

# VICON iQ 1.5

## Tutorial 3: Using the Real Time Mode

---

## CONTENTS

VICON iQ 1.5 .....	1
CONTENTS.....	2
Chapter 1 - New Improved Real Time Mode.....	4
Introduction .....	4
Chapter 2 - Real Time Processing for Capture and Visualisation.....	6
Introduction .....	6
Lesson 1 - Setting up Real Time .....	6
Step 1: Set Up .....	7
Step 2: Simultaneous Capture.....	7
Step 3: Configuration Options.....	8
Step 4: Active Subjects.....	9
Step 5: Control .....	9
Step 6: Capture .....	10
Chapter 3 - Real-time Visualisation .....	11
Introduction .....	11
Visualising Real Time Data .....	11
View Pane Types .....	12
View Options .....	13
Marker Residual .....	13
Chapter 4 - General Workflow using Real Time.....	14
Introduction .....	14
Real Time Workflow for Broadcast and Virtual Characters.....	14
Real Time Pre-Visualisation and Capture for Post Production. ....	16
Chapter 5 - Real-time Configuration Options and How to use them.....	18
Introduction .....	18
The Basic Configuration Options.....	18
Datastation IP.....	18
Trajectory Prediction Error.....	18
Proportional Marker Tolerances .....	19
Ray Noise Factor.....	19
Camera Noise Factor .....	20
Circle Fitting Error .....	20
Minimum Circle Fitting Quality.....	20
Minimum Circle Fitting Diameter.....	21
Maximum Circle Fit Diameter .....	21
Min. Cameras to Reconstruct .....	21

## TUTORIAL 3: Using the Real Time Mode

---

Min. Reconstruction Separation.....	21
Include Cameras .....	21
Mask File.....	21
Advanced Configuration Options.....	22
Predictor Order.....	22
Skip Rate.....	22
Fixed Marker Tolerance.....	22
Marker Statistics Threshold .....	22
Tolerance on Joint Placement .....	23
Proportional Joint Limit Tolerance .....	23
Start Labelling and Labelling Increase Ratio .....	23
Maximum Iterations of Kinematic Fitter.....	23
Kinematic Prediction Order.....	24
Kinematic Prediction Revolute.....	24
Kinematic Prediction Translation .....	24
Marker Stickiness When Previously Unlabelled.....	24
Marker Stickiness When Previously Labelled .....	25

# TUTORIAL 3: Using the Real Time Mode

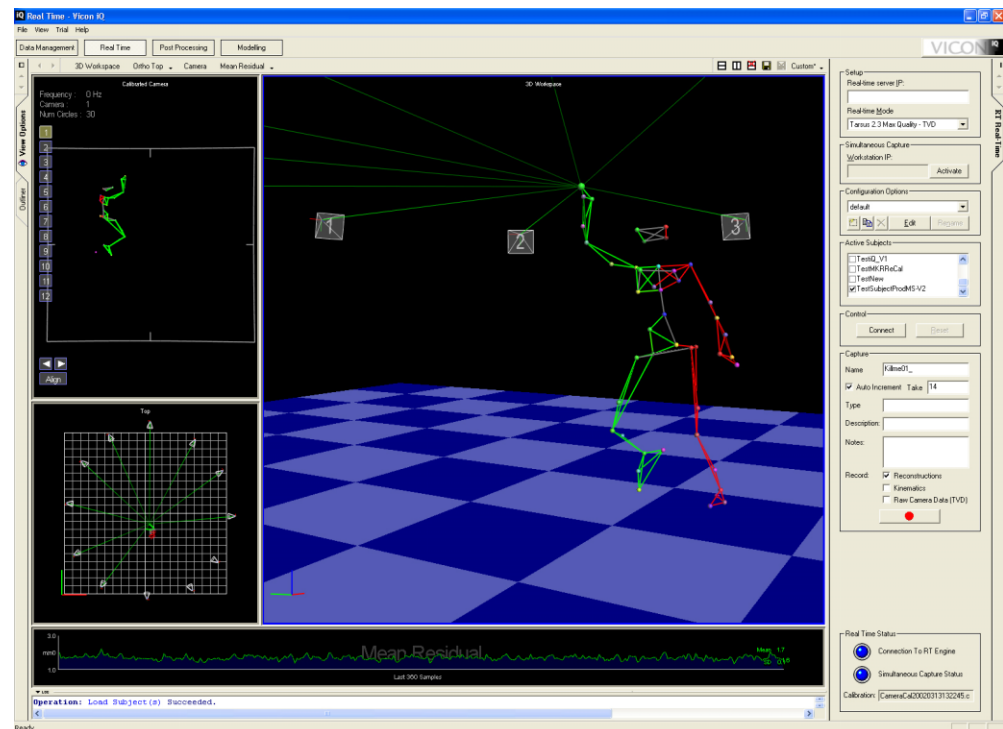
## Chapter 1 - New Improved Real Time Mode

### Introduction

Vicon iQ 1.5 incorporates a new powerful Real-Time data visualization mode.

If you have ever worked with the Real-Time Engine (Tarsus) and Vicon iQ 1.0 in the past, you will notice significant improvements in both data quality, and your ability to visualize data within Vicon iQ 1.5

*Note: This document outlines the major additions to Vicon iQ 1.5's real time mode. It assumes you are familiar with Vicon iQ 1.0's real time mode. Please refer to the Vicon iQ 1.0 Manuals, in particular the Vicon iQ 1.0 Real Time Mode section.*



Vicon iQ 1.5 allows you to set parameters and see the results instantly in real-time, and record the Real-Time data stream as a Vicon iQ trial file for subsequent post processing and editing.

Vicon iQ 1.5 can also remotely trigger Workstation data capture of camera data (TVD) and other external devices. All files captured with Vicon iQ 1.5 and Workstation will all be integrated within the Datamangement Mode of Vicon iQ automatically. These can then be brought into Vicon iQ 1.5 for processing in the same way as traditionally captured offline data. Vicon iQ 1.5 can also simulate Real-Time data by streaming reconstructions from a previously captured TVD file.

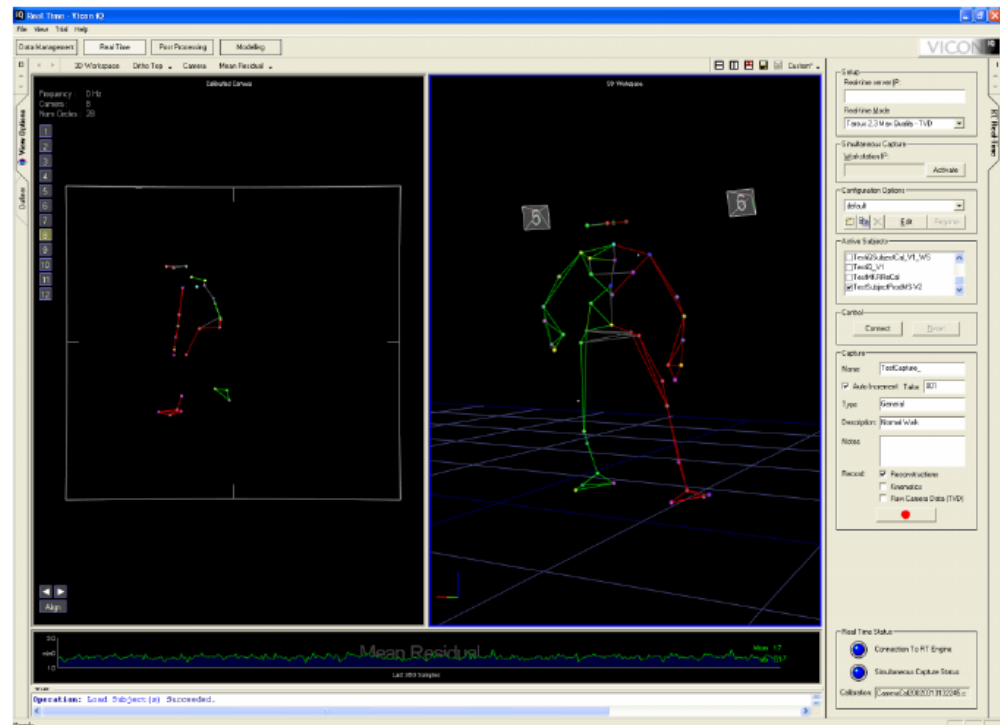
Vicon iQ 1.5, in Real-Time mode, is used to connect, set up, and manage the Vicon Real-Time Engine (Tarsus). Using the Real-Time mode of Vicon iQ 1.5, you will be able to control the Vicon Real-Time Engine, and control Workstation to capture all the raw Motion Capture data. Vicon iQ 1.5's Real-Time mode, provides real time feedback of what is going on with the system and the captured subjects (actors).

You will easily tell the state of the system, state of calibration, state of subject calibration and how well your capture session is proceeding.

## TUTORIAL 3: Using the Real Time Mode

The following is now available within the Real Time Mode of Vicon iQ 1.5

- All new Real Time Processing for Capture and Visualization
- Display of Raw Camera Data, 2D edges, circles, labeled circles.
- Diagnostic display, calibration status, reconstruction rays, contribution rays etc.
- Improved Real Time Diagnostic Graphs
- New Maximum Speed, and Maximum Quality Modes.
- Record trials, and playback in Real Time.
- Record Raw Camera Data, 3D and Kinematics
- Improved Real Time Labeling and Kinematics.
- Direct Integration of capture trials and TVD into Data Management, along with Auto Incremented Take Number, Trial Info, Description and Trial Type.
- Remotely Connect to Workstation to trigger external device capture.



## TUTORIAL 3: Using the Real Time Mode

### Chapter 2 – Real Time Processing for Capture and Visualisation

#### Introduction

This tutorial is intended as a quick start guide to using the Real Time mode of Vicon iQ 1.5.

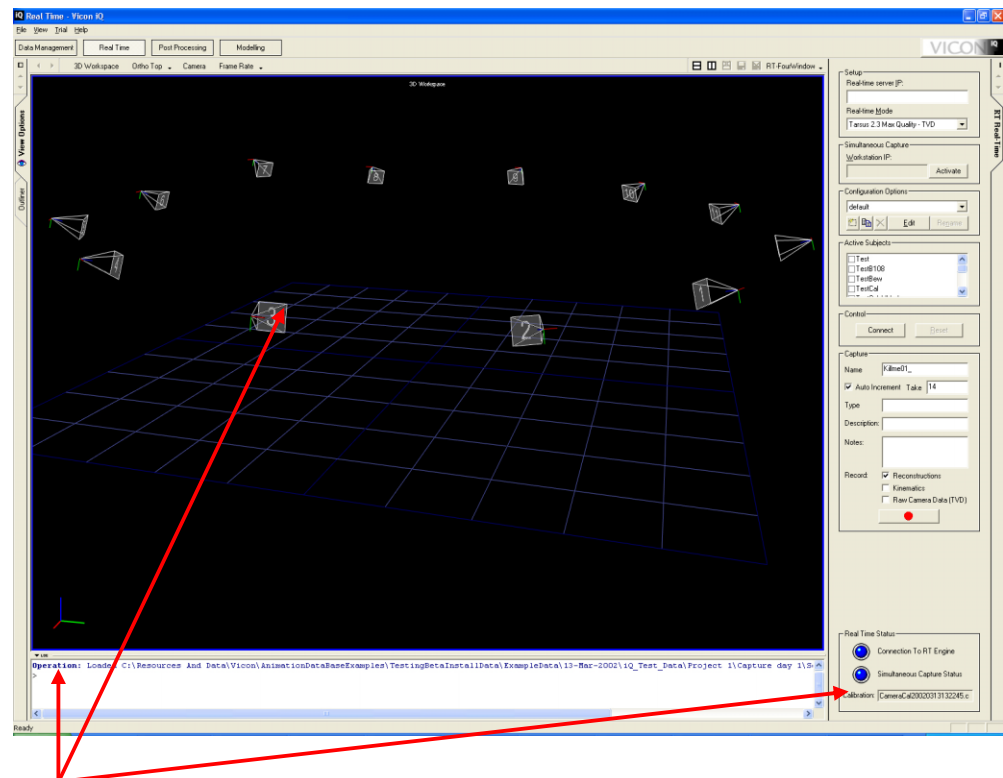
This tutorial assumes that a successful system calibration has been achieved as well as a successful subject calibration for all talent and props and that the session with these calibrations is selected in the Eclipse (Data Management) menu.

*It is recommended that you complete Tutorial 1 - Subject Calibration, and familiarise yourself with the concepts of camera calibration using Workstation.*

#### Lesson 1 – Setting up Real Time

Start by selecting the Real Time tab at the top of the screen. This places Vicon iQ 1.5 into Real Time mode and enables you to both view and capture data in real time.

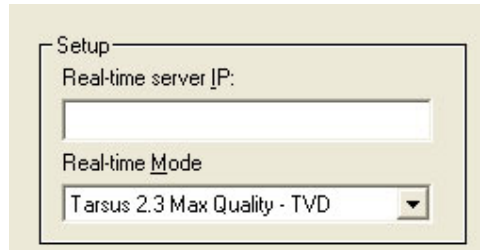
*Tip: Vicon iQ 1.5 will only allow you to enter the Real Time Mode, if you have a valid Eclipse session activated within the Datamangement mode, and this Eclipse session contains a valid camera calibration file (\*.cp)*



*Tip2: When you enter the Real Time Mode in Vicon iQ 1.5, you can check that the camera calibration is correct, and from the right Eclipse Session, by looking at the camera calibration info, in the Real Time Status Area, by looking in the Log, and or course by visualising the cameras in the 3D Workspace.*

## TUTORIAL 3: Using the Real Time Mode

### Step 1: Set Up



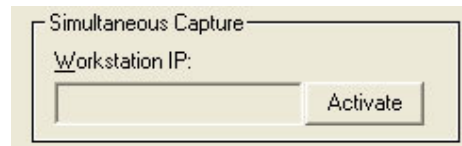
The set-up section on the top right of the screen requires the IP address of the computer that is running the Real Time Engine (Tarsus client). If it is running on the same machine as Vicon iQ 1.5, then this field can be left blank.

*Note: For optimum performance you can run the Real Time Engine on a separate machine, or separate CPU if on a dual processor PC. If running on a dual processor PC, you can ensure that the Real Time Engine is running on a single CPU, with Vicon iQ 1.5 running on the 2<sup>nd</sup> CPU, by going to the Task Manager, and setting the "Affinity".*

The Real-time Mode determines whether real time is run in Max Quality - TVD (The best labelled and reconstructed data streaming from a pre recorded TVD), Max Quality - Live (The best labelled and reconstructed data from talent and props in the volume), Max Speed - TVD (The fastest possible reconstructions and labels from a pre recorded TVD), or Max Speed - Live (The fastest possible reconstructions and labels from talent and props in the volume)

### Step 2: Simultaneous Capture

Vicon iQ 1.5, allows you to easily remotely connect to Workstation, across a network, or with Workstation running on the same PC. This allows Vicon iQ 1.5, to simultaneously trigger Workstation to capture any other devices, along side TVD and Vicon iQ 1.5's real time stream.



Simultaneous capture requires the IP address of the computer running Vicon Workstation software. If it is running on the same machine as Vicon iQ 1.5, this can be left blank.

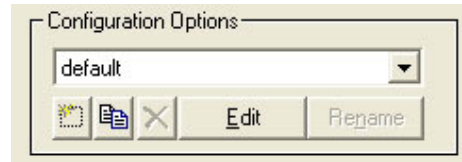
The ACTIVATE button must be depressed if TVD capture is going to be performed in conjunction with real time data capture.

Note: For best results, set up a "Trial Type" in Workstation, that :

- Does not open any workspace windows after capture.
- Does not perform any pipeline operations after capture
- Only captures the data you want.
- Has Workstation looking at the same Eclipse database, that Vicon iQ 1.5 is.

## TUTORIAL 3: Using the Real Time Mode

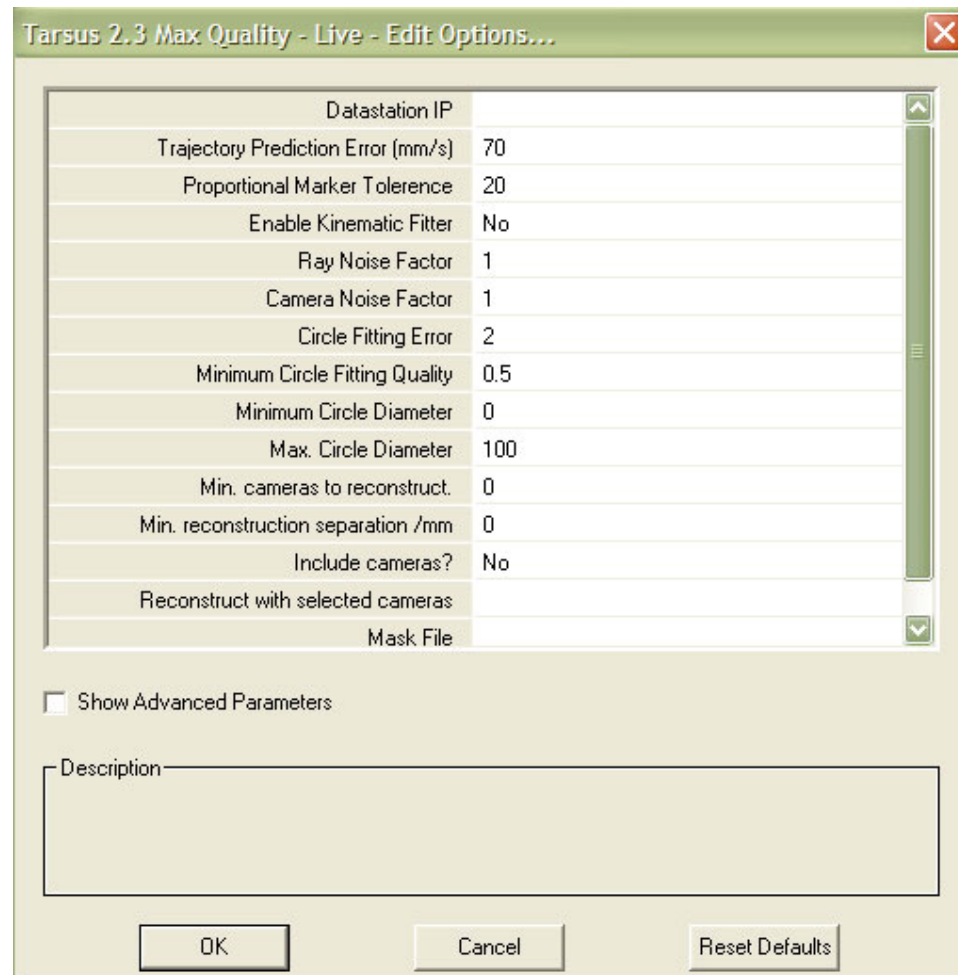
### Step 3: Configuration Options



The Configuration Options section is the most important part of real time. It determines the settings necessary to optimise real time performance.

### Basic Configuration Options

A description of each of the parameters are discussed at the end of this chapter



Here the configuration options control the reconstruction and labelling procedures in real time. Many of the options are identical to those found in Vicon iQ 1.5's Post Process mode for the "Circle Fit, Reconstruct, and Trajectory Fit" Pipeline Operation.

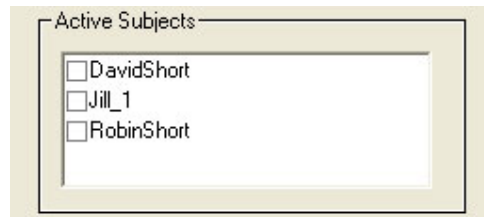
When you make changes to the active configuration, during a Real Time stream, the changes will be immediately applied when you press the OK button. You will see the Real Time Engine restart, and real time capture will commence immediately, with the changes you made, applied.



## TUTORIAL 3: Using the Real Time Mode

In general the default options used by Vicon iQ 1.5, will be good for single person real time capture, with any number of cameras. You will also find several configurations supplied for different real time situations. Use these to start from, and tweak the options until you feel you have reached an optimum setting.

### Step 4: Active Subjects



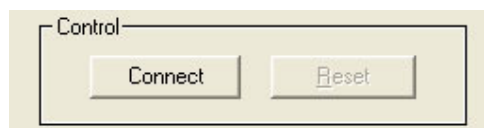
Active subjects are created by subject calibrating a range of motion or by importing in a properly calibrated VSK file from another location on the computer.

Once the VSK is available, it will be displayed in the "Active Subjects" menu.

To label the subject(s) that are in the capture volume, simply tick the box corresponding to the name. When ever you make a change to selected Subjects, Vicon iQ 1.5 and the Real Time Engine will immediately be updated, to reflect the change.

*Note: It is not necessary to have a subject selected for real time. If a subject name is not available or not ticked, it will simply be displayed in the real time workspace as unlabelled 3D points.*

### Step 5: Control



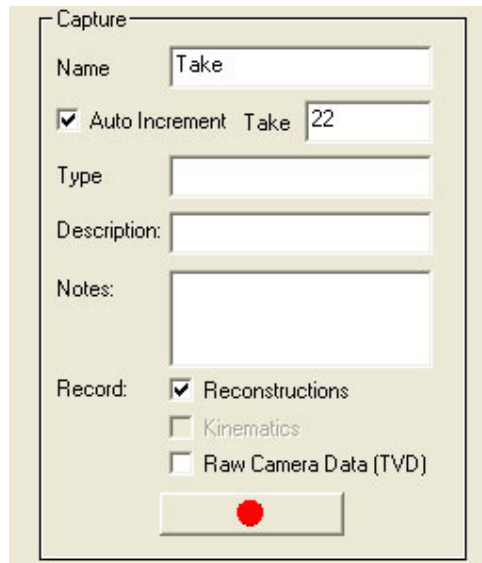
Control takes the settings chosen from the above menus and connects the appropriate devices and sets the chosen parameters.

Once the "Connect" button is pressed, Vicon iQ 1.5, will connect to the Real Time Engine, and will start processing and displaying the Real Time data.

The "Reset" button will help the real time tracker correct itself should mislabelling occur.

## TUTORIAL 3: Using the Real Time Mode

### Step 6: Capture



The capture menu simply controls the capture of data, that is currently being streamed and displayed within the 3D Workspace.

To capture data, enter a trial name in the “Name” field, a take number in the “Take” field. If you wish you can also enter “Notes” and a “Description” in the corresponding fields and select what type of data you want to record.

The reconstruction tick box enables capturing real time data in the frame rate it is reconstructing at, but saved in the Vicon iQ 1.5 “trial file” format at the rate the cameras are set to record at.

Thus, the system is set to run at 120 fps, but real time is running at 80 fps, the trial file will be missing frames of data due to the recorded versus reconstruction difference.

To capture full speed files, the “Raw Camera Data (TVD)” box needs to be checked as well. This will remotely control the Workstation application and record a TVD file that can be reconstructed offline.

Note: The “Raw Camera Data (TVD)” check box will only become active, if you have Activated “Simultaneous” Capture (See above)

# TUTORIAL 3: Using the Real Time Mode

## Chapter 3 - Real-time Visualisation

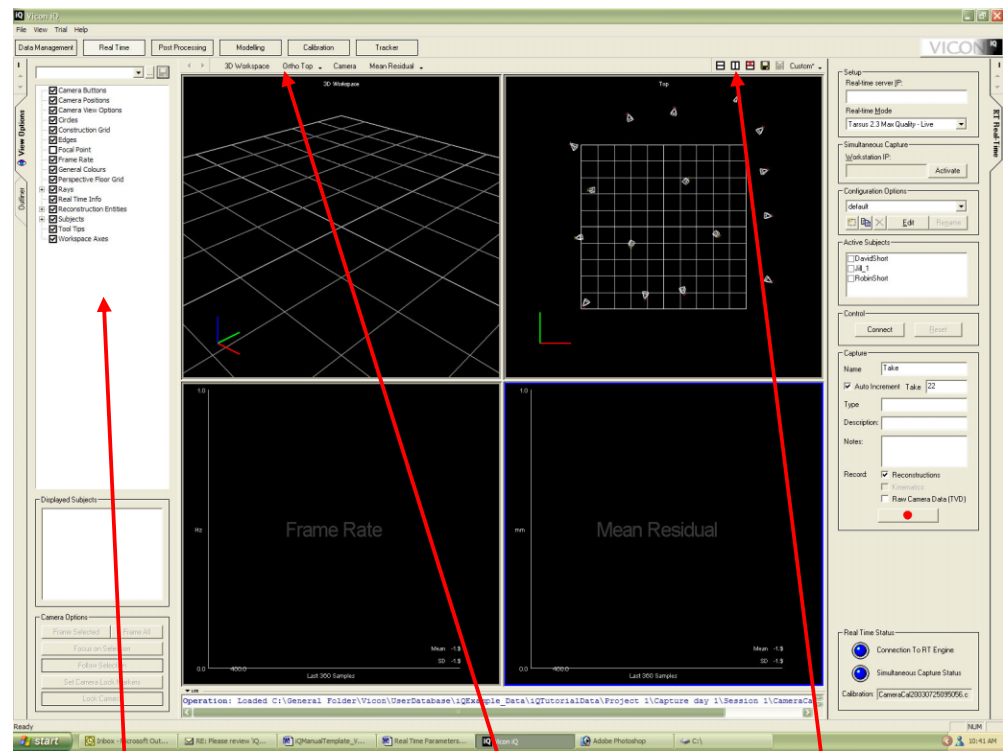
### Introduction

There are many ways to visualise the real time data in Vicon iQ 1.5. With careful use of the 3D workspace, Raw Camera views, diagnostic graphs, View Options, and the Real Time Activity Bar, you can set up and configure you Real Time Mode to display exactly what you want.

Vicon iQ 1.5's visualisation tools allow you to set up your capture volume, and check that your camera calibration is good, and to maintain diagnostic information in real time during your production shoot.

Typically for a production shoot, you will use Vicon iQ 1.5 to visualise your real time data, to capture data on set, in real time, and to provide real time playback. At the same time you will trigger Workstation to capture raw camera data, to be post processed later for optimum data quality.

### Visualising Real Time Data



General View Options

Available View Layouts

Layout Manager

By selecting and using *the Layout Manager* tool buttons, located just above the right hand side of the Workspace Window, custom views can be selected, constructed, edited and saved.

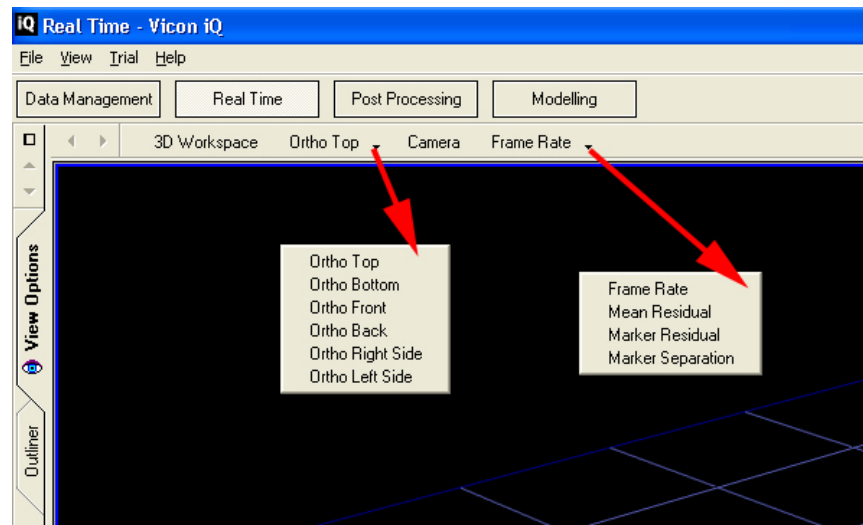
## TUTORIAL 3: Using the Real Time Mode

By Selecting on the divided frame, and then choosing from the *Available View Layouts* from the top left hand portion of the Workspace Window, different views can be chosen for the sub divided workspace windows.

*Tip: It is worth spending some time familiarising yourself with the Workspace, and Layout Manager. You can split the Workspace Window into any number of usable panes, and assign a different view to each pane.*

*Tip2: To quickly maximise any view pane, move the mouse cursor over the view pane of interest, and tap the space bar. To return to the previous view layout, tap the space bar again.*

**View Pane Types** There are several different view pane types, that you can use to visualise the Real Time Data.



### General

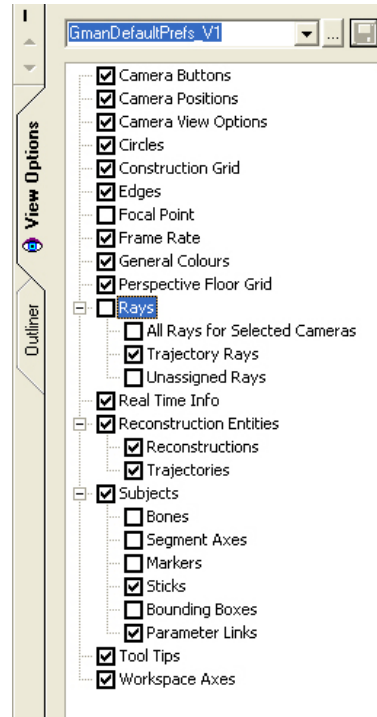
- 3D Workspace. - General 3D perspective window.
- Ortho Views. - Orthographic views, covering all axes.
- Camera - Raw 2D Camera view, displaying real time camera views.

### Diagnostic

- Frame Rate. - Real Time Graph showing actual real time frame rate.
- Mean Residual. - Real Time Graph indicating the mean (approx. average) residual between all markers being reconstructed in the volume.
- Marker Residual. - Real Time Graph indicating the mean (approx. average) residual for the selected markers being reconstructed in the volume.
- Marker Separation. - Real Time Graph indicated the distance (mean and standard deviation) between two markers that are selected in the 3D workspace.

## TUTORIAL 3: Using the Real Time Mode

**View Options** Additional view options can be selected and altered using the global view options located on the left hand side of the 3D workspace. Options like marker size, segment size, joint size, and kinematic skeleton can be altered or turned on or off from this menu.



Vicon iQ 1.5, will allow the display of camera contribution and camera rays for selected reconstructions (View Options->Rays), this can be very useful for diagnosing camera overlap and camera contribution during the set up of a new capture volume and camera arrangement.

The View Option "Real Time Info" allows the control of what information is displayed in the 3D workspace, such as frame rate and mean marker residual.

**Marker Residual** The diagnostic graph views, "Marker Residual", and "Mean Residual", are both very useful indicators of how accurate, and how good the current camera calibration is. The Mean Residual is also displayed in the 3D workspace, in the bottom right hand corner of the workspace window.

Marker residual, is the average error between all the camera rays contributing to a reconstruction. Once you have calibrated a volume, and have some an actor or prop with markers attached in the volume, and being displayed in real time, you can use this measurement to indicate any changes in the camera calibration over time, and also to indicate if you have a good camera calibration.

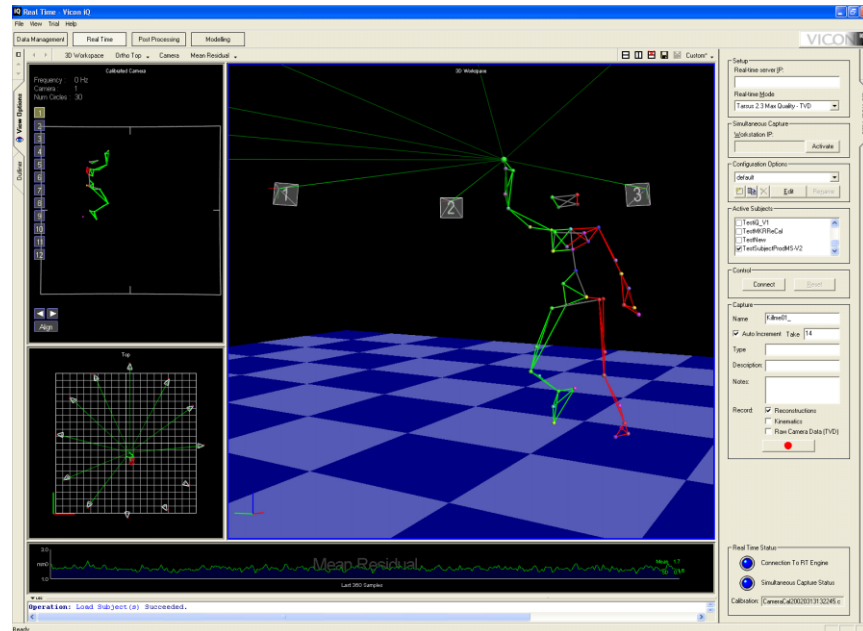
Typically for a full body capture volume, you should expect to see values of around 1.0-2.0. Anything higher, will indicate some camera calibration issues.

## TUTORIAL 3: Using the Real Time Mode

### Chapter 4 - General Workflow using Real Time

#### Introduction

This section is designed to give a brief overview of the two most common workflow scenarios for using Vicon iQ 1.5 during production capture.



#### Real Time Workflow for Broadcast and Virtual Characters.

1. Set up System.
2. Start Workstation.
3. Start Vicon iQ 1.5
4. Start the Real Time Engine (separate PC/CPU)
5. Using Vicon iQ 1.5, Create new Eclipse Project, Capture Day and Session.
6. Using Workstation Check your Camera sensitivities.
7. Using Workstation, capture a 2 second background TVD
8. Using Vicon iQ 1.5, load the Background TVD, and check for reflections, mask any reflections out, and save the camera masks to the active session.
9. Using Workstation calibrate your cameras, and use the MASK file generated in Vicon iQ 1.5 to mask out reflections during camera calibration.
10. Once Camera Calibration is complete, randomly place some markers on the floor of the capture volume, and capture a quick "Floor Plane" trial in Workstation. (this will have the affect of copying the new Camera Calibration (CP) file to the new session.)
11. Load the "Floor Plane trial into Vicon iQ 1.5, reconstruct the floor plane. This will confirm the camera calibration you have just done.
12. Now use Vicon iQ 1.5, to calibrate the floor plane, from the floor plane trial you have just captured, this will update the camera calibration so you have a level

## TUTORIAL 3: Using the Real Time Mode

---

floor. Export the new calibration as a CP file, and refer to this for subsequent trials.

13. Using Vicon iQ 1.5, set up the appropriate real time parameters, and mask file, and connect to the Real Time Engine.
14. Using Vicon iQ 1.5, check that you are getting good unlabelled 3D reconstructions.
15. Using Workstation, set a general trial type, that does not do anything, and only captures TVD data.
16. Using Vicon iQ 1.5, enable simultaneous capture.
17. Using Vicon iQ 1.5 Capture a ROM trial for your actor, ensuring you have checked "Raw Camera Data (TVD)" in Vicon iQ 1.5.
18. Using Vicon iQ 1.5, process the ROM trial, and calibrate your subject. Export the Subject . Test the Subject in Vicon iQ 1.5, by using it to label the ROM trial.
19. Using Vicon iQ 1.5, start Real Time, and use the new subject to label the actor in real time.
20. Start the 3rd party animation package (Filmbox, Maya etc.), and connect to the Real Time stream, using the suitable Vicon Device driver for that animation package.
21. Keep an eye on the Mean Marker residual display in Vicon iQ 1.5, for highlighting any changes to the camera calibration during the shoot.

### Real Time Pre-Visualisation and Capture for Post Production.

1. Set up System.
2. Start Workstation.
3. Start Vicon iQ 1.5
4. Start the Real Time Engine (separate PC/CPU)
5. Using Vicon iQ 1.5, Create new Eclipse Project, Capture Day and Session.
6. Using Workstation Check your Camera sensitivities.
7. Using Workstation, capture a 2 second background TVD
8. Using Vicon iQ 1.5, load the Background TVD, and check for reflections, mask any reflections out, and save the camera masks to the active session.
9. Using Workstation calibrate your cameras, and use the MASK file generated in Vicon iQ 1.5 to mask out reflections during camera calibration.
10. Once Camera Calibration is complete, randomly place some markers on the floor of the capture volume, and capture a quick "Floor Plane" trial in Workstation. (this will have the affect of copying the new Camera Calibration (CP) file to the new session.)
11. Load the "Floor Plane trial into Vicon iQ 1.5, reconstruct the floor plane. This will confirm the camera calibration you have just done.
12. Now use Vicon iQ 1.5, to calibrate the floor plane, from the floor plane trial you have just captured, this will update the camera calibration so you have a level floor. Export the new calibration as a CP file, and refer to this for subsequent trials.
13. Using Vicon iQ 1.5, set up the appropriate real time parameters, and mask file, and connect to the Real Time Engine.
14. Using Vicon iQ 1.5, check that you are getting good unlabelled 3D reconstructions.
15. Using Workstation, set a general trial type, that does not do anything, and only captures TVD data.
16. Using Vicon iQ 1.5, enable simultaneous capture.
17. Using Vicon iQ 1.5 Capture a ROM trial for each of your actors and props, ensuring you have checked "Raw Camera Data (TVD)" in Vicon iQ 1.5.
18. Using Vicon iQ 1.5, process each of the ROM trials, and calibrate your subjects. Export the Subjects to the active eclipse session .Test the Subjects in Vicon iQ 1.5, by using it to label the ROM trials.
19. Depending on the performance of your CPU that you are running the real time engine on, the number of cameras, and the number of actors/props you are capturing, you can optionally use the Real Time Engine to label the actors, or just use it to preview unlabeled reconstructions.
20. Set the record option in Vicon iQ 1.5, to "Raw Camera Data (TVD), and "Reconstructions"
21. Using Vicon iQ 1.5, preview Unlabelled or labelled reconstructions for every capture. When capturing a trial, you will capture a Vicon iQ 1.5 Trial file, and a simultaneous TVD file.



## TUTORIAL 3: Using the Real Time Mode

---

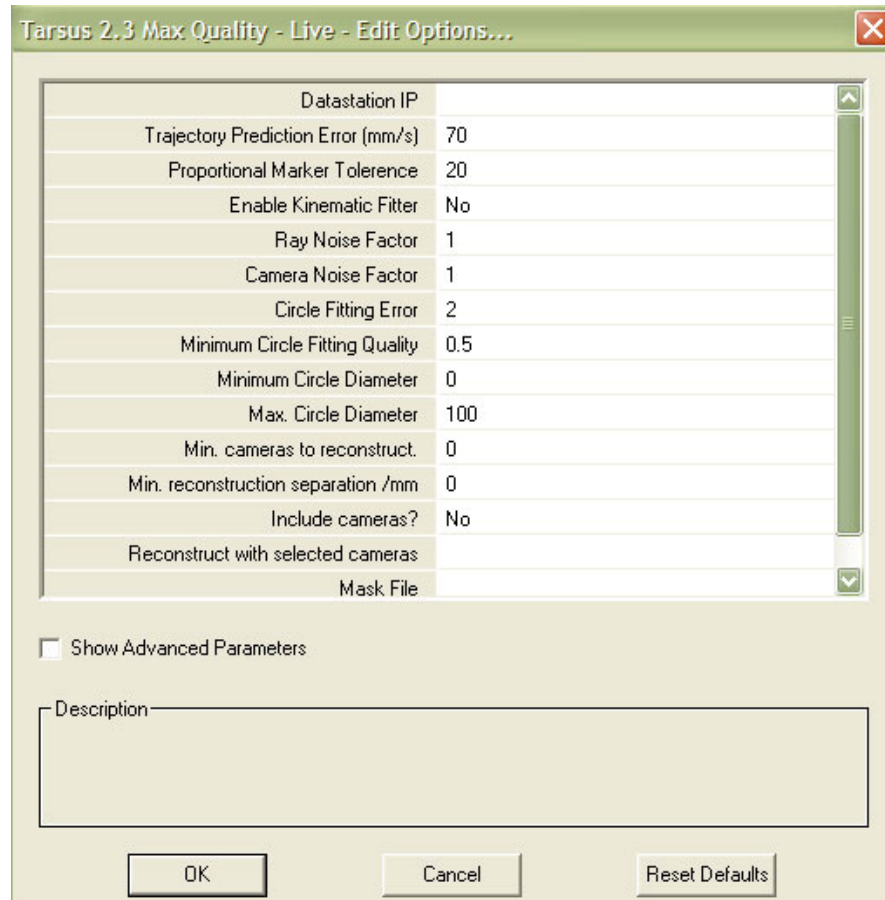


22. Use the Vicon iQ 1.5 trial files for immediate playback on set, and for QA complicated moves on set to make sure you have them.
23. Through out the shoot use Vicon iQ 1.5 to ensure you have good data, and all markers are present.
24. Keep an eye on the Mean Marker residual display in Vicon iQ 1.5, for highlighting any changes to the camera calibration during the shoot.
25. Use Datamangement Mode to enter any extra notes and shot selection information as the shoot progresses.
26. Once the shoot has concluded, use the TVD files to post process in Vicon iQ 1.5 as normal.

## Chapter 5 - Real-time Configuration Options and How to use them

### Introduction

### The Basic Configuration Options



**Datastation IP** This value must be set to the IP address configured on the Vicon Datastation.

**Trajectory Prediction Error** The predictor error expresses an upper limit of the deviation of a single marker from its predicted path based on historical information. The high quality tracking algorithm will attempt to predict the locations of all markers when a new frame comes in. Given that markers move in mysterious ways, the marker is very unlikely to move exactly as predicted. So, the predictor error parameter is used to establish a *radius of uncertainty* around the prediction. If the measured marker falls within this radius, the tracking algorithm concludes that the measured marker is the same as the predicted one.

The purpose of using predictions in reconstructions is chiefly to aid in situations where there are conflicts in ray assignment - in other words, a given ray from one camera intersects two or more rays from another camera within the limit set by the "Ray Noise

## TUTORIAL 3: Using the Real Time Mode

Factor" parameter (see above). In this situation, knowledge of where the marker is most likely to be based on historical information is very likely to resolve the conflict.

The predictions are also used to aid the labeller. In this case, the prediction and its radius of uncertainty will give the labeller a certain number of candidate reconstructions for a given label. If the radius of uncertainty is set such that only one marker falls within this radius the task is very simple, but if two or more fall within the radius things get rapidly more complicated.

As with most parameters, there's a trade-off. If the value is set low, the radius of uncertainty around a given marker is small and a fast moving marker will have a higher likelihood of falling outside this radius. If this happens, the benefits of the prediction goes away and ray conflicts may not be solvable - which again can result in ghost markers and/or mislabelled markers (or simply slow labelling).

Conversely, if the value is set too high, the likelihood of more than one marker falling within the radius of uncertainty increases, which means that the algorithm cannot determine which marker corresponds to the tracked one.

Thus, the correct thing to do is to match the value of the parameter to the type of tracking being attempted. If there's a lot of fast movement, a higher value will be necessary, but if marker movement in general is quite slow a value as low as 35 mm/s can be appropriate.

The default is 70 mm/s which gives reasonable results for a wide range of moves.

### **Proportional Marker Tolerances**

Affects labelling.

The real time labeller labels markers by looking at *distance statistics*. These are pre-calculated values which could either be from the subject calibration or, in case of rigid body tracking, from the object definition. The labeller compares the candidate markers against the pre-calculated known distances between *pairs of markers*.

The information about the pre-calculated (or calibrated) distances between marker pairs is stored in the subject's (or object's) VSK file. The labeller will, when facing a bunch of markers for labelling, calculate the distances from a given marker to all the other ones, and do so for each and every marker. Then, using a clever algorithm, it will try to assign labels based on these distances being similar to the calibrated ones. A distance between two markers is deemed to be similar enough to a calibrated when the measured falls within the calibrated distance plus/minus a threshold.

The Proportional Marker Slack determines part of this threshold by adding a percentage of the calibrated distance. So, if the calibrated distance was 200mm and this parameter is 2%, 4mm will be added to the threshold.

The other part of the threshold is the "Fixed Marker Slack" parameter which is grouped in the advanced section and will be described later.

### **Ray Noise Factor**

The Ray Noise parameter is a factor which allows the user to weight the contribution of the circle fitting uncertainty to the calculation of the standard deviation (SD) used in the "Ray Intersect SDs" parameter. If this parameter is set to 0, the uncertainty in the circle fitting will NOT influence the ray SD whereas a value of 2 will double the uncertainty.

The default value is 1. In some cases, especially when cameras are using very wide angle lenses, it is reasonable to assume that the extra field of view introduces extra uncertainty in the circle fitting algorithm. In this case, a higher weight than 1 may be appropriate for this parameter.

## TUTORIAL 3: Using the Real Time Mode

---

**Camera Noise Factor** The Camera Noise parameter is directly equivalent to the Ray Noise parameter above, except that it is a factor that is applied to the uncertainty in camera position from the calibration. If the camera position is deemed to be very accurate you may choose to not let this influence the ray's uncertainty at all by setting this parameter to 0. Its default value is set at 1.

**Circle Fitting Error** The circle fitting parameter is a threshold measured in camera pixels. When doing the circle fitting, the algorithm will propose a candidate circle centre based on three randomly chosen edges of any given blob - a blob being defined as a connected area of above-threshold data from the camera.

The ideal circle given the proposed centre and its radius is then evaluated against the remaining edges in the blob. If the actual edge is within the ideal edge plus or minus the circle slack parameter, the edge in question will be assigned to the circle. When a high enough proportion of edges in any given blob has been assigned to a circle, the centre of the circle is calculated using all the assigned edges (rather than just the original three, which were used to come up with a candidate only).

The necessary proportion of edges to become a circle is discussed in the next parameter. Also see the next parameter for a full discussion of how these two parameters can be combined to enhance certain aspects of the circle fitting.

**Minimum Circle Fitting Quality** The Minimum Circle Fitting Quality parameter specifies the proportion of edges from a blob that is necessary to be allocated to a candidate circle centre before it is actually flagged as a valid circle centre. For example, a value of 0.5 states that 50% of edges in any given circle needs to be allocated to a centre candidate. If the edge assignment proportion falls below this threshold, which could happen if the blob is non-circular for whatever reason (say if two markers are merging in the image), the circle fit algorithm will discard the candidate centre and choose three new edges at random to generate a new candidate.

The Minimum Circle Fitting Quality and the Circle Slack parameters work very much in partnership. If the slack parameter is high, you increase the likelihood of any given edge being included in any given centre candidate. This means that blobs that aren't very circular will be resolved as a circle regardless. It also increases the likelihood that two merging markers will be resolved as a single one, especially when they get close to perfectly overlapping.

Clearly, if the circle slack parameter is quite high, the Minimum Circle Fitting Quality parameter can and should be somewhat higher as well, otherwise you may end up in a situation where almost any blob is resolved as a single circle.

To improve the circle fitter's ability to resolve merging markers you may try to lower the circle slack parameter. This means that edges are much less likely to be assigned to a given circle centre candidate, and the ability to resolve two merging markers will, in theory, improve. However, as is usually the case with all parameters, there's a trade off. If the input data is quite noisy (which could happen if the markers are far away, partially occluded, whatever), tightening this parameter may mean that the circle fitter stops resolving the marker blob altogether, which means that the camera in question will not contribute to the reconstruction.

Furthermore, if your aim is to improve the characteristics of marker merge, you will need to set the Minimum Circle Fitting Quality parameter to a value less than 0.5. The reason for this is quite obvious - if two circles are merged, neither can claim more than half the edges and any value greater than 0.5 will result in, at best, one of the two merged markers being resolved. A value of 0.3 to 0.4 will probably give reasonable results for merged markers.

## TUTORIAL 3: Using the Real Time Mode

---

- Minimum Circle Fitting Diameter** This parameter allows the user to specify a minimum 2D size of circle blob before they're even considered for circle fitting. A value of 0 or 1 means that the circle fitter will have a go at any blob, whereas a value of 2 will mean that the circle fitter only attempts to circle fit blobs that are at least 2 *scan lines* high.
- Maximum Circle Fit Diameter** This parameter is equivalent to Minimum Circle Fitting Diameter, except that it sets a *maximum* number of scan lines for a given blob before the circle fitter attempts to process it. The prime purpose of this is to filter out unwanted noise such as reflections from body parts that are too close to the cameras.
- Min. Cameras to Reconstruct** This parameter allows the user to specify a minimum number of cameras to contribute to a 3D marker position before it is reconstructed. The purpose of this is to make sure that only high confidence markers are reconstructed. In some cases, when there are lots of cameras and lots of markers, if certain cameras are far away from the markers or are slightly out of calibration they can "team up" with other cameras and create ghost markers. By requiring all markers to have contributions from at least 3 or more cameras, it is possible to filter out 2-camera reconstructions which are likely to be ghosts at a very early stage. This parameter is especially useful for high camera count situations or if markers can be tracked with no occlusions.
- Min. Reconstruction Separation** This parameter allows the user to specify a minimum distance one circle fit must be from another before it is reconstructed. The value is in millimetres.
- Include Cameras** This parameter relates directly to the next parameter called "Reconstruct with selected cameras".
- If this parameter is "yes" and there's a list of camera numbers in the next parameter, the reconstructor will *include only the listed cameras*.
- If this parameter is "no" and there's a list of camera numbers in the next parameter the reconstructor will *include all cameras that are not listed*.
- Mask File** Allows the user to browse for a mask file to be used during reconstruction. When left blank it will not be used.

### Advanced Configuration Options

**Predictor Order** This parameter determines which model the real time engine will use when trying to predict the location of a given reconstruction in the next frame.

Constant position means simply that the prediction will happen exactly at the same location the last position was measured.

Constant velocity means that the prediction will assume that the marker has constant velocity, and will thus predict the marker to have moved exactly the same length in exactly the same direction as between the two previous samples. This is equivalent to a first order polynomial extrapolation (i.e. a straight line).

Constant acceleration means that the prediction will look at the last three samples and assume that the marker maintains the same acceleration it did during those samples. This is equivalent to a second order polynomial extrapolation.

Constant jerk is also known as constant change of acceleration, and means that the prediction is based on a third order polynomial extrapolation.

The default is constant acceleration, and this gives reasonable results for most common moves. However, if the moves are especially jerky it may be advisable to attempt constant jerk. Conversely, if the moves are very slow, it may be enough to go for a constant velocity prediction.

**Skip Rate** This parameter is mainly used when streaming real time from TVD. Setting the skip rate to 0 means that the TVD data is processed as if it was captured real time, in other words the camera data is fed to the real time engine at the same rate as if it came straight from the cameras. Setting the skip rate to 1, however, means that the real time engine processes each and every frame from the TVD file as fast as it can. This could mean that the processing happens either slower or faster than what would be real time - depending on how much data the TVD contains and, of course, the processing hardware.

**Fixed Marker Tolerance** The fixed marker tolerance parameter corresponds to the proportional one mentioned earlier, except that the fixed marker slack is not dependent on the distance between the two candidate markers. Thus, the combined marker tolerance is the fixed slack plus the proportional slack.

**Marker Statistics Threshold** Affects labelling.  
This parameter is used by the labeller to determine at what level a likelihood for two markers to be a "label pair" becomes 0. The higher this value, the more labelling hypotheses will be discarded because the likelihood falls below the threshold and is set to exactly zero.

This means that a low value will result in higher quality labelling because even very unlikely hypotheses are not discarded, but it also means that the algorithm is slower and consumes more memory (since it has to keep track of all the hypotheses).

The correct value of the parameter is the usual trade-off between quality and speed. A high value such as  $1e-2$  (i.e. 0.01) could well be appropriate for complex labelling problems.

## TUTORIAL 3: Using the Real Time Mode

---

- Tolerance on Joint Placement** Affects labelling, but only for multi-segment models linked by joints.  
This parameter states how much a joint position is allowed to deviate from its calibrated position in the kinematic model that drives the labelling model. This parameter is equivalent to the fixed marker slack parameter in that it adds a set distance (slack) to a given position.
- Proportional Joint Limit Tolerance** Affects labelling, but only for multi-segment models linked by joints.  
This parameter indicates how much beyond the measured range of motion a joint is “allowed” to move beyond the calibrated range of motion. A value of 0.1 for this parameter means that the subject’s movement during the tracking is allowed to exceed the calibrated range of motion by a factor of 0.1, i.e. 10%.
- Start Labelling and Labelling Increase Ratio** Affects labelling  
The Start Labelling Rate is a statistical measure which is used iteratively in the labelling process to make sure that the labeller iterates towards the correct solution. In other words, the Labelling Rate factor will increase for each labelling iteration and every time this happens, wrong labelling hypotheses will be more likely to be discarded.  
The Start Labelling Rate parameter works together with the Labelling Rate Increase Ratio parameter (see below) to establish the iterative “path” of the Start Labelling Rate factor. The start value is the Labelling Rate parameter for the first iteration, whereas the start value will be multiplied by the Labelling Rate Increase Ratio for the next one. Thus, if you use the value 1.0 and 2.0 for Start Labelling Rate and Labelling Rate Increase Ratio respectively, the beta values will be 1-2-4-8-16-etc. The Beta factor is increased iteratively until the labeller has made its mind up, so the sooner the Beta factor reaches a high value, the faster the labeller will produce a result.  
As usual, there’s the speed versus quality trade-off. By starting at a low Beta factor you will basically tell the labeller to consider very many hypotheses from the very start. Then, by having a low Labelling Rate Increase Ratio, you tell the labeller to consider each step very carefully. The point of doing this is that the labeller is continuously adjusting probabilities and weights as it goes along. Thus, a low Start Labelling Rate value combined with a low Labelling Rate Increase Ratio will mean that the end result is more likely to be correct.  
Conversely, a high Start Labelling Rate and a high Labelling Rate Increase Ratio means that the labeller will produce a result faster, but that the result is more likely to be wrong. A low Start Labelling Rate and a high Labelling Rate Increase Ratio will mean that the labeller will consider many hypotheses initially, but “weed” them very quickly, whereas a high Start Labelling Rate combined with a low Labelling Rate Increase Ratio will make the labeller consider a lower number of hypotheses more carefully.  
In general, it is a trade-off versus speed and quality, so keeping both low to maximise quality or both high to maximise speed is the correct thing to do.
- Maximum Iterations of Kinematic Fitter** Affects kinematic fitting  
The maximum iterations of the kinematic fitter tells the real time tracking algorithm how many iterations it is allowed to do in order to arrive at a converged solution to the fitting problem. The higher this number, the more closely the kinematic fitting will be to the calibrated kinematic model, but the longer the tracking will take. The default is 1, for fast kinematic fitting.

## TUTORIAL 3: Using the Real Time Mode

---

<b>Kinematic Prediction Order</b>	<p>Affects kinematic filtering</p> <p>This value tells the real time engine how much filtering to apply to the kinematically fitted results. The value 0 means that the filter is off, whereas 1 means that the Kalman filter uses a constant position filtering model, 2 means that the filter uses a constant velocity filtering model.</p> <p>As far as Kalman filtering is concerned, a constant position model is appropriate if steady <i>positional</i> data is required. For example, if you are tracking an interaction object that is used to identify positional data (such as pointing at virtual objects), the correct filter method is probably constant position filtering as this will mean that the pointer will be most steady. The trade-off is that the filter will, to a certain extent (determined by the parameters below) <i>resist</i> movement which means that the virtual object will, to a greater or lesser extent, lag behind the real object in the virtual world.</p> <p>However, if smooth movement is more important than steady positioning, constant velocity filtering may be appropriate. This will mean that the virtual object will not lag behind the real one when the object is moving, but will also mean that the virtual object is less steady when it is stationary.</p> <p>One thing to note about Kalman filtering is that using it in conjunction with a higher Maximum Iterations of Kinematic Fitter value may be a good idea.</p>
<b>Kinematic Prediction Revolute</b>	<p>Affects kinematic filtering</p> <p>The lower this value, the more filtering is applied to the rotational aspect of the tracked object(s). A value of 1.0 gives a very moderate amount of filtering, whereas a value of around 0.3 filters rather heavily. The value has to be set according to the motion of the tracked object or subject. If the object will experience rapid rotations, a low value for this parameter will mean that the tracked object will "lag behind". In other words, a low filter parameter will prevent the virtual version of the tracked object from being rotated very fast.</p>
<b>Kinematic Prediction Translation</b>	<p>Affects kinematic filtering</p> <p>The lower this value, the more filtering is applied to the translational aspect of the tracked object(s). A value of 100 mm/s gives a moderate amount of filtering whereas a value as low as 30 mm/s means that the virtual object will "resist" rapid translational movements. The correct value of this filter depends directly on the speed of the object's movements. If the object moves fast, too low a value will result in its virtual representation lagging behind. Conversely, too high a value will fail to filter out some of the noise.</p>
<b>Marker Stickiness When Previously Unlabelled</b>	<p>Affects labelling</p> <p>This parameter is used by the labeller to determine how likely it is that a given marker which was unlabelled in the last frame is still missing. Thus, a low value such as 1e-20 means that the labeller will try very hard to label a given marker cloud, whereas a low value such as 1e-10 will instruct the labeller to give up much quicker and rather leave markers unlabelled.</p> <p>In other words, the trade-off means that you have the choice between labelling the best fit markers and leaving the others unlabelled - or forcing the labeller to try really hard but run the risk of the labeller choosing wrong labels for some markers in its quest to label everything.</p>



## TUTORIAL 3: Using the Real Time Mode

---

**Marker Stickiness  
When Previously  
Labelled**

This parameter is used by the labeller to determine the likelihood of a previously labelled marker now having gone missing. A low value, such as  $1e-20$ , tells the labeller that markers, once labelled, are overall extremely unlikely to go missing and the labeller will try its utmost to come up with a fully labelled solution.

A higher value tells the labeller that markers are not unlikely to go missing and that it's better to leave markers unlabelled rather than choosing the wrong solution.

The correct value depends on the type of move. If the subject's markers will often be occluded - say, when rolling around on the floor - a low value will not be appropriate. However, in situations where markers are very unlikely to be occluded, go out of the volume or otherwise end up in positions where they cannot be reconstructed, the value of this parameter can be lowered.