects or objects with many markers, evaluation can take a long time, so we recommend that, when you are running a live experience, you clear this check box to turn off object evaluation.

   - **Object evaluation step limit** - Maximum number of iterations (steps) that are expended in evaluating each object pair. Increasing this value may enable the evaluation to find more potential problems, particularly for objects with many markers, but at the cost of increased computation time. The default limit is 1000 steps.
   - **Object confusion warning threshold** - A warning is displayed if the match distance in millimeters between a pair of objects falls below this threshold. (The match distance is the approximate average distance that the object markers would have to move to exactly match the other object, or part of it.) You may want to adjust this setting so that it is higher than the typical level of noise (resulting in 3D positional jitter) in your volume. The default threshold is 6 mm.
4. Click OK to accept any changes and close the dialog box.


Preferences
System
User
Set to Defaults
Show Advanced
Auto-save interval (seconds)
60
Start playing automatically in review mode
Warn if free space on optical data capture drive is low
Optical free space warning threshold (GB)
10
Warn if free space on any video capture drive is low
Video free space warning threshold (GB)
50
Warn if video cameras are enabled but video capture i...
Update Notifications
Show notification for application version 1.3.1.127694h
Show notification for firmware version 725
Subject Store
Automatically manage smart object patterns
Enable object evaluation
Object evaluation step limit
1000
Object confusion warning threshold (mm)
6.00
OK

Calibrate cameras

When you first connect up your Vicon system and start Vicon Evoke, notice that on the **System** tab and in the **Cameras** view, icons give you feedback on the current status of the cameras. The cyan icon indicates that although the cameras are connected, they are not yet calibrated.



To calibrate your Vicon cameras, complete these procedures in order:

- Mask cameras, page 77
- Wand wave, page 81
- Set the volume origin, page 83
- Set the floor plane, page 86
- Autonumber cameras, page 89

Important
Before you begin, ensure that cameras have fully warmed up to a stable operating temperature (a minimum 30–60 minute warm-up period is recommended).

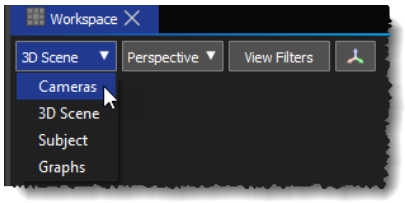
Mask cameras

You mask cameras during camera calibration to eliminate any unwanted reflections in the capture volume, so that they are not mistaken for markers by the cameras. Before you start masking, you can see these reflections represented by light pixels in the **Cameras** view. During masking, blue pixels are drawn in the **Cameras** view, enabling you to see how much of the view is masked.

> **Note**
> If your volume has windows or skylights, the natural light level will vary throughout the day, so reflections can be difficult to mask. Ideally, to prevent any natural light from entering the volume, cover any windows with curtains/blinds. If this is not possible, mask any windows and reflections in Evoke, either automatically (if it's sunny enough) or manually.

To automatically mask reflections:

1. Ensure you have disabled or turned off any Pulsars and props that use active markers. Note that, to start masking, at least one camera must be contributing data.
2. In the **Workspace**, select the **Cameras** view.



3. On the **System** tab, select all the cameras (SHIFT+click or drag or right-click and then click **Select All Cameras**).
4. On the **Camera Calibration** tab, click **Start Masking (All)**.
   The button displays **Stop Masking** and at the top of the workspace, **Auto Mask Active** and a flashing red circle is displayed.
   On the cameras, the status lights turn cyan while auto masking is in progress.

Evoke starts tracking the data visible to each of the connected cameras. Any camera masks created are displayed as blue cells in the **Cameras** views for affected cameras. If no data is visible to a particular camera, Evoke does not create any masks for it.

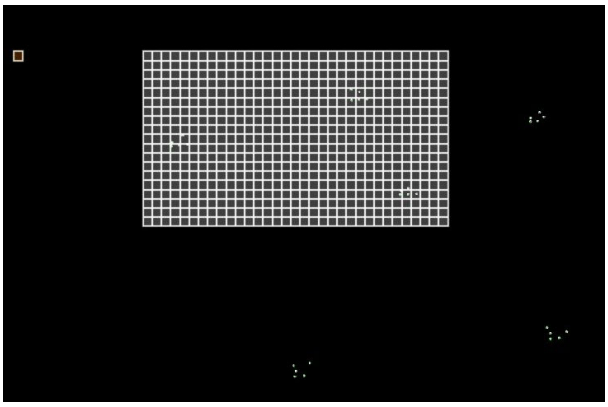
5. After about 5–10 seconds, click **Stop Masking**.
6. In the **Cameras** view for each camera, ensure that any unwanted reflections are eliminated. (Each view should either be completely blank or should contain some blue pixels.)

If the automatic camera masking that is described above does not eliminate all the unwanted reflections, you can manually remove any remaining reflections as follows.

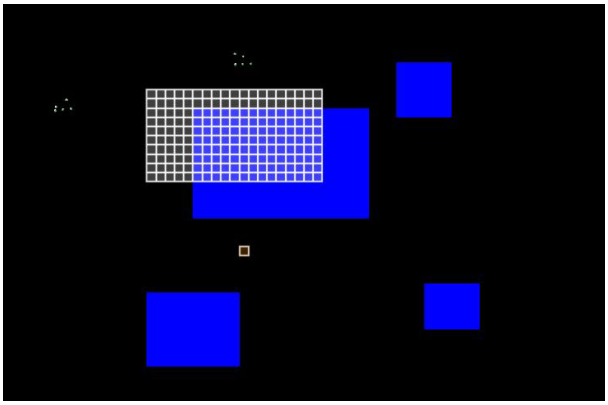
To define camera masks manually:

1. In the **System** tree or **3D Scene**, select a single camera.
2. On the **Camera Calibration** tab, ensure the advanced options are displayed (if necessary, click **Show Advanced** at the top right).
3. To begin editing the selected camera mask, on the **Camera Calibration** tab, click **Start Manual Mask Paint**.

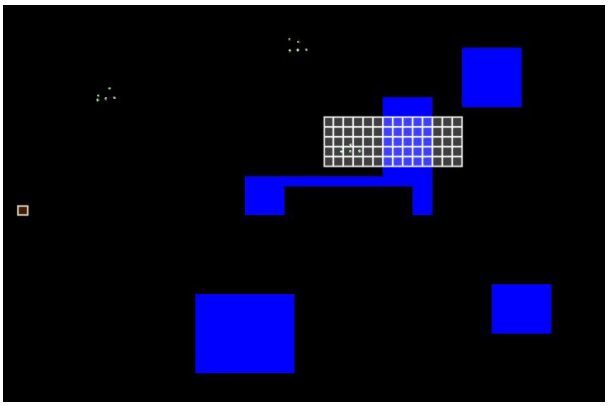
4. In the **Cameras** view:
   - Alt+drag to select a mask area



   - Press E to add the current selection to the mask



   - Press R to remove the current selection



5. To apply the changes, click **Stop Manual Mask Paint**.
   To revert to the original mask, click **Cancel**.

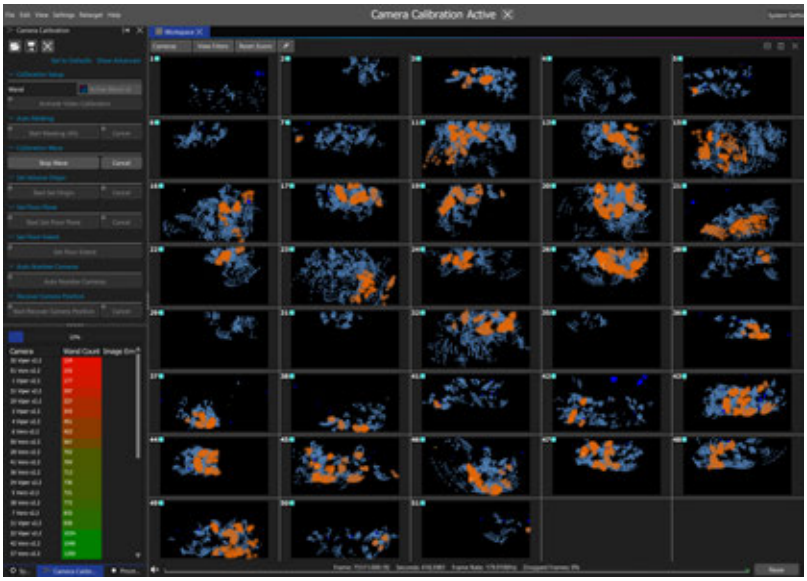
You can now perform a wand wave.

Wand wave

During camera calibration, after you have masked any reflections, you calibrate the cameras by waving a wand (the calibration device) throughout the volume to enable the cameras to capture movements over the whole area.

To perform a wand wave:

1. Ensure the Wand is in **Continuous** mode.
2. In Evoke, click **Start Wave (All)**.
   The button displays **Stop Wave** and in the menu bar, the text **Camera Calibration Active** is displayed, next to a flashing red circle.
3. Have someone wave the wand throughout the capture volume, covering depth as well as height, while you watch the **Cameras** views for all cameras to ensure you get full coverage. Ensure that the markers (LEDs) on the wand remain visible to all the cameras as much as possible while the wand is moved throughout the volume.
   As an indication of the age of the wand detection, to help you see where the wand is being waved and getting detections, the display in the each view changes from orange to blue.



On the **Camera Calibration** tab, notice that the **Wand Count** column changes from red to green as sufficient data per camera is captured.

This helps you concentrate on waving the wand for cameras that need more data.

> **Tip**
> By default, camera calibration stops automatically when each camera has seen enough of the wand to ensure calibration. To adjust this or turn it off, at the top right of the **Camera Calibration** tab, first click **Show Advanced**. To adjust the amount of data needed before the camera calibration stops, in the **Calibration Wave** section, change the value of **Auto Stop Minimum Wands**. To turn off the automatic stop, clear the **Auto Stop** check box. Note that if you do this, the **Wand Count** column does not change color, and you will have to estimate when sufficient data has been captured.

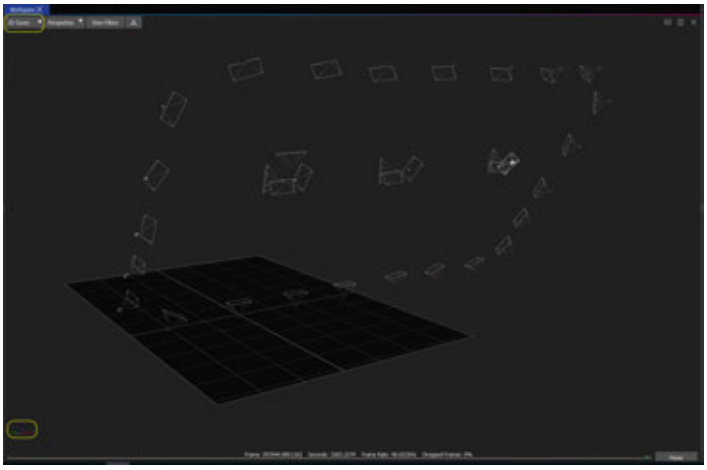
4. After the wand wave has stopped, in the **Image Error** column, in addition to displaying the values, Evoke grades the status of each camera between red (poor) and green (excellent), depending on how much the cameras see the wand.

In the volume, the status lights on the Viper cameras turn magenta and blink during calibration, becoming green and then blue when fully calibrated.

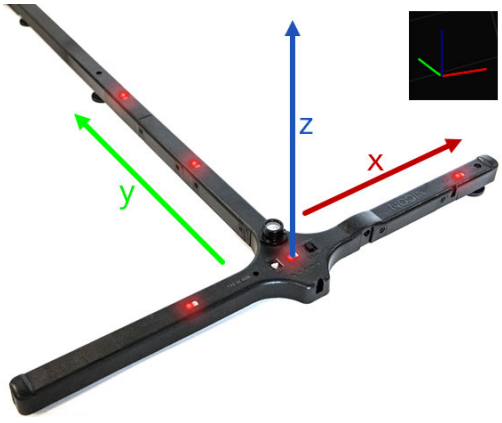
Set the volume origin

After you have captured a wand wave, you set the volume origin and axes so the cameras and volume in Evoke reflect the actual positions of the cameras in relation to the volume, as well as to each other.

1. To enable you to see the axes in relation to the volume, on the **Workspace** tab, change the view to **3D Scene**.



2. Place the calibration device on the volume floor in the position you want the volume origin to be and in the orientation you want the axes to be (reflected in the axes displayed in the **3D Scene** view).

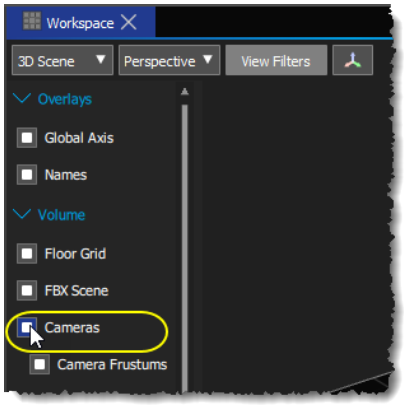


- Z axis: blue line
- X axis: red line
- Y axis: green line

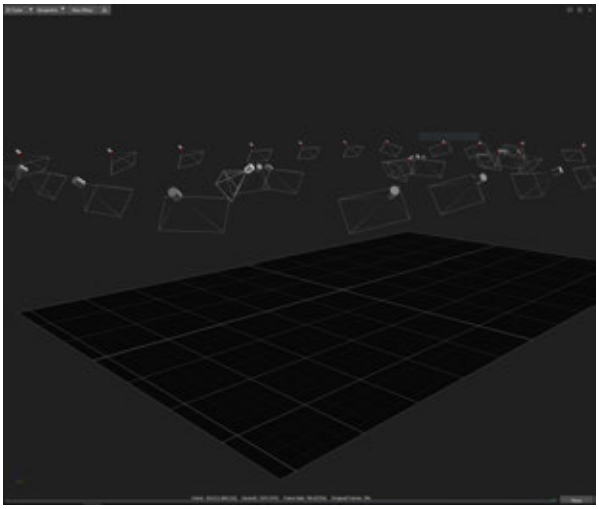
3. At the top of the **Camera Calibration** tab, in the **Wand** list, ensure the appropriate Active Wand is selected (normally Active Wand v2).

> **Tip**
> Using a Vicon Active Wand enables you to set up your volume coordinate system quickly and easily. However, using a larger calibration object (for example, markers embedded in the volume floor and wall) can improve calibration stability and consistency over time, particularly in larger volumes. For more information, see Create a custom calibration object, page 66.

4. Click **Start Set Origin**.
   The button displays **Set Origin** and is unavailable until Evoke has enough data to set the origin.
   If you need more information about why the button is unavailable, hover the mouse pointer over the button to view its tooltip. For more information, see About collecting frames, page 85.
5. Click **Set Origin**.
6. In the **3D Scene**, ensure that in the **View Filters** options, **Cameras** is selected.



In the **3D Scene**, **Perspective** view, all of the cameras shift as a group, so the origin of the volume is aligned with the wand.



About collecting frames

If Evoke detects an issue with setting the origin, the tooltip for the **Set Origin** button displays the following information in the format 'Collecting frames (*reason origin can't be set*)' so that you can take the appropriate action:

Tooltip text (reason)	Action
L-Frame motion detected	Ensure the calibration object remains motionless while you are setting the origin.
No labeled data	Nothing is being tracked in the volume. Ensure the wand is switched on and is in Continuous mode.
L-Frame not tracked	The chosen calibration object was not detected. Ensure the wand is switched on and in Continuous mode.
Multiple objects detected	Remove any extra objects that are being tracked in the volume.
Insufficient frames	The calibration object hasn't been tracked in enough frames yet. Check that the L-frame is being tracked consistently.

Set the floor plane

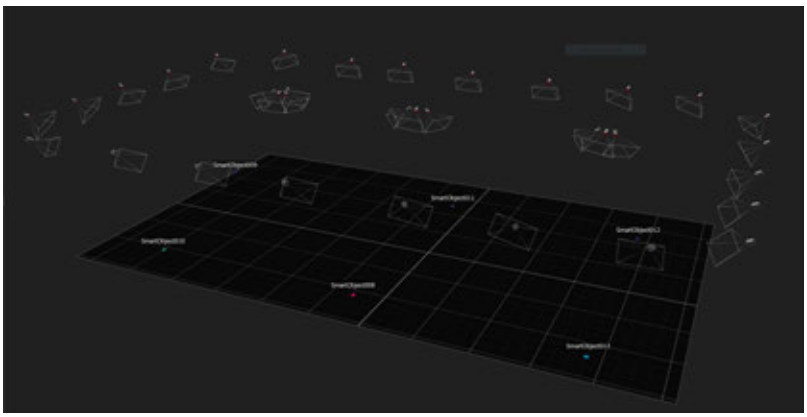
The final stage in calibrating your Vicon cameras is to set the floor plane, using Pulsars in the volume to automatically define it.

The position of the virtual floor that is derived during setting the origin is extrapolated from the position of the wand in relation to floor of the volume. As the wand is a small object compared with the size of the volume, any slight discrepancy from the wand being level has a large effect over the rest of the volume when you set the origin. To account for any discrepancy, you set a floor plane, which takes a much larger area into account, so that the virtual floor lines up correctly with the actual floor plane.

To set the floor plane:

1. Ensure you have completed the rest of the camera calibration procedure and set the origin (see Set the volume origin, page 83).
2. Turn off the wand or remove it from the volume.
3. Place a minimum of four Pulsar clusters across the volume floor.
   (If you need to change the default 7 mm floor plane setting (the setting for Pulsars is normally 14 mm), see Adjust the Set Floor Plane settings, page 88.)
4. On the **Camera Calibration** tab, click **Start Set Floor Plane**.
   The button displays **Set Floor Plane**.
5. After a few seconds, click **Set Floor Plane**.
   In the **3D Scene**, ensure that in the **View Settings** options, **Cameras** is selected.

In the **Perspective** view, notice that the cameras shift as a group slightly along one or more rotation axes to better reflect an average of the markers scattered across the floor, taking into account any offsets that you specified.

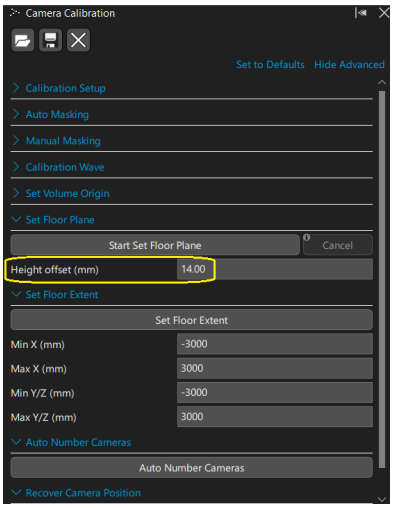


Tip
To more accurately visualize the size of your volume in Evoke, you can change the size and shape of the floor grid. To do this, on the **Camera Calibration** tab ensure the Advanced options are displayed (click **Show Advanced** at the top right), and in the **Set Floor Extent** section, change the values (in mm) to produce the required result.

Adjust the Set Floor Plane settings

If you need to change the default floor plane setting, to enable you to set the floor plane accurately, click **Show Advanced** and change the **Height offset** value to an appropriate value.

The **Height offset** is the amount (in mm) by which to adjust the floor plane (the default is 7 mm). Because Evoke finds the centers of the Pulsars, set a **Height offset** that accounts for the size of the Pulsars (normally 14 mm). If the Pulsars include a base, take this into account in your calculations.



Autonumber cameras

The **Auto Number Cameras** feature numbers the currently connected Vicon cameras in ascending order, according to their position in the volume. You may want to do this after you calibrate your Vicon system, so that your cameras are logically numbered before you begin tracking.

Automatic numbering starts with the camera that is furthest from the volume origin. The cameras are then numbered in a clockwise direction around the volume. If your cameras are positioned at different levels, the cameras in the level that contains the most cameras are numbered first.

To automatically number Vicon cameras:

1. Ensure that the cameras are positioned as required, and that you have calibrated the cameras and set the volume origin.
2. To enable you to check the camera numbering for all cameras, on the **System** tab, ensure that you can see the list of Vicon cameras.
3. On the **Camera Calibration** tab, in the **Auto Number Cameras** section, click **Auto Number Cameras**.
   The cameras are automatically numbered in ascending order, according to their position in the volume.
4. In the volume, check that the cameras are now numbered as required.

Scale calibration and set a fixed origin

After you've calibrated the system in the usual way, including setting the volume origin, page 83, for improved scaling across calibrations and a permanently fixed origin point, you can create a large custom calibration object from markers permanently placed across the volume. You can then use this calibration object, coupled with using the Perform Rescale option, for subsequent setting of the system origin. This improves calibration stability and consistency over time.

> **Note**
> If it is impractical to use permanently fixed markers in your volume, you can still benefit from following this procedure, but when you come to recalibrate, substitute a scaled wand object in place of the custom calibration object. Your results will not be as accurate as they would be with a large calibration object, but you will still benefit from the scaling involved. For details, see Step 2 of Recalibrate with the scaled wand and custom calibration object, page 95.

These topics explain how to do this:

- Set up the custom calibration object, page 91
- Scale the Active Wand and the custom calibration object, page 91
- Recalibrate with the scaled wand and custom calibration object, page 95

Set up the custom calibration object

To initially set up the custom calibration object:

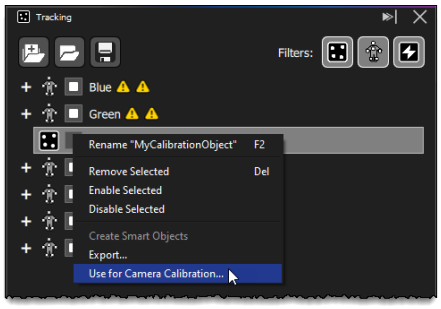
1. In the volume, place permanently static active markers to create a custom calibration object. Note the following points:
   - For best performance, **fix the markers to the floor** (provides both stability and the ability to spread across the volume).
     If this is not possible, position them on a wall or truss.
   - To guarantee a good measurement, locate the custom calibration object in an area of **good camera coverage**.
   - For best scaling performance, position the custom calibration object across the **central two-thirds of the volume** (to avoid variability introduced by gaps in coverage at the volume edges). If this is unfeasible, try to make the custom calibration object as large as possible, and position it towards the center of the volume.
2. As accurately as possible, measure the distances between points at two furthest edges of the object, and record the results. This can be a single measurement (eg, if the object's sides are equal), or two measurements across two different axes (eg, if the volume is not square).

Scale the Active Wand and the custom calibration object

Each time you change the custom calibration object:

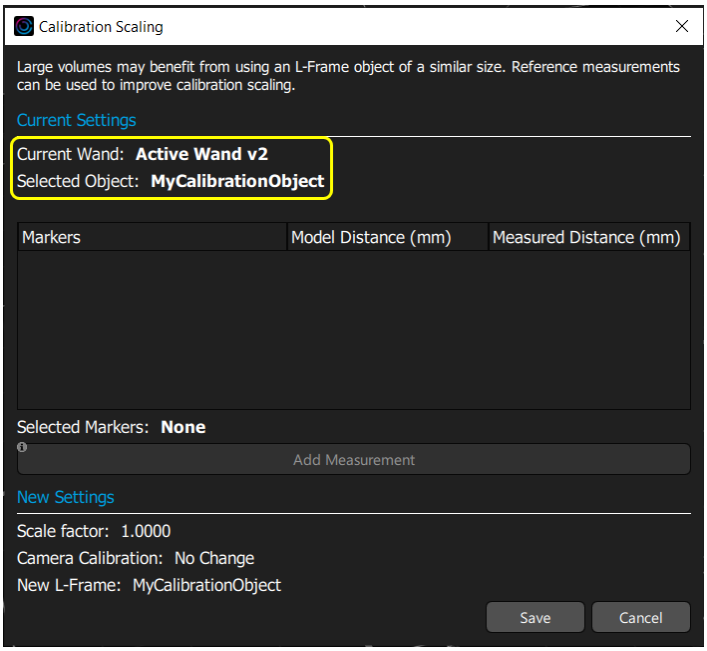
1. Calibrate using the normal process and standard *.vsk* files (see Calibrate cameras, page 76).
2. Set the origin using a standard calibration object (see Setting the volume origin, page 83).
3. In Evoke, create an object for the markers of the custom calibration object (for details, see Create basic objects, page 60).
   Do not save this object: it is important that the object that is used for this scaling is created from the current marker observations.

4. In the **Tracking** panel, ensure the custom calibration object is selected, right-click and then click **Use for Camera Calibration**.



The **Calibration Scaling** dialog box is displayed and you are alerted that the origin of the selected object will be set to the global origin (0,0,0) when you click **Save**.

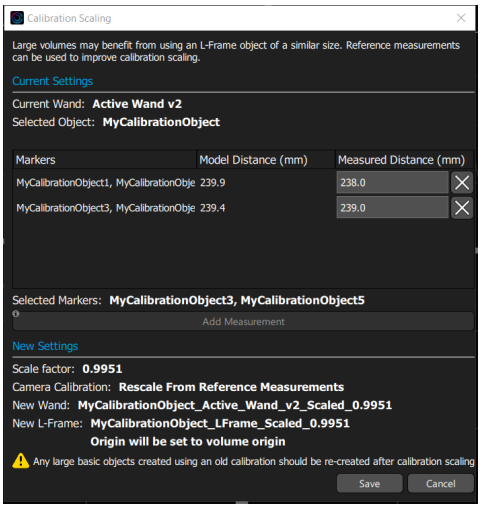
5. In the **Calibration Scaling** dialog box:
   a. In the **Current Settings** section, check that the **Current Wand** is the correct wand for your calibration, and that the **Selected Object** is also correct.



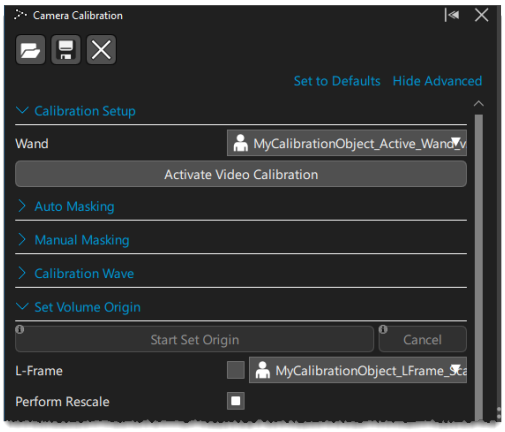
b. Ensure the custom calibration object is enabled, then, in the 3D Scene or Object view, select a pair of markers that correspond to a measurement that you took earlier, and click **Add Measurement.**
A line is added to the table containing the Model Distance, which is the distance between the markers as measured by the current Vicon calibration

c. In the **Measured Distance** column, enter the distance that you measured in millimeters.

d. For each set of measurements you have taken from the object, repeat steps b and c.
In the **New Settings** section, the scale factor that will be applied is displayed, together with the output names for the scaled wand and calibration objects.



e. If you are happy with this measurement, click **Save** to exit the dialog box.
A scaled wand and a calibration object file are saved by default to *C:\Users\Public\Documents\Vicon\CalibrationObjects* and are displayed in the Wand and L-Frame lists on the **Camera Calibration** tab.



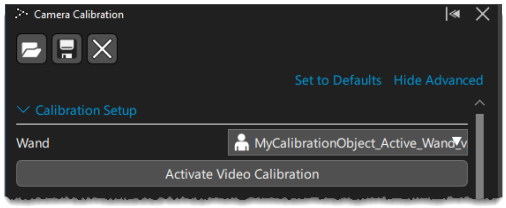
The current calibration is also scaled by the same factor, so you do not need to recalibrate after this operation.

6. In future, to obtain an accurately scaled volume, use the new scaled wand and calibration object when performing the calibration and set object operations (see Recalibrate with the scaled wand and custom calibration object, page 95).

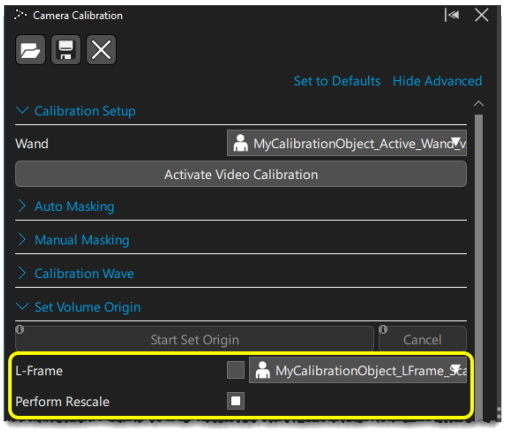
If you do not add measurements, the object is saved as a calibration object with no scaling modification, and no scaled wand is saved. You can still use the new calibration object for set origin operations, including rescaling. This will provide consistency of scale across set origin operations in different calibrations, but the volume will not be scaled to verified external measurements.

Recalibrate with the scaled wand and custom calibration object

1. Calibrate using the normal process, but in the **Wand** field, select the new scaled Active Wand *.vsk* file.



2. For the best results, set the origin using the normal process, but:
   - In the **L-Frame** field, select the scaled custom calibration object.
   - Ensure Perform Rescale is selected.



This ensures both a correctly scaled volume according to the measurements that you recorded of the custom calibration object; and also a permanently fixed origin, based on the static objects.

> **Tip**
> If you can't use permanently fixed objects in your volume, in Step 2 above, substitute the scaled wand object in place of the custom calibration object. Your results will not be as accurate as they would with the custom calibration object, but you will still benefit from the scaling of the wand.

3. Validate the new calibration by repeating the measurements from Step 5 of Scale the Active Wand and the custom calibration object, page 91. The measurements in Evoke should now closely align to the real world measurements of the fixed objects, as recorded in Step 2 of Set up the custom calibration object, page 91.

Check camera positions

Evoke enables you to check whether any cameras have moved by comparing the current camera positions with a snapshot (baseline) of camera LEDs from the previous Evoke session. It also provides information on when a re-calibration is required.

When you first start using your Origin system, it is advisable to run a calibration assessment daily, before use. When you have become familiar with how much the rig shifts or cameras get knocked, you may be able to reduce the frequency of this check.

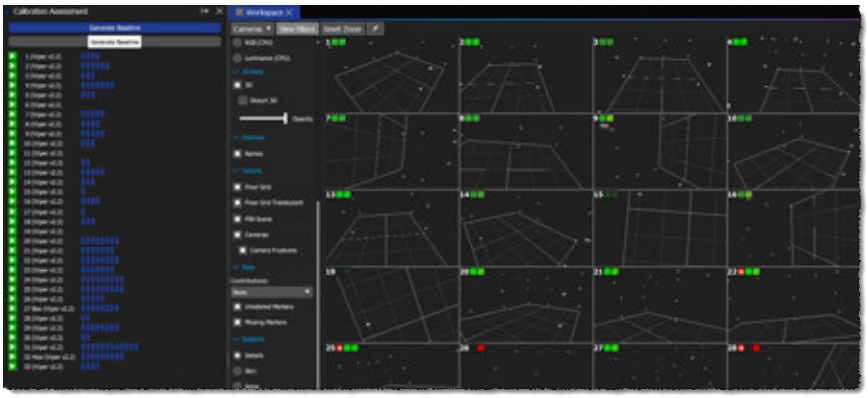
- Generate baseline, page 98
- Assess current calibration, page 99
- Understand automatic recovery of moved cameras, page 103

Generate baseline

> **Important**
> Before you begin, ensure that cameras have fully warmed up to a stable operating temperature (a minimum 30–60 minute warm-up period is recommended).

To generate information on the cameras' current positions:

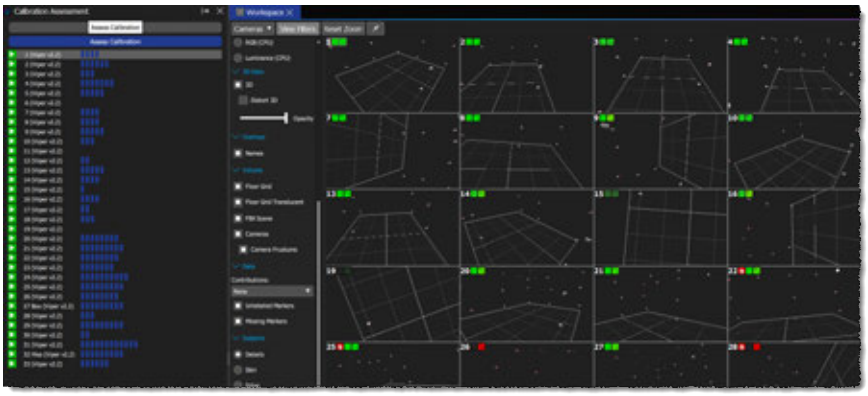
- On the **Calibration Assessment** tab, click **Generate Baseline**.
  Blue marks are displayed next to most cameras. Each mark represents the LEDs of the other cameras that can be seen by the camera.



Assess current calibration

After you have generated a baseline from which the camera positions can be assessed, you check for camera movement. To do this:

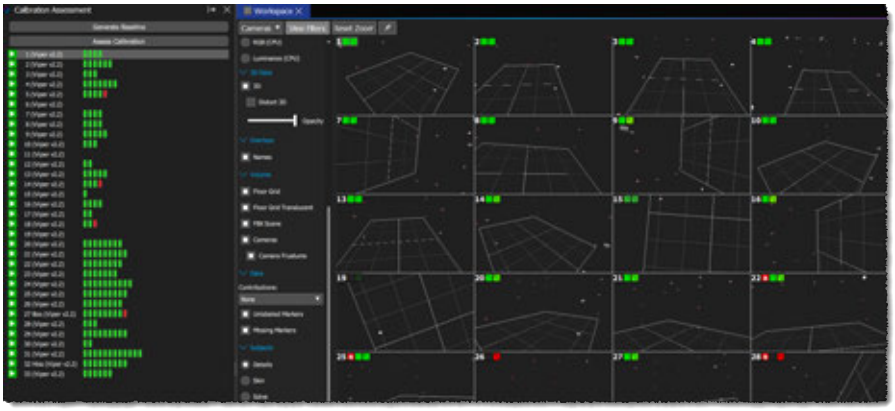
- On the **Calibration Assessment** tab, click **Assess Calibration**.



On the **Calibration Assessment** tab, most or all of the blue marks turn green.

Green marks indicate that the camera position is unchanged since the last calibration.

Red marks indicate that something has changed in the calibration, either due to a bump or knock, or because of environmental drift.



Depending on whether you have automatic bump healing selected (the default), you may need to take action to correct any changes.

- Enable automatic recovery of moved cameras, page 100
- Enable automatic bump healing in passive systems, page 101
- Manually recover moved cameras, page 102

See also Enable automatic bump healing in passive systems, page 101.

Enable automatic recovery of moved cameras

When you run **Assess Calibration**, Evoke may indicate that a camera has been moved. If a camera has been bumped enough to move its position, the move is detected, so the LEDs flash red on the camera, and the **System** pane and **Cameras** view display a bump icon to the right of the camera name.

To enable the recovery of a moved camera:

1. On the **Processing** tab, scroll to the **Camera Healing** section and make sure **Enable Auto Bump Healing** is selected (it is selected by default).

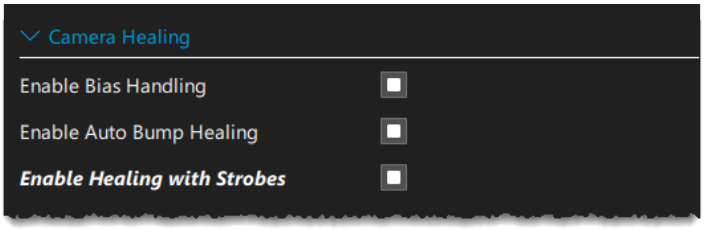
2. In the volume, have someone wave two of the Pulsars used to set the floor plane in front of the camera until its status light turns blue (or magenta if selected).
3. To generate a new baseline, on the **Calibration Assessment** tab click **Generate Baseline**. This creates a baseline that contains the updated information about the camera positions.

If auto-healing doesn't seem to be recognizing or recovering cameras that have moved, see Understand automatic recovery of moved cameras, page 103.

Enable automatic bump healing in passive systems

Auto-healing in systems with passive reflective markers is turned off by default.

To turn it on, in the **Camera Healing** section, select **Enable Healing With Strobes**.



> **Caution**
> If the environment has many infra-red sources or reflective surfaces that interfere with data that is being processed by the camera, auto-healing can adversely affect the calibration rather than rectifying it.

Manually recover moved cameras

When you run **Assess Calibration**, Evoke may indicate that a camera has been moved. On the moved camera, the LEDs to flash red and on the **System** tab and **Cameras** view, the bump icon is displayed to the right of the camera name.

If **Enable Auto Bump Healing** (see above) is not selected, you can recover a moved camera by using Evoke's **Recover Camera Position** option.

To recover a moved camera:

1. In the **Tools** pane, on the **Camera Calibration** tab, click **Start Recover Camera Position** and in the volume, wave two of the Pulsars in front of the affected camera.
   In the **3D Scene** and **Cameras** view, orange trails are displayed as the system determines the offset between the camera and the rest of the calibration.
2. When enough of the view has been covered (indicated by orange trails that thickly cover the affected camera view), click **Recover Camera Position**.
3. In the **Cameras** view, zoom in and check that the centroids are now green, and the red icon to the right of the camera names have gone.
4. To generate a new baseline, on the **Calibration Assessment** tab click **Generate Baseline** . This creates a baseline that contains the updated information about the camera positions.

Understand automatic recovery of moved cameras

When **Enable Auto Bump Healing** is selected, the system detects cameras that have stopped contributing centroids to the tracked objects in the volume if:

- The system health connection score for the camera is below 25% (to check this, at the top of the **Cameras** view, hover the mouse pointer over the Connection score icon ).
- Enough centroids are visible to this camera.

> **Important**
> For Enable Auto Bump Healing to work, at least 50% of the enabled objects must currently be tracked by the cameras.
> For this reason, disable any objects that can't be tracked (ie, any Pulsars that are charging, powered off or out of the volume).

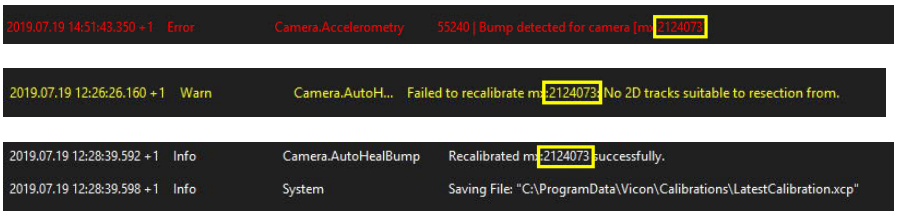
If these criteria are met, the system attempts to bump-heal (recover the position of) the camera. It collects data for a short time and if there is sufficient coverage in the cameras view and the system can find a solution, it applies a new calibration for this camera. This causes the system calibration file, to be re-saved to the following default location:

C:\ProgramData\Vicon\Calibrations\LatestCalibration.xcp

For successful automatic recovery of moved cameras, ensure that your system meets the following criteria:

- The cameras' internal parameters (focal length, radial distortion, etc) do not change.
- The cameras have achieved a stable operating temperature before calibration and before starting an experience.
- Objects are being tracked by a majority cameras. The easiest way to check this is to select all cameras (or no cameras), look in the **Cameras** view and in **View Filters** ensure **Centroids** is selected. Check that the centroids are green and that no calibration warning icons are displayed.
- The bumped camera(s) can see objects and they are generating 2D tracks.
- No unmasked camera strobes or reflective objects are present in the system. These may cause the system health of the camera to remain low and to trigger further recalibration.

If you need further information about which camera has moved, what may be causing an issue and whether it has been successfully recovered, check the Log (if it isn't displayed, on the **View** menu, select **Log**). The errors, warnings and messages will look similar to this:



In the above examples, the camera's Device ID is highlighted.

To find a camera's Device ID, on the **System** tab, select the camera, click **Show Advanced** in the **General** section, scroll to **Device ID**.

Prepare for a live experience

Before using Pulsars in a live experience, make the following checks:

1. Ensure Pulsars are paired with a Beacon in the volume (see Turn on the Beacon and enable pairing, page 13 and Pair Pulsars with a Beacon, page 20).
2. Ensure the Beacon is enabled and that on the **System** tab, no warning indicators are displayed.
3. Turn on the Pulsars and in Evoke, check the following indicators:
   - On the **System** tab, in the **Clusters** section, Pulsars are displayed.
   - The connection status for all clusters is green.
   - Battery levels are sufficient for the experience.
     If any of the Pulsar batteries is running low (indicated in the **System** tree by the low battery indicator), replace it before continuing (see Swap clusters, page 58).
   - No warning indicators are displayed.
   - Objects are tracked and labeled in the **3D Scene**.

> **Tip**
> If you need to find out which physical Pulsar is linked to its representation in Evoke, press the Pulsar's power button. It is then selected in Evoke and its status light blinks to indicate that it's currently selected.

Monitor your system

To monitor your system to ensure consistent performance, from Evoke 1.3 and later, you can remotely trigger the collection of data for a system health report while continuing to run a live experience. The report includes raw data for each camera, indicating how much data it has seen and how well it corresponds to the other cameras.

> **Important**
> If data is collected over too short a time period, these metrics may be highly variable for some cameras.
> It is strongly recommended that before analysis, you collect data over a sustained period of time, and during a standardized type of activity.

These API services are available for system health reporting:

- Start/stop/cancel report collection
- Get latest report

Information on the status of your system is accessible via the API.

An example implementation (*system_health_report.py*) can be found in the supplied Evoke API sample scripts.

To access the example, extract the files from the *.zip* file installed in this default location to a suitable folder:

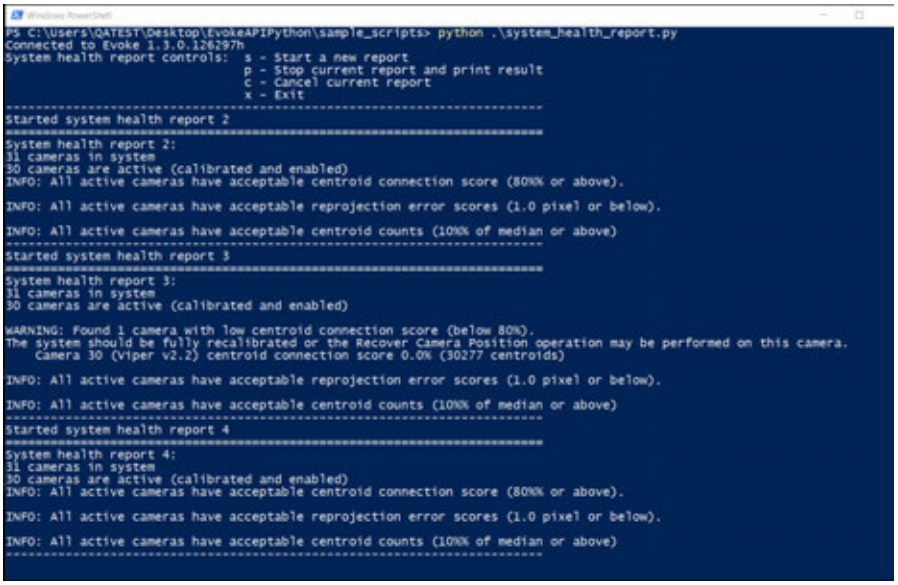
C:\Program Files\Vicon\Evoke1.4\SDK

You can find *system_health_report.py* in the *sample_scripts* subfolder.

The information generated for each camera includes:

- Centroid count: The number of 2D centroids seen by the camera
- Labeled centroid count: The number of centroids that correspond to a 3D model point on a tracked object
- Average reprojection error: RMS error between the observed centroid coordinates and the coordinates computed by projecting the 3D model point onto the image

The following reports were generated by running the example. Three reports were generated, showing the system running normally (Report 1), a camera having been moved (Report 2), the issue fixed and the system healthy again (Report 3):



For more information, see Vicon Evoke API & automation.

Set up character solving

Evoke provides human skeleton solving, enabling you to drive characters from clusters (Pulsars). The following topics explain the procedures that are related to character solving in the order you are likely to need them:

See also the Vicon video:

Evoke 1.2 - CFC Workflow[7]

7 https://youtu.be/TRUZVM2KaRY

Prepare for character solving

Before you begin, ensure you have created any necessary Smart Object templates:

- Smart Object templates are provided for Backstrap, Hat, OculusRift, HPReverb, HTCVive, and Pulsar, so you don't need to create these.
- Note that you can merge Smart Objects, page 51, which is useful for the head in particular.
- If you are using a backpack, you must create your own template (see Use Smart Object templates, page 54). When you do this, adjust the object origin offset (see Change a Smart Object's origin, page 57) to match the depth of the backpack, so Evoke knows where the participant's back is in relation to the markers that it tracks on the backpack.

Also ensure you have enabled Evoke to track Pulsars and props by creating the necessary Smart Objects and basic objects (see Create Smart Objects, page 40 and Create basic objects, page 60).

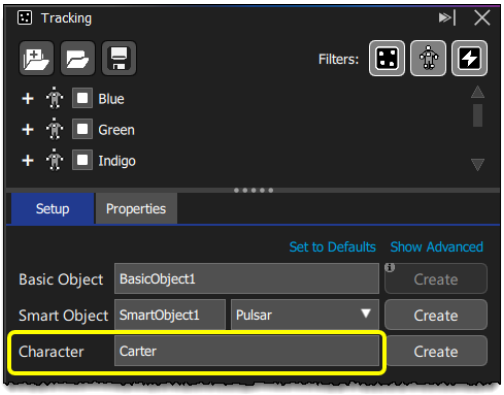
Object tracking only
If only object tracking is required, you can auto-assign clusters without character solving or retargeting.
To use this option, ensure the Advanced options are displayed and in the **Processing** panel, under **Characters From Clusters**, select **Disable Solving**.

Create characters

Create characters in Evoke to represent each participant.

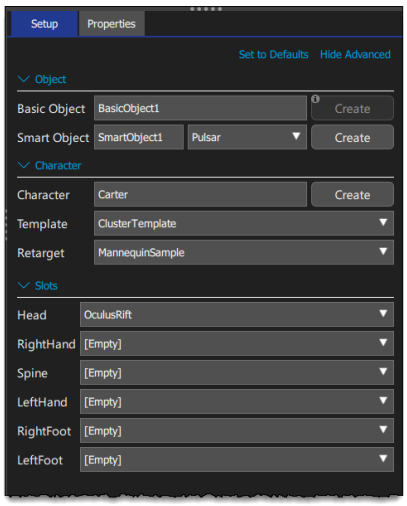
To create a character:

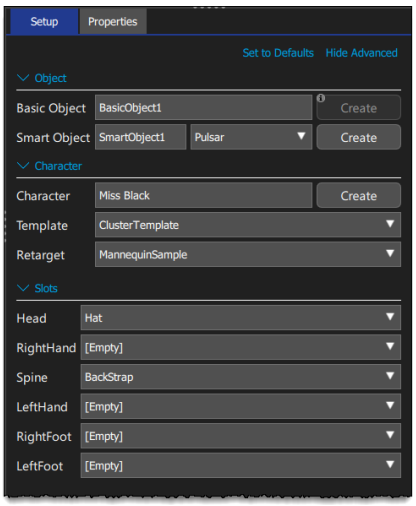
1. In the **Tracking** pane, on the **Setup** tab, enter the name of the character that you want to create from clusters in the volume.



2. At the top right of the pane below the Tracking tree, click **Show Advanced**, and in the **Template** list, ensure **ClusterTemplate** is selected.
3. From the **Retarget** list, select a retarget file. (You can use the supplied **ViconFemaleSample** or **ViconMaleSample** .)
4. In the **Slots** section below, click the first slot (**Head**), select either an **OculusRift, HPReverb, HTCVive** or a **Hat**, depending on the accessories that you are using. (Note that the supplied templates for supported headsets include two device slots for the front and top of the head.)

5. If you will be using two reference objects (typically Pulsar clusters for both the head and the spine), in the **Spine** slot, select the **Backstrap** template or your own template for the backpack you're using.
   As shown in the following examples, which show commonly used setups, you can leave the rest of the slots empty.





6. At the right of the the **Character** line, click **Create**.

Prepare the participants

1. Attach the Pulsars to the Vicon accessories, making sure you use the correct mounting plates for each accessory, normally:
   - Flexible mounting plates for the foot straps and chest strap
   - Rigid mounting plates for the gloves and hat
2. Make sure each participant is wearing the relevant Vicon accessories. At a minimum, these are:
   - Mocap hat (Pulsar attached with status light facing forward)
     or
     HMD clips attached to HMD (two Pulsars (front and top) attached - status light facing up)
   - Gloves (status light facing down)
   - Foot straps (status light facing forward)
   - Back strap, with Pulsar on back (status light facing up)
     or
     Backpack PC (status light facing up)

The following images show Pulsars correctly attached to a participant.

Assign objects and calibrate characters

After you have created the characters in Evoke, you can assign objects and calibrate the character for each participant in these ways:

- **Automated workflow**: For each character, assign one or two reference objects (typically these are the Pulsar clusters for the head or spine) to the correct slots, designating the remaining objects as auto-assignable. The remaining objects are automatically assigned to the correct slots in a single step when you calibrate the character (see Assign clusters and calibrate characters (automated workflow), page 114). For best results, use two reference objects, although this is not essential.
  This is normally the quickest and easiest way to assign objects and calibrate characters.

  > **Auto-assign for object tracking only**
  > If only object tracking is required, you can auto-assign clusters without character solving or retargeting.
  > To use this option, ensure the Advanced options are displayed and in the **Processing** panel, under **Characters From Clusters**, select **Disable Solving** .

  or

- **Manual workflow**: Manually assign each Smart Object (or basic object) to the correct slot, and finally, calibrate the character (see Assign clusters (manual workflow), page 121).

Assign clusters and calibrate characters (automated workflow)

Object labeling
In the volume, physically label the Pulsars that are linked to the reference objects (usually the headset and backpack), to indicate where to place them (for example, Player1_Head). You can leave the auto-assignable objects in a general charging area and place them onto any character and any limb.

To quickly assign objects to a character and calibrate it, use the following automated workflow.

Prepare objects for auto-assignment

Prepare the objects for auto-assignment, as described in the following steps, which are required for first time setup only.

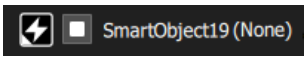
- Prepare the unassigned objects, page 115
- Prepare the reference object(s), page 116

Prepare the unassigned objects

In the 3D Scene, select the unassigned objects that you want to be auto-assigned and in the **Tracking** panel, on the **Properties** tab, select **Auto Assign Enabled**.

> **Tip**
> You may find it easiest to select the required unassigned clusters by first arranging them in a group in the volume, so you can select them easily.

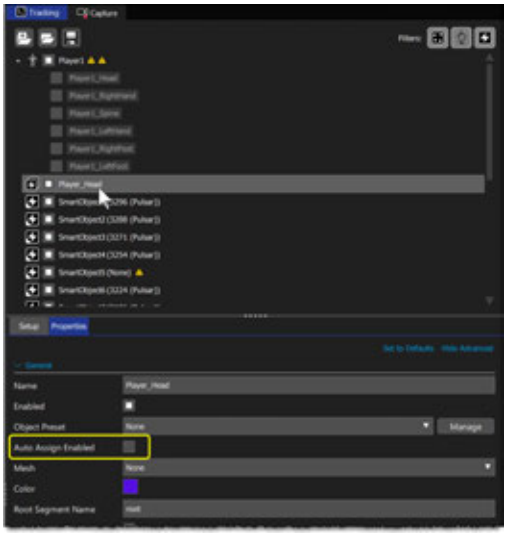
In the **Tracking** tree, when an object has **Auto Assign Enabled** selected, its icon displays a small triangle in the lower left corner, giving you a quick visual indication of the object's status.



Prepare the reference object(s)

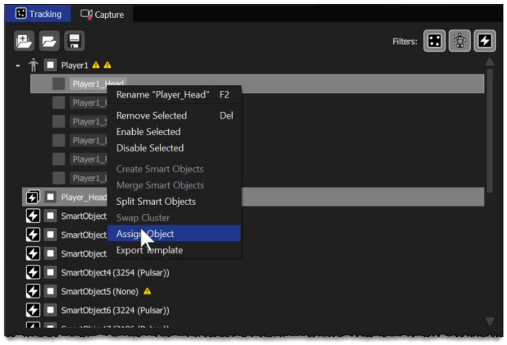
Assign one or (preferably) two objects (these can be Smart Objects, composite Smart Objects or basic objects) to the appropriate slots. Typically the reference objects are the Pulsar clusters for the head and/or spine. To assign them to their slots:

1. In the **Tracking** panel, select the reference object, display the Advanced options and on the **Properties** tab make sure that **Auto Assign Enabled** is cleared.

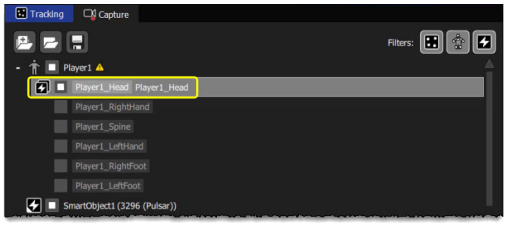


2. Ensure the objects are positioned in the volume in a way that makes it easy to tell which one is which (you may want to place them on a person or mannequin).

3. In the 3D Scene or in the Tracking tree, select the object and Ctrl+select its slot in the Tracking tree, then right-click either the object or the slot and select **Assign Object**.



4. Ensure the reference object is displayed in the correct slot for the character.



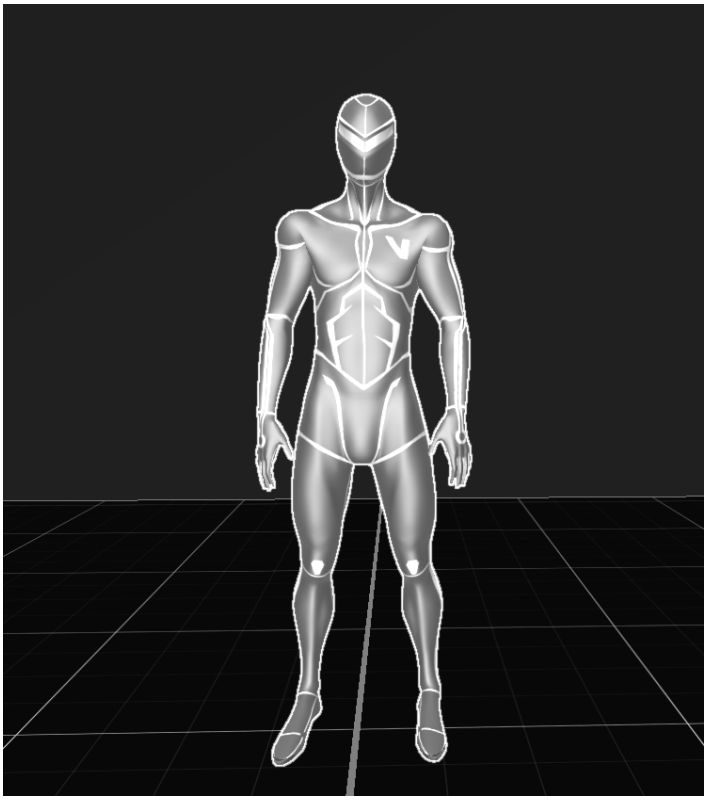
5. You can now auto-assign the remaining objects for the character, as described in Auto-assign objects and calibrate characters, page 118.

Auto-assign objects and calibrate characters

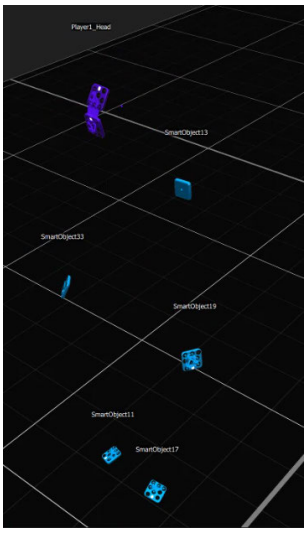
Ensure you have prepared both the unassigned objects and one or more reference object(s) (see Prepare objects for auto-assignment, page 114), and the participants (see Prepare the participants, page 112).

At the start of each experience, auto-assign objects and calibrate each character, as described in these steps:

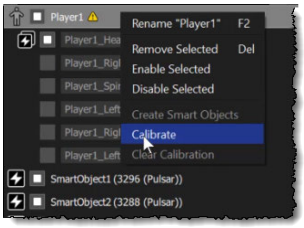
1. Get the participant(s) to stand in the capture volume in a neutral pose (known as an N-pose), which is a relaxed pose with the hands by the sides.
   The following image shows a character in a neutral pose (to show the pose clearly, the character has been calibrated).



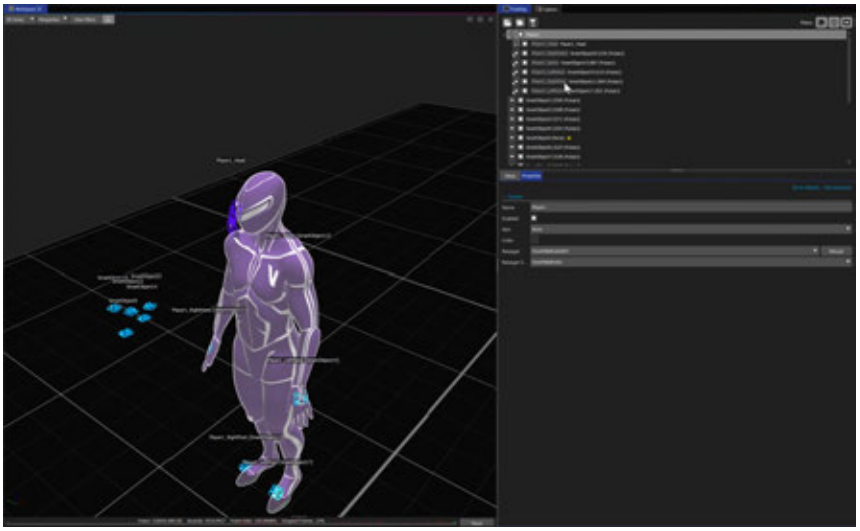
2. Ensure that the following objects are attached to each participant:
   - One or (preferably) two reference object(s). Typically these are the Pulsar clusters for the head or spine. For best results, use both, although this is not essential.
   - Auto-assignable objects on their other limbs (eg, hands and feet).



3. In the **Tracking** panel, select one or more participant's characters, right-click and then click **Calibrate** (or, to calibrate all characters, press the shortcut Shift-C).

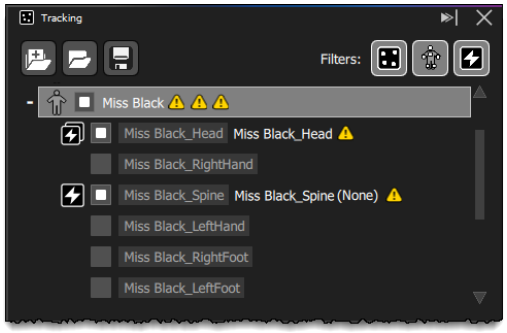


The objects are 'fitted', based on their positions in relation to the available and unpopulated slots, and assigned correctly.
If you're solving a character, it is displayed with a skeleton.

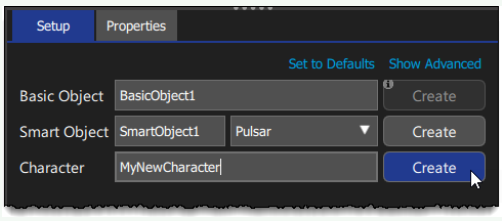


Assign clusters (manual workflow)

1. As for the automated workflow, set up the name, template and retarget file, and select the required templates for the slots (see Create characters, page 110).
2. At the right of the the **Character** line, click **Create**.
   The new character is added to the Tracking tree, with the populated slots as child nodes. Slots for which Smart Objects were selected are automatically named to match the character and slot (eg, Miss Black_Head in the following example). The yellow warning icons indicate that some information is missing.



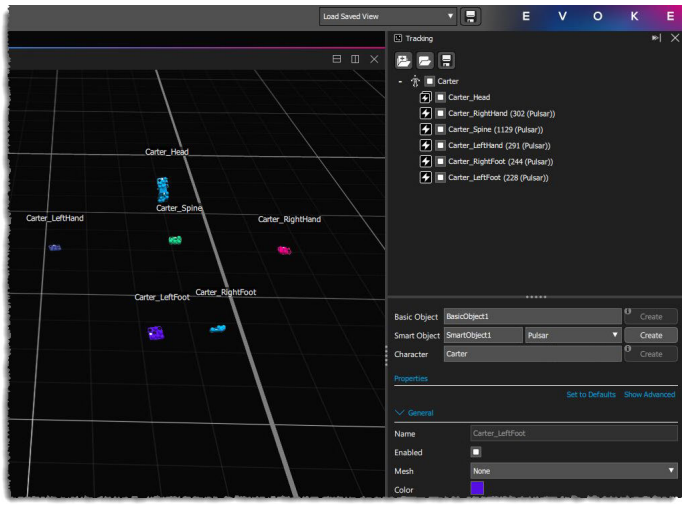
Tip
After you have created a character, you can create multiple new characters based on the same configuration without having to use the Advanced properties. Just enter a new character name and click **Create**.



You then assign objects to the empty fields as described in the next step.

3. Assign objects (these can be a single Smart Object, a composite Smart Object or a basic object) to the slots. To do this:
   a. Ensure the objects are positioned in the volume in a way that makes it easy to tell which one is which (you may want to place them on a person or mannequin).
   b. In the **Tracking** panel, select the relevant objects, display the Advanced options and on the **Properties** tab make sure that **Auto Assign Enabled** is cleared.
   c. In the 3D Scene, select an object and Ctrl+select its slot in the Tracking tree, then right-click either the object or the slot and select **Assign Object**.
   d. In the same way, assign the remaining objects to the appropriate slots.

   In the Tracking tree, the slots now all have the correctly assigned objects, which are also displayed in the 3D Scene.



4. If you have set up characters or composite Smart Objects, but have not linked Pulsars to them (eg, if you created Smart Objects from the supplied templates for some or all of the slots in a new character), link the new Smart Objects to the correct devices. To do this:
   a. Select an object and Ctrl+select the required Smart Object in the Tracking tree.
   b. If either of the Smart Objects is a composite Smart Object, from the sub-menu, select which device slot is to be affected by the swap.
   c. Right-click either object and select **Swap Cluster**.
   For more information, see Swap clusters, page 58.

When you have finished assigning objects, you can calibrate the character (see Calibrate characters, page 124).

To display or hide character slots and assigned objects in the Tracking tree, click the + or - symbol next to the character icon.

You can clear slots that you have assigned manually at any time after creation:

- To clear manual slot assignments, in the Tracking tree, right-click one or more character slots and then select **Unassign object(s)**.

Calibrate characters

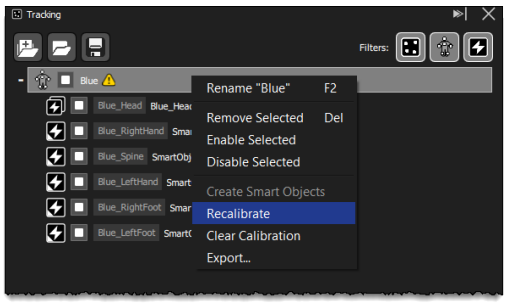
You must ensure each character is calibrated, but depending on the way in which you create your characters, the workflow is slightly different:

- If you use the automated workflow for object assignment and character calibration, page 114, both object assignment and calibration occur when you click **Calibrate**. Calibration assigns the clusters for which **Auto Assign Enabled** was selected to empty character slots; scales the source skeleton; and accounts for differences between the 'reference' position and where the clusters were actually placed on the participant, eg, if the participant was wearing heels, the clusters slipped, or the backpack straps were loose, etc (see About cluster calibration, page 126). Calibration also starts retargeting (if required).
- If you have used the manual workflow for object assignment and character calibration, page 121 and have therefore manually assigned the clusters to the slots, calibration does not perform any further cluster assignment and just scales the source skeleton to the participant and accounts for differences between the reference position and actual cluster placement (see About cluster calibration, page 126). Calibration also starts retargeting (if required).
  To calibrate your characters, complete the following steps.

To calibrate characters:

1. Ensure each person to be calibrated is standing in the volume in a neutral pose (known as an N-pose), which is a relaxed pose with the hands by the sides.
2. In the **Tracking** tab tree, select the character(s) that are to be calibrated and right-click.

3. In the context menu, click **Calibrate** (or to calibrate all characters, press the shortcut Shift-C).
   (If the selected character is already calibrated, **Recalibrate** is displayed on the context menu. If you select this option, the existing calibration is overwritten.)



Each character is calibrated and Evoke renders the character in the **3D Scene** view. You can set the **View Filters** to show either the character source (solving) view or the retargeting view. The scale value that Evoke calculates for each character is used by the Unreal or Unity Plugin to render the character at the correct size for each participant.

About cluster calibration

Cluster calibration corrects for minor variability in Pulsar placement on the feet and spine, along with refining the skeleton to give more lifelike character-solving. This helps adapt Characters From Clusters (CFC) to inconsistent application of the clusters to the body for specific objects on certain common axes. This happens automatically when you calibrate characters.

Example of a misaligned foot cluster:

Without cluster calibration

With cluster calibration

> **Note**
> Not all clusters are calibrated on all axes. This feature adapts to minor deviations for foot and spine orientation and position to adjust for commonly misplaced locations.

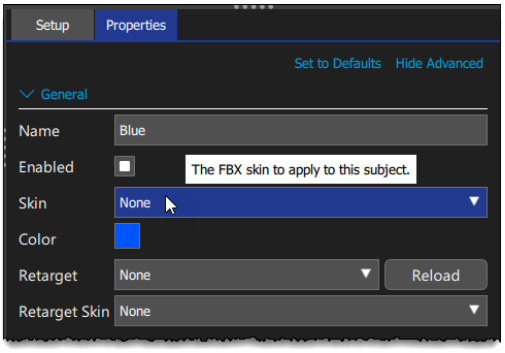
Change a character's properties (optional)

Select the skin and color

By default, to enable pre-visualization of solve quality, new characters are displayed in the Workspace with the Vicon source skin. To enable you to identify the character more easily, you can select a male or female skin variant as well as a skin color.

To change a character's skin and/or color:

1. Ensure the character whose skin or color you want to change is selected.
2. In the **Tracking** pane, on the **Properties** tab, click **Skin** or **Color** in the **General** section.
3. Select the required skin or color.



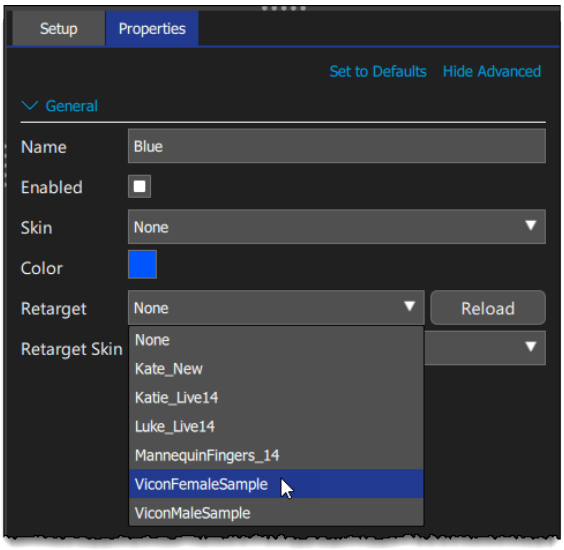
To view the character's skin, you must calibrate the character.

Select the retarget file and retarget skin

You can retarget the source skeleton to a suitable character skeleton for use in a game engine or visualization tool. You can also preview the character mesh in Evoke. Retargeting requires a retarget setup file (*.vsr) that has been created in the Vicon Retarget application (see Set up character retargeting, page 130).

You can choose the retarget file for the character when you first create the character (see Step 3, page 110 of *Create characters*), but you can change both the retarget file and select a retarget skin.

1. In the **Tracking** tree, ensure the required character is selected.
2. On the **Properties** tab, in the **General** section, from the **Retarget** menu, select the required file.



3. From the **Retarget Skin** menu, you can also select a retarget skin for visualization in Evoke. For example you might have one retarget file for your character, but a number of different colored skins to represent each player.
   To display correctly, the retarget skin must have the same skeleton as the retarget setup.

Retarget files are found in this default location:

C:\Users\Public\Documents\Vicon\Retargets

Installed retarget subject files are located by default in:

C:\Program Files\Vicon\Evoke1.4\Configuration\Retargets

Clear a calibration and un-assign clusters

Clearing a calibration stops the character solve and returns the skeleton to the default scale, ready for the next participant.

It also returns a character's assignable objects to their unassigned state, ready for their next use.

To return a character's calibration to the default scale and un-assign clusters:

- In the **Tracking** tree, right-click the character name and then select **Clear Calibration**.

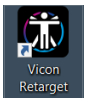
Set up character retargeting

The Vicon Retarget app is installed with Evoke. You can start it in any of these ways:

- From within Evoke:
  - On the **Retarget** menu, click **Launch Setup**.

    or
- From the Windows Start menu:
  - Expand **Vicon** and then select **Vicon Retarget**.

    or
- On the desktop, double-click the **Vicon Retarget** shortcut.



Vicon Evoke 1.4 and retargeting

- Evoke 1.4 reads both VST 3.5 and 3.6 files, but exports VST 3.5 files. VST 3.4 files, as used with Evoke 1.2, are compatible.
- Avoid using Evoke 1.4 with Retarget 1.0.2 or earlier as the cluster template from Evoke 1.4 will not load into these earlier versions of Retarget.
- If you created retarget files in Retarget 1.0.2 or earlier, to benefit from the improved scale estimation and cluster calibration in Evoke 1.3 and later, recreate the retarget files in Retarget 1.1.2 or later.

You can create retarget subjects from any suitable FBX file, using position and rotation constraints to drive target skeleton bones from the Vicon source skeleton bones. As the Vicon source skeleton is human, retargeting works best with biped targets.

Save the retarget subject files to a filename that you will be able to identify easily. Pre-installed examples are included for retargeting the Vicon source skeleton to a Vicon target skeleton.

> **VSX files**
> If you created VSX files in earlier versions of Retarget and want to use them in Evoke 1.2 or later, load the VSX files into the latest version of Retarget and save them as VSR files.

The following topics explain the procedures that are related to character retargeting in the order you are likely to use them:

- Copy the relevant files, page 132
- Select the skeleton and the FBX, page 132
- Set the root and scale the target subject, page 134
- Align the target joints, page 136
- Add position constraints, page 141
- Add rotation constraints, page 144
- Remove unnecessary joints, page 146
- Save the finished setup, page 147

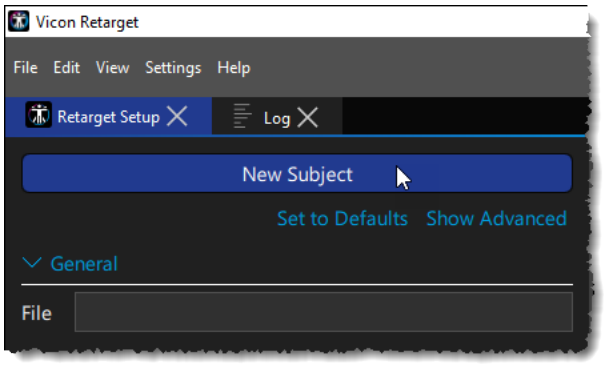
Copy the relevant files

If you want to use your own retarget subject instead of using one of the supplied samples, ViconFemaleSample or ViconMaleSample, copy your FBX file to:

C:\Users\Public\Documents\Vicon\Skins

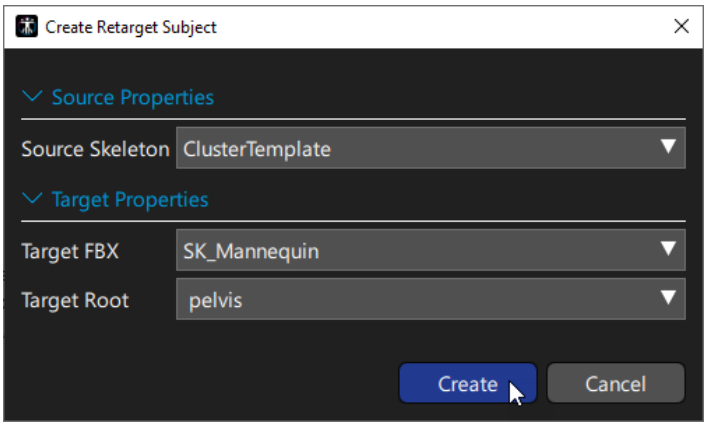
Select the skeleton and the FBX

1. Start Evoke, and on the **Retarget** menu, select **Launch Setup** (or start Retarget in one of the other ways described in Set up character retargeting, page 130).
2. In the **Vicon Retarget** window, click the **New Subject** button at the top of the Retarget Setup tab.

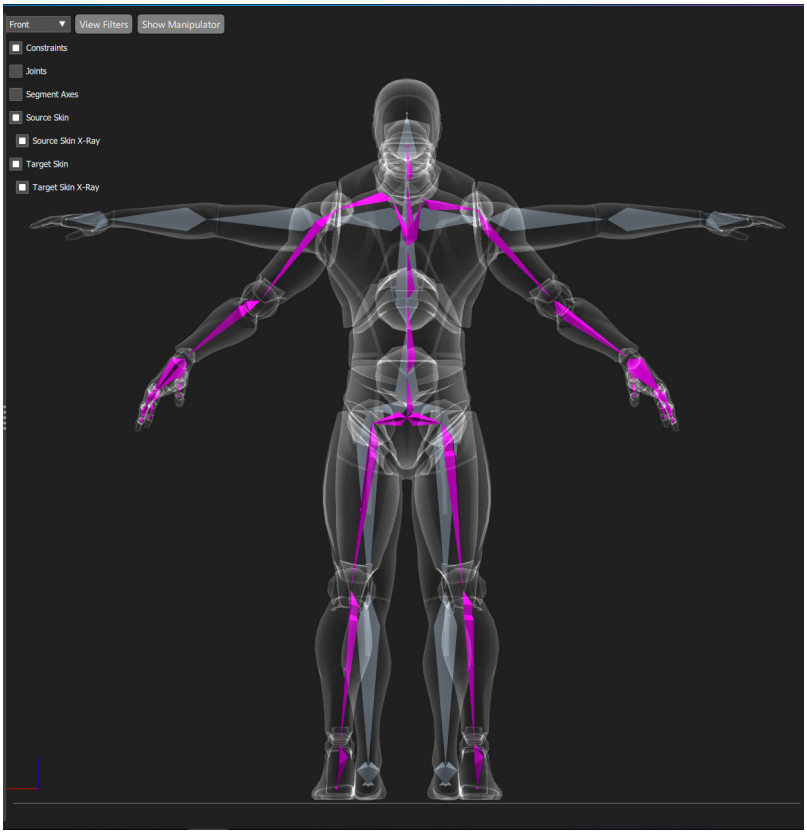


3. In the Create Retarget Subject dialog box:
   a. In the **Source Skeleton** section, click in the **Filepath** field and select the required template.
   b. In the **Target FBX** section, click in the **Filepath** field and select the required FBX file.

c. In the **Root Segment** field, select the appropriate root (if you are not sure, try **hips** or **pelvis**).

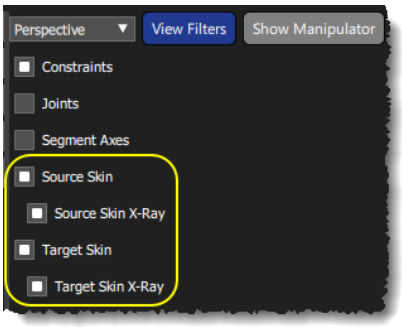


4. To create the target subject, click **Create**.
   The target FBX appears in the middle of the view.

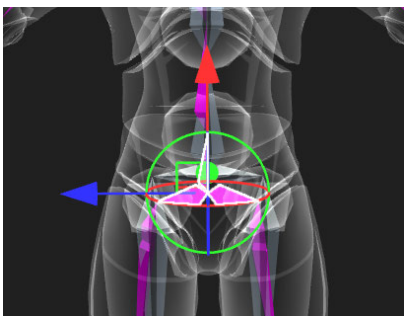


Set the root and scale the target subject

Note that in the view pane, by default, the **View Filters** options **Source Skin**, **Target Skin** and both **Skin X-Ray** options are selected. As you work, you may find it easier to see the relevant details if you select or clear the appropriate options.



1. Ensure you have completed the steps in Select the skeleton and the FBX, page 132, and that the target and source skeletons are displayed in the middle of the view.
2. In the **Target Properties** section on the left, in the **Scale** field, set an appropriate value to get the size of the target skeleton to match the source skeleton.
3. In the view pane, click on the target pelvis (or the root that you selected) to activate the Manipulator tool, and drag to align the target and source skeletons.

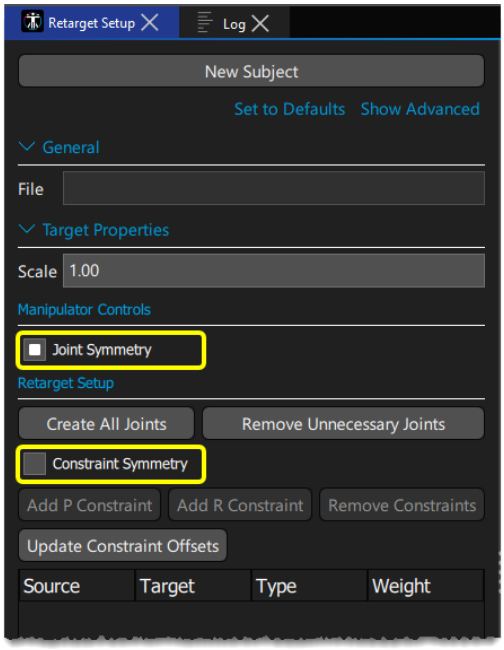


> **Tip**
> Adjust the target scale so that the head and shoulders match as closely as possible, with the feet on or slightly above the floor. You may need to align the target joints then adjust scale, and repeat until you're satisfied.

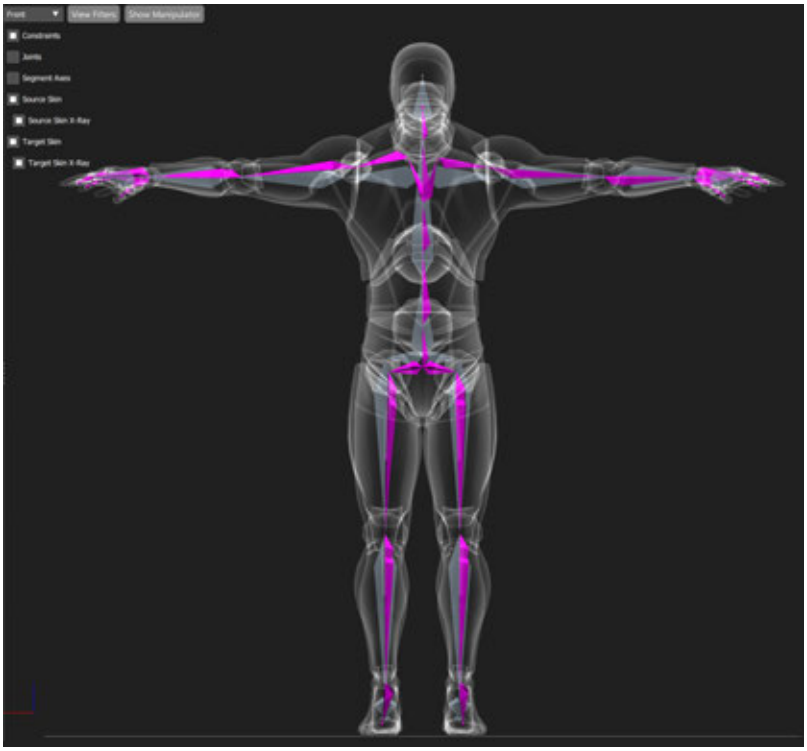
Align the target joints

For best results, align the target skeleton as closely as possible to the source skeleton.

1. Before you begin, review the symmetry options in the **Manipulator Controls** and **Retarget Setup** sections on the left.
   - To mirror changes to a joint rotation from one side to the other (from the left side side to the right side or vice versa), in the **Manipulator Controls** section, ensure **Joint Symmetry** is selected.
   - To mirror constraints created on one side to the other (from the left side side to the right side or vice versa), in the **Retarget Setup** section, select **Constraint Symmetry**.

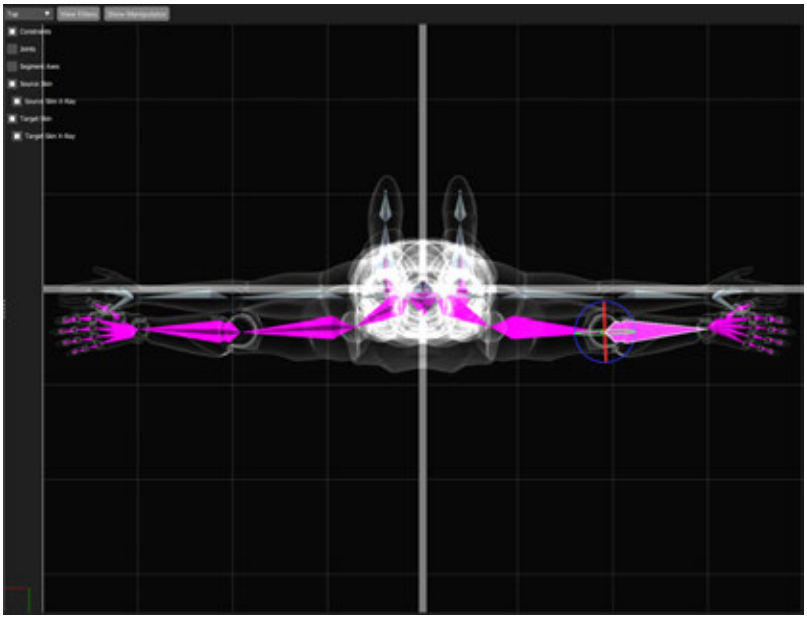


2. Use the Manipulator tool to roughly align the main joints, switching between different camera views as required.

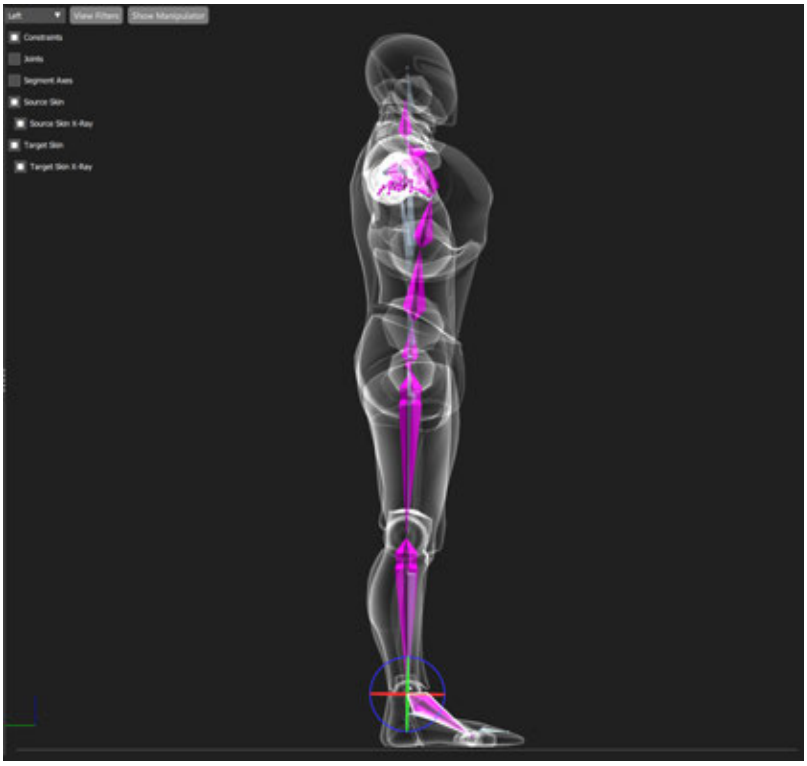


> **Tip**
> To scale the Manipulator, on the numeric keypad, press + (scale up) or - (scale down).

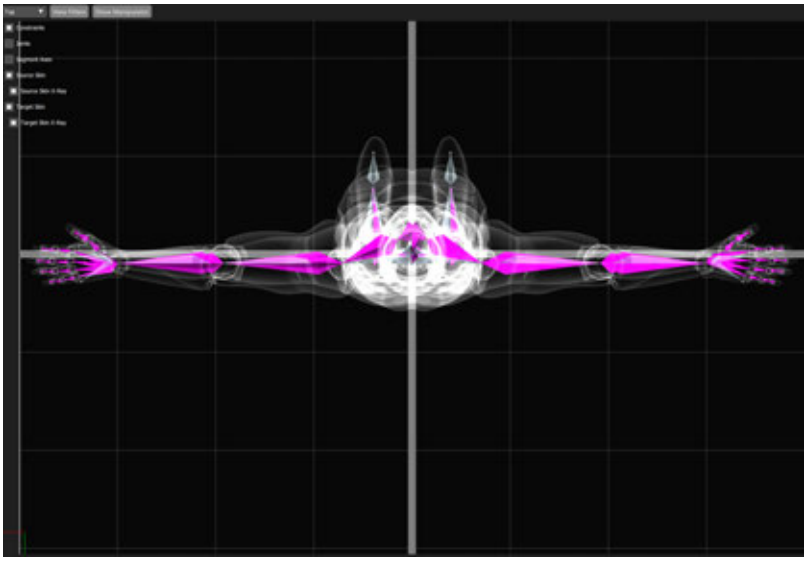
From the top view, you can see that the target shoulders are pushed too far back. This must be corrected in the spine joints, so initially you just straighten out the arms.



3. Compensate for differences in skeletal proportions by adjusting intermediate joints.



In this example, some curvature to the target spine has been added. This has the effect of moving the shoulder joints down and forward to match the source skeleton.



The target foot is also rigged quite differently to the source skeleton, which must be accounted for when the position targets are created.

4. Repeat steps 2 and 3 until you are happy with the alignment of the target and source skeletons.

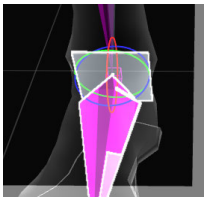
Compensating for differences in proportion

- To achieve good hand-eye coordination in VR, you must accurately match the hand position and rotation; the head and shoulders also affect coordination to a lesser extent.
- When you align the target skeleton, try to prioritize matching the hands for best results. The feet are similarly important to aid navigation.
- For intermediate joints, the choice is often aesthetic, and you may need to experiment with a moving subject to see what looks best.

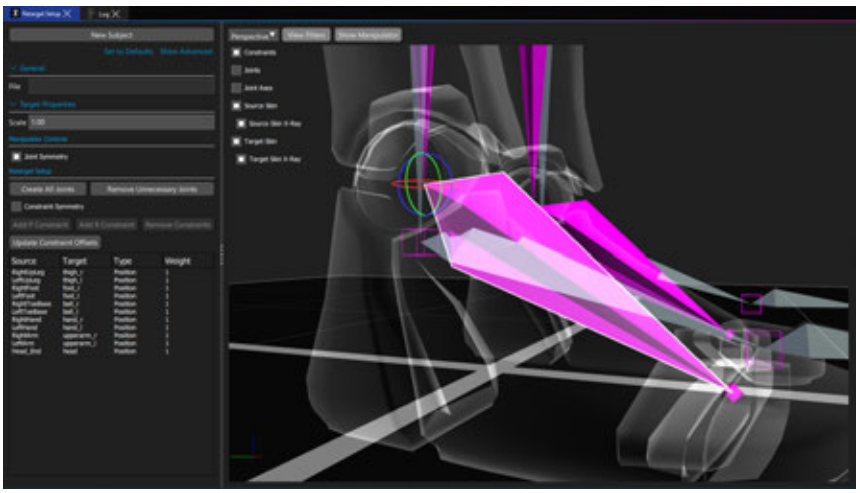
Add position constraints

Position constraints (P Constraints) are used to ensure that parts of the retarget skeleton match the position of parts of the source skeleton.

1. Add position constraints to the ends of the body (hands, feet and head). To do this, click the bone on the target skeleton (pink), then click the equivalent bone on the source skeleton (gray), or select from the Source Joints and Target Joints lists on the right, and then on the **Retarget Setup** tab, in the **Retarget Setup** section, click **Add P Constraint**.
   The constraints are displayed on the Retarget Setup tab as a list and are also displayed graphically.



In some cases, the bones may be quite different between the source and target skeletons. Take extra care to ensure you apply position constraints to the correct bones.

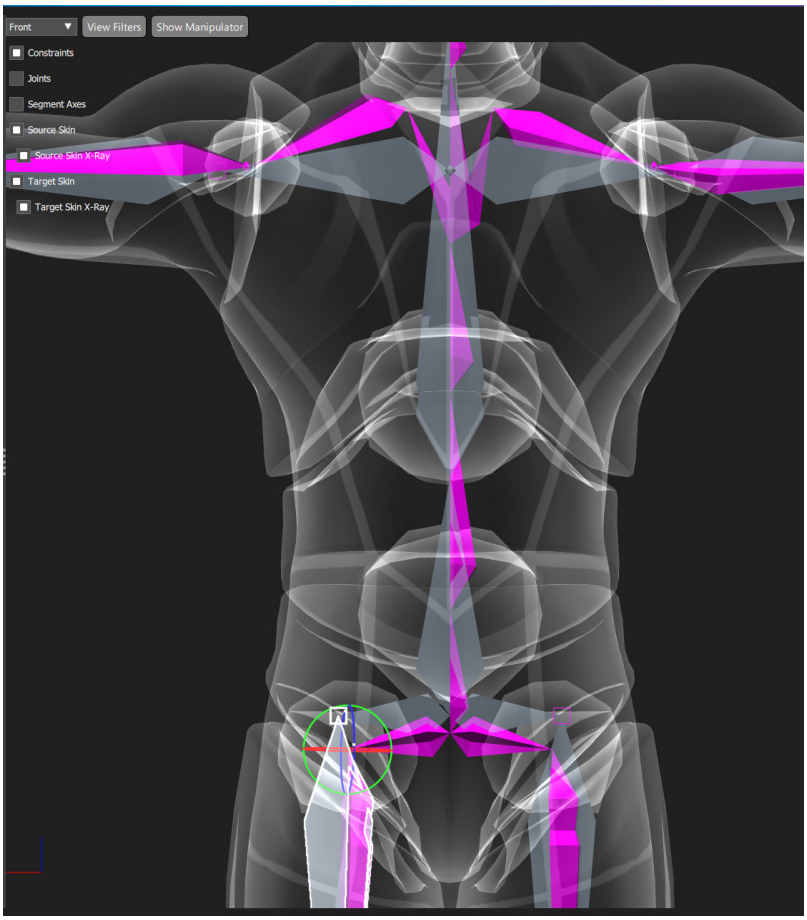


Understanding position constraints

A position constraint adds a constraint to the retarget solver in Evoke, so the position of the target bone relative to the source is preserved. The wireframe cube represents this offset, and by default is set to origin of the source bone. This means that Evoke tries to pose the retarget so that the cube is centered on the source bone origin.

The default offset is usually a good choice, assuming you have aligned the target skeleton well to the source skeleton. However, you can adjust the offset by selecting the constraint from the list on the Retarget Setup tab and changing the **Offset X**, **Offset Y** and **Offset Z** fields in the bottom-left of the Retarget Setup tab. For example, if you set the offset to (0 0 0), the target bone origin is forced to match the source bone origin exactly. This can sometimes improve retarget results when the default position offset is very small, as it is easier to find the correct target joint rotation when the target position is exactly the same as the source.

2. Add at least one position constraint to the trunk of the body. This is necessary to prevent the body 'floating' when only constrained by rotation constraints. If you find that these target joints slip out of position when solving in Evoke, adding position constraints on the hips and shoulders may be a good alternative.

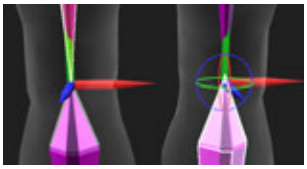


Note that you don't need to set position constraints on 'in-between' joints (eg, elbows), because usually, some differences between source and target are acceptable here. It is recommended to keep the number of position constraints to a minimum. Over-constraining the solver in Evoke is counter-productive, usually resulting in persistent retarget errors and/or instability.

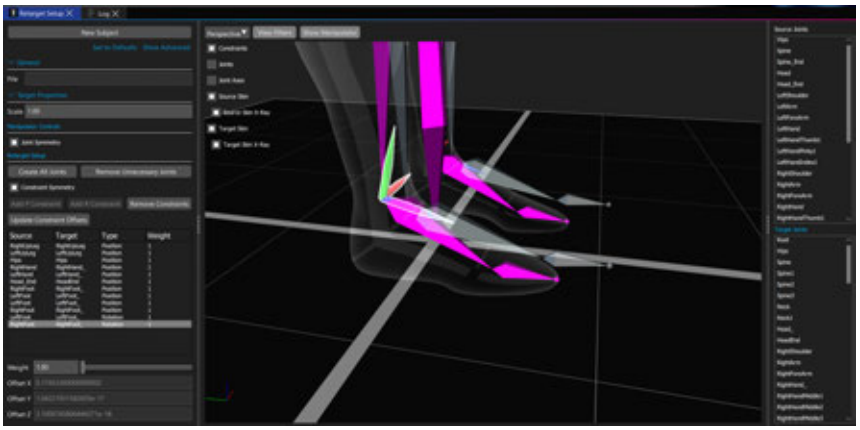
Add rotation constraints

Rotation constraints (R Constraints) serve a similar purpose to position targets, but constrain the rotation of joints in the retarget.

1. As with position constraints (see Add position constraints, page 141), click on the target bone then the source bone (or select from the lists on the right), and then click **Add R Constraint**.
   Each rotation constraint is added to the list on the Retarget Setup tab and is also displayed graphically.
   In general, add a rotation constraint for each bone in the target skeleton. You can use the same source bone for multiple target bones; this is necessary if the target skeleton has more bones than the source skeleton in the same body segment.



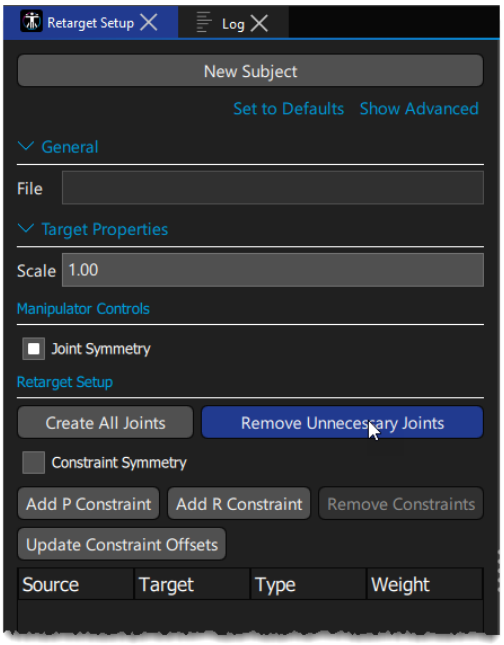
2. The relative importance of rotation and position targets is determined by their respective weights. To adjust the weight of a constraint, in the list in the Retarget Setup section, select the constraint and change the value in the **Weight** field displayed below. A higher weight makes the constraint more effective in constraining the retarget pose, at the expense of the other targets.



Understanding rotation constraints
A rotation constraint adds a constraint to the retarget solver in Evoke, so the rotation of the target joint relative to the source is preserved. The RGB axes centered on the joint represent this target offset, and by default is set to the rotation of the source joint. This means that Evoke tries to pose the target joint so that the RGB axes have the same rotation as the source joint. The joint state is an angle-axis value, and manually editing it is not recommended.

Remove unnecessary joints

To remove unconstrained joints on the template, on the Retarget Setup tab, click **Remove unnecessary joints**.

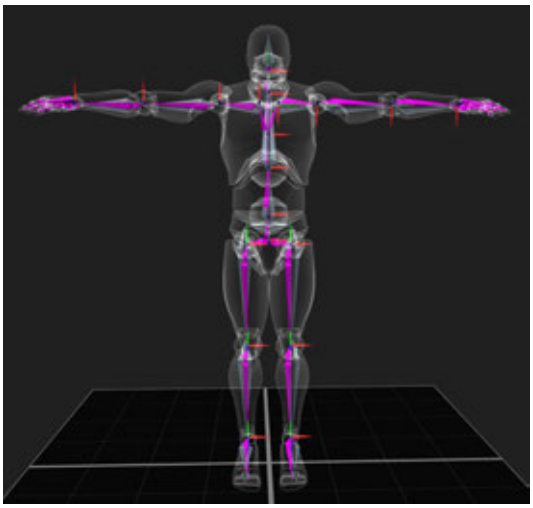


This step is necessary because unconstrained joints still have to be fitted by Evoke, which increases computational load and may cause dropped frames. For example, finger or face bones don't have an equivalent on the source skeleton, so nothing needs to be constrained. They are just fixed in place in the solve when 'removed' from the retarget subject.

Save the finished setup

To save your setup, on the **File** menu, click **Save As** and save your retarget file to the *Retargets* folder, by default in:

C:\Users\Public\Documents\Vicon\Retargets



Use proximity-based tracking

Proximity grouping enables more than 70 Pulsar clusters to be used in a single volume. To achieve this, Evoke permits some duplication of patterns between Pulsars. It distinguishes between the duplicated patterns by linking the patterns to unique objects by their proximity to these unique objects.

For more information, see the following topics:

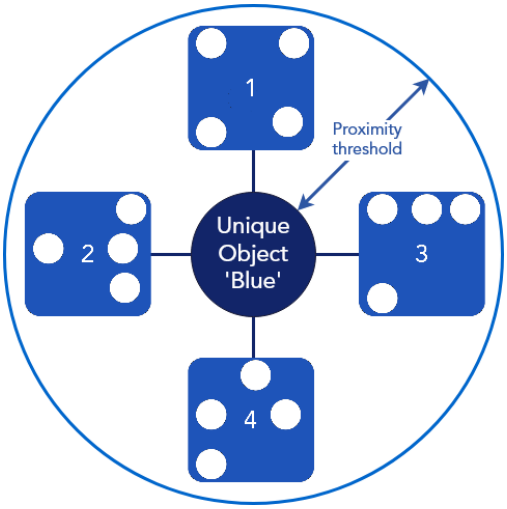
- Introduction to proximity-based tracking, page 149
- Set up a proximity group in Evoke, page 151
- Run proximity-based tracking, page 154
- Use proximity grouping and auto-assignment together, page 156

Introduction to proximity-based tracking

Origin uses 'always-on' tracking for its active LED devices, identifying objects by the spatial marker arrangement, rather than strobing LEDs on and off during tracking frames to create an ID. While this enables continuous tracking across every frame, even with the fully asymmetrical spacing of LEDs, this limits the number of Pulsars that can be tracked (4 from 8 LEDs, 4C8) to 70 unique patterns.

To exceed this number of tracked Pulsars, Evoke uses proximity-based tracking to 're-use' patterns within a volume in a logical way, without compromising tracking performance.

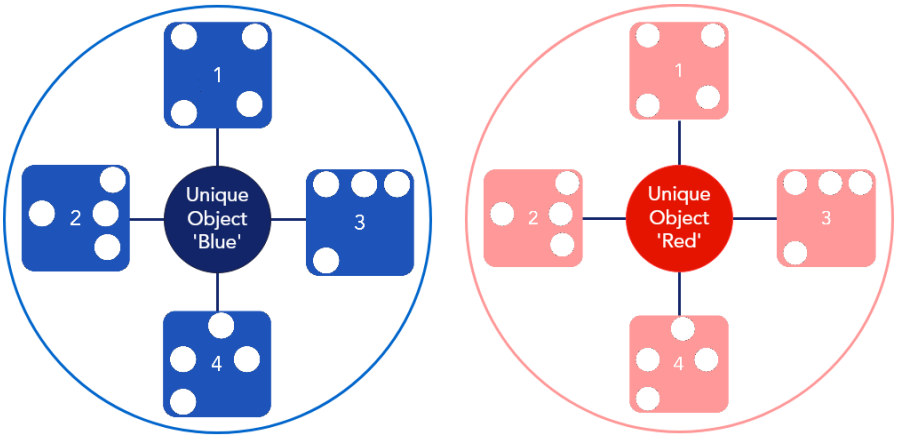
What is proximity-based tracking?



Proximity grouping works by assigning the grouped identity (e.g. 'Blue') to an object based on both its pattern AND its distance to another object (that has a unique pattern), which acts as an identifier.

While the identifier must be unique, the related objects can share patterns with other smart objects.

Objects are labeled correctly as long as they are within a defined distance (the proximity threshold) of the identifying object.



To prevent Evoke from misidentifying objects, follow these guidelines when setting up proximity tracking:

- Minimize use of template objects; use unique objects whenever possible.
- Avoid template objects coming within the proximity threshold of multiple unique objects. The default distance is 1.5 m: to change this, see Set the proximity threshold, page 151.

Although it is possible to use objects in the same template group interchangeably, doing so is *not* recommended as it complicates the management of the clusters.

Set up a proximity group in Evoke

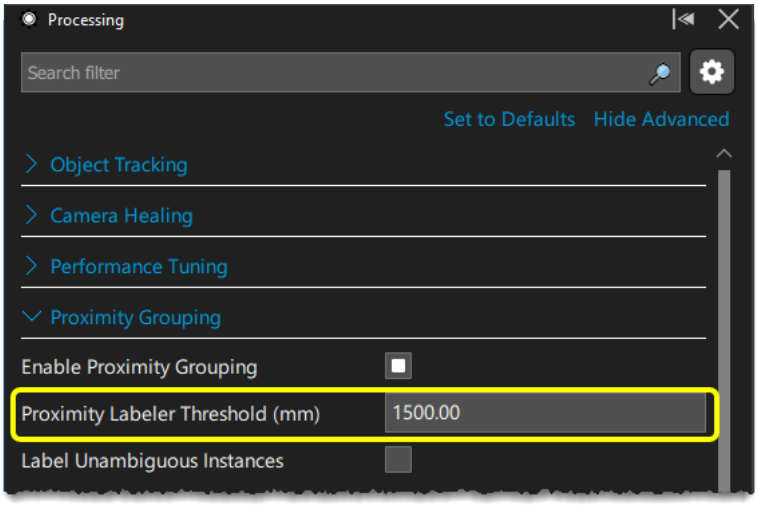
> **Note**
> This section describes how to set up proximity grouping from within Evoke. In addition to setting up proximity groups by specifying the parameters described in this section, you can also do this via the API, which may be more convenient if you need to change the setup regularly.

Set the proximity threshold

The proximity threshold is the maximum distance between an object and another object that has both the same proximity group and a unique identity, within which objects can be labeled with their correct identity. The default proximity threshold is 1.5 m.

To set the identifying distance:

On the **Processing** tab, ensure that the Advanced options are displayed and in the **Proximity Grouping** section, set the required distance in millimeters, specifying a value that is appropriate for your environment.



Set up the objects to be used in proximity grouping

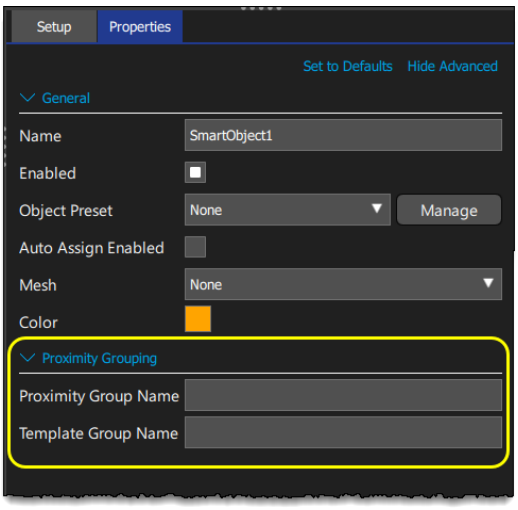
These types of objects are involved in proximity grouping:

- **Template objects**. Objects with the same Template Group Name must have the same configuration of markers (LEDs). Any object with a non-empty string for Template Group Name is a template object.
- **Unique objects**. Objects with no Template Group Name specified are unique objects, and must not have the same configuration of markers as any other object in the system.

The template objects are identified by their proximity to unique objects that share the same Proximity Group Name. For an object in a template group to be labeled correctly, it must be within a pre-defined distance (known as the proximity threshold) of a unique object in the proximity group. The default distance is 1.5 m; to change this, see Set the proximity threshold, page 151.

To prepare objects for proximity grouping:

1. Ensure you have created and linked the necessary objects in Evoke (see Work with Smart Objects, page 39 and Work with basic objects, page 59).
2. In the Tracking tree, select each object and on the **Properties** tab, set the following advanced parameters:



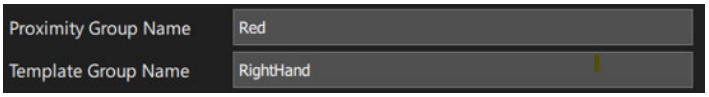
- In the **Proximity Group Name** field, specify the proximity group to which the object belongs. Note that this name is case-sensitive.

- If it's a template object, in the **Template Group Name** field, specify the template group to which the object belongs. If it's a unique object, leave this field empty. Note that this name is case-sensitive.

3. Ensure you have set an appropriate proximity threshold (see Set the proximity threshold, page 151).

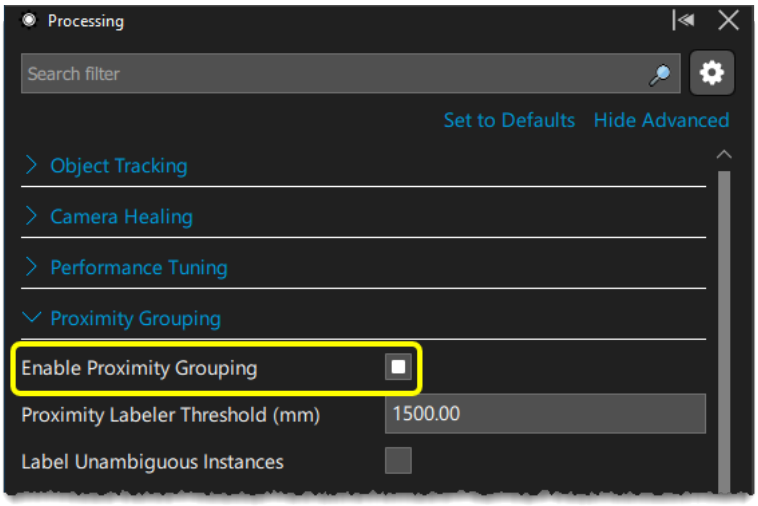
This table shows a simple example with two characters, Red and Green, each made up of three objects: Head, LeftHand, and RightHand:

Object	Proximity Group Name	Template Group Name	Pattern
Red_Head	Red		31
Red_LeftHand	Red	LeftHand	47
Red_RightHand	Red	RightHand	79
Green_Head	Green		143
Green_LeftHand	Green	LeftHand	47
Green_RightHand	Green	RightHand	79



Run proximity-based tracking

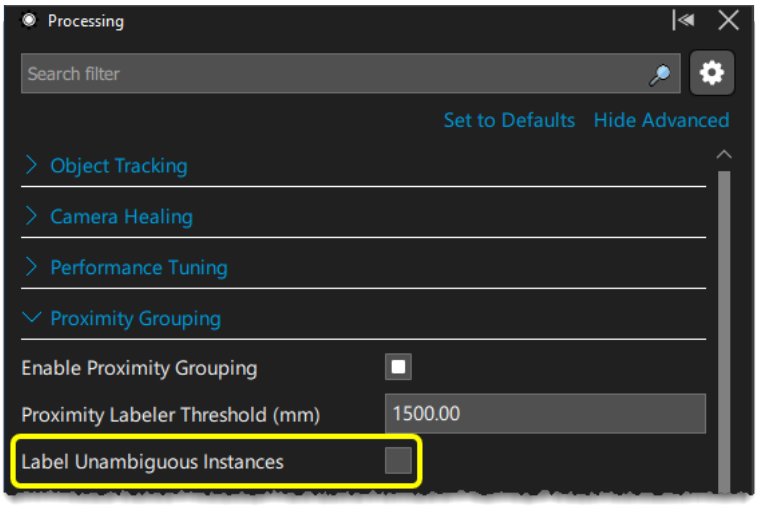
After the objects are set up in Evoke, on the **Processing** tab, in the **Proximity Grouping** section, ensure **Enable Proximity Grouping** is selected.



Use the Label Unambiguous Instances option

The **Label Unambiguous Instances** processing option enables you to control the labeling of template objects that aren't in proximity to a unique object in their proximity group.

- If disabled (the default), template objects are not labeled unless they are in proximity to a tracked unique object in their proximity group.
- If enabled, template objects are also labeled when there is no ambiguity (there is only one option for labeling).
  This is the case when all other template objects with the same pattern have already been labeled, due to proximity to a unique object in their proximity group.



If you enable this feature, template objects are labeled without a positive identification of the unique object in proximity. This means that objects are labeled more quickly than when this option is disabled, but introduces an additional risk of mis-labeling if unique objects were previously misidentified for any reason.

Use proximity grouping and auto-assignment together

To use proximity grouping and auto-assignment (see Assign clusters and calibrate characters (automated workflow), page 114) together, the following setup is recommended.

This example setup consists of several groups of people, Red, Green, Blue, etc. Each group consists of multiple users, eg, Red consists of Red1, Red2, ... ; Green consists of Green1, Green2 ...

Every user must have at least one *unique* object. This is usually the headset, eg, Red1_Headset, Red2_Headset, ...

Each group also has a set of Pulsar objects, (Red : Red_AA, Red_AB, Red_AC ... ; Green : Green_AA, Green_AB, Green_AC ...). Each user in the group takes several of the Pulsar objects and ***provided they are in the same group***, those Pulsar objects can be used interchangeably between roles (eg, LeftFoot, RightFoot, LeftHand, RightHand) for ***any*** of the users in the group.

To ensure that proximity-based tracking works correctly, it is recommended that only objects in the same group are near each other in the volume (ie, come within the same proximity threshold). This works well when only a single group is in the volume at one time, or the volume is large enough to avoid groups merging.

The following table shows how to set up the Proximity Group Name and Template Group Name for objects.

Object	Proximity Group Name	Template Group Name
Red1_Headset	Red	
Red2_Headset	Red	
Green1_Headset	Green	
Green2_Headset	Green	
Red_AA	Red	AA
Red_AB	Red	AB
Red_AC	Red	AC
Green_AA	Green	AA
Green_AB	Green	AB
Green_AC	Green	AC