OptiTrack

NatNet API

User's Guide

Version 3.1
2018

Technical support
help.naturalpoint.com
1-888-965-0435

TABLE OF CONTENTS

NATNET DOCUMENTATION

This PDF document contains only the instructions on running the sample applications included in the SDK. For most up-to-date documentation of the NatNet SDK, please visit the NatNet SDK page on the OptiTrack documentation wiki:

- Documentation Wiki: NatNet SDK

The wiki page contains information on the NatNet SDK contents, class references, update notes, and instructions on using the SDK for receiving tracking data from a server application, Motive.

GENERAL NOTES

- Supported Platform:
    - Windows: 32-bit and 64-bit. Ships in DLL format
- The NatNet SDK library is designed to work with **Motive only**. Motive must be used as the tracking server in order to connect and stream tracking data into client applications that are developed against the SDK.
- For NatNet SDK versions 3.0 and above, *Tracking Tools* or *Arena* cannot be used as a tracking server. When working with a legacy software, use SDK versions below 3.0.
- Direct depacketization protocol could be used for streaming onto unsupported platforms and parsing the tracking data directly from the bit-stream. However, the bit-stream syntax is subject to change in future releases.

NATNET SDK CLIENT SAMPLES

The following projects are located in the *Samples* folder, and they utilizes NatNet SDK library for obtaining tracking data from a connected server application. ple only)

Sample	NatNet Platform	Description
Matlab	Managed: Matlab	Sample MATLAB code file (.m) for using MATLAB with the NatNet managed assembly (NatNetML.dll) through the provided *natnet.p* wrapper class.
RebroadcastMotiveData	Native: C++	Sample application that receives tracking data from the NatNet Server and redistributing it in other forms. Currently there are two protocols supported for pipelining tracking data into Unity via compatible XML packets and into Previzion software via serial port interface.
MayaPlugIn	Native: C++	GitHub open source project: mayaMotive
SampleClient	Native: C++	Sample NatNet console app that connects to a NatNet server, receives a data stream, and writes that data stream to an ASCII file. This sample also demonstrates
SampleClient3D	Native: C++	Sample NatNet application that connects to a NatNet server, receives a data stream, and displays that data in an OpenGL 3D window.
SampleClientML	Managed: .NET (C#)	Sample NatNet C# console appication that connects to a NatNet server on the local IP address, receives data stream, and outputs the received data.
TimingClient	Native: C++	This program connects to a NatNet server and can be used as a quick check to determine packet timing information.
WinFormsSample	Managed: .NET (C#)	Simple C# .NET sample showing how to use the NatNet managed assembly (NatNetML.dll). This sample also demonstrates how to send and receive the NatNet commands.

RUNNING THE CLIENT SAMPLES

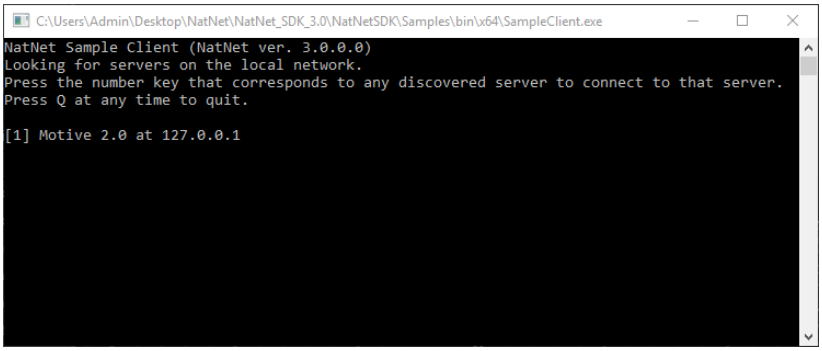
Pre-compiled versions of the NatNet samples have been provided in the *Samples**bin* folder. These versions can be used to quickly test your application. For running the sample applications, following items must be installed on the computer:

- Visual Studio 2015 / Visual C++ Redistributable 2015
- Microsoft .NET framework version 4 (ships with Windows 10)
- Python version 3.6 or above (Only for PythonClient sample)

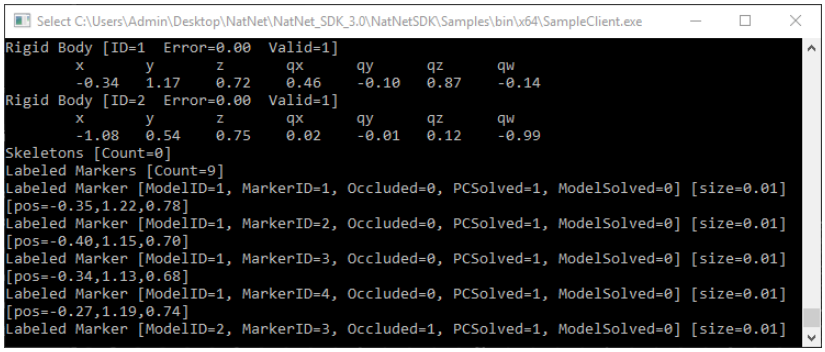
Note! The Visual C++ runtime libraries are required to run the samples. If you encounter an error message when attempting to run the samples, especially on machines without Visual C++ installed, please install the VC runtime redistributable package located in *Samples**VCRedist*. If the problem persists, please try rebuilding the samples using Visual C++, or contact support.

RUNNING THE CONSOLE OUTPUT SAMPLE (SAMPLECLIENT)

1. **[Motive]** Start the Optitrack Server (e.g. Motive) and begin streaming data via the Streaming Panel
2. **[SampleClient]** Start the client application from the command prompt or directly from the *NatNet SDK/Samples/bin* folder.
3. **[SampleClient]** Once the sample application starts up, it will search the local network and list out IP addresses of available servers where tracking data is streamed from. Select a server address by pressing the corresponding number key.



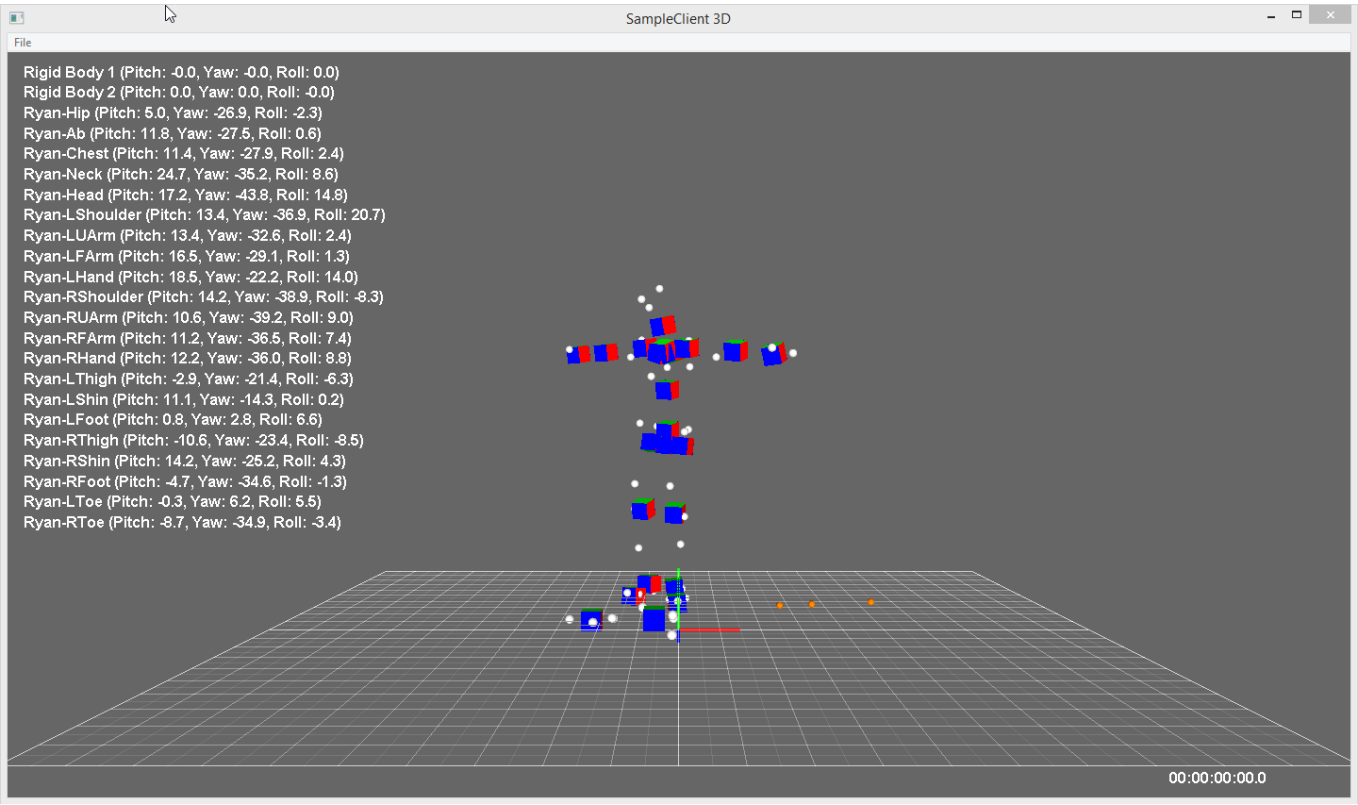
4. **[SampleClient]** The client application connects to the server and you should begin to see data streaming in the client window.



RUNNING THE RIGID BODY SAMPLE (SAMPLECLIENT3D)

The Rigid Body sample (SampleClient3D) illustrates how to decode NatNet 6DOF Rigid Body and Skeleton Segment data from OptiTrack quaternion format to euler angles and display them in a simple OpenGL 3D viewer. This sample also illustrates how to associate RigidBody/Skeleton Segment names and IDs from the data descriptions with the IDs streamed in the FrameOfMocapData packet.

SampleClient3D - Decoding and drawing labeled rigid body position and orientation (6DOF) data



With Client/Server on same machine:

1. **[Motive]** Load a dataset with rigid body or skeleton definitions
2. **[Motive]** Enable network streaming (Data Streaming Pane -> Check Broadcast Frame Data)
3. **[Motive]** Enable streaming rigid body data (check Stream Options -> Stream Rigid Bodies = True)
4. **[Sample3D]** File -> Connect

With Client/Server on separate machines:

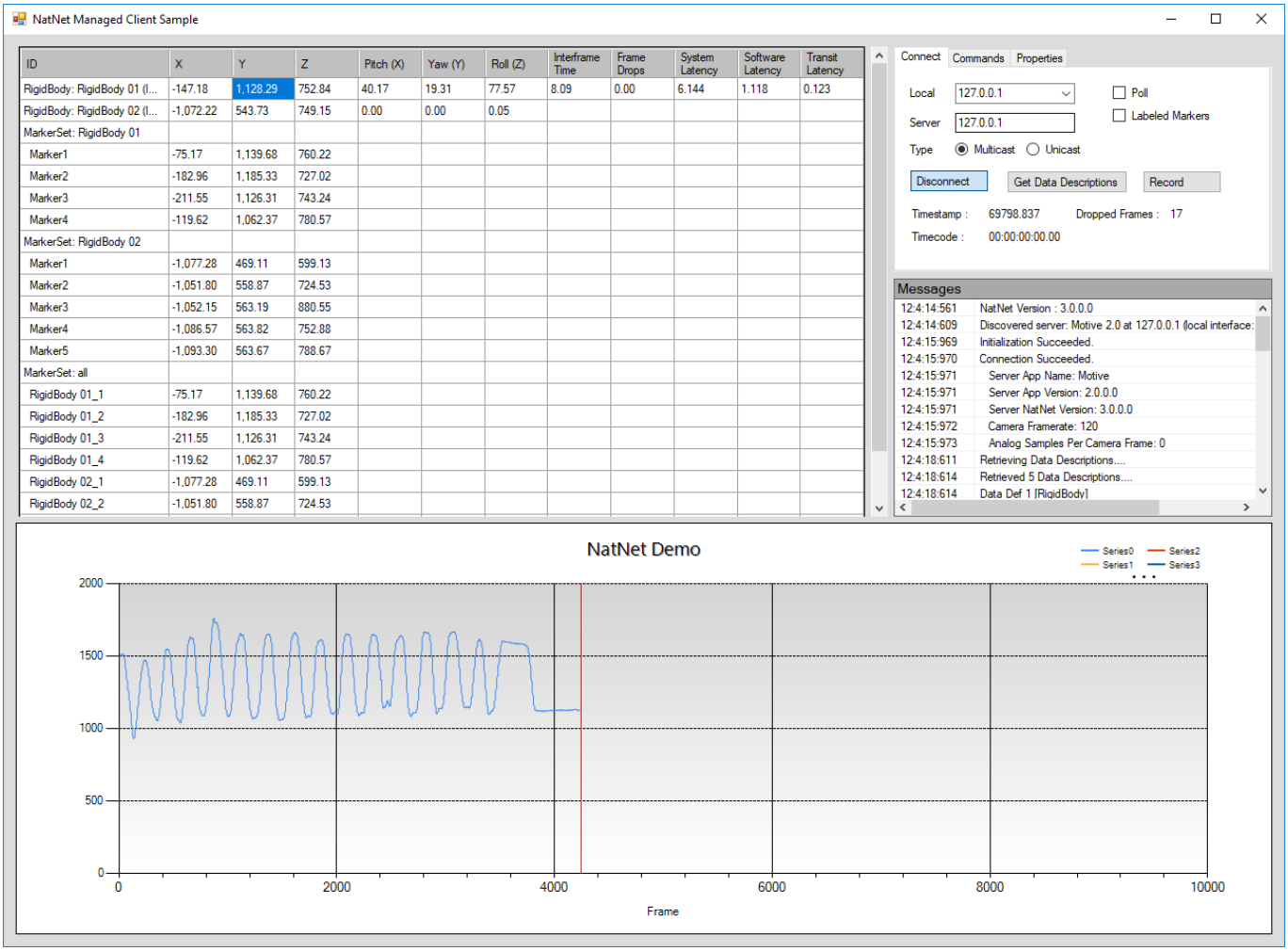
1. **[Motive]** Load a dataset with rigid body or skeleton definitions
2. **[Motive]** Set IP address to stream from (Network Interface Selection -> Local Interface)
3. **[Motive]** Enable network streaming (Data Streaming Pane -> Check Broadcast Frame Data)
4. **[Motive]** Enable streaming rigid body data (check Stream Options -> Stream Rigid Bodies = True)
5. **[Sample3D]** Set Client and Server IP addresses
6. **[Sample3D]** File -> Connect

- **IP Address** IP Address of client NIC card you wish to use.
- **Server IP Address** IP Address of server entered in step 2 above.

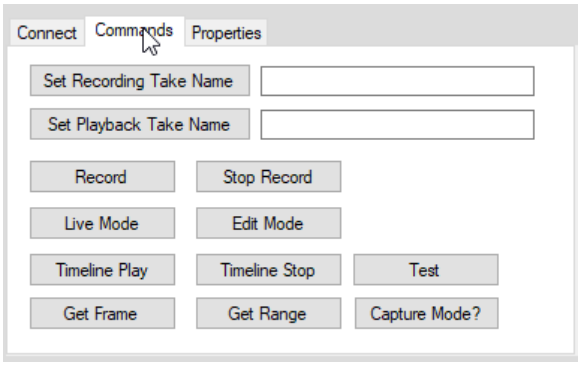
RUNNING THE .NET SAMPLE

1. **[Motive]** Start a NatNet server application (e.g. Motive).
2. **[Motive]** Enable NatNet streaming from the Server application.
3. **[WinFormTestApp]** Start the WinForms sample application from the NatNet Samples folder.
4. **[WinFormTestApp]** Update the “Local” and “Server” IP Addresses as necessary.
5. **[WinFormTestApp]** Press the “Connect” button to connect to the server.
6. **[WinFormTestApp]** Press the “Get Data Descriptions” button to request and display a detailed description of the Server’s currently streamed objects.
7. **[WinFormTestApp]** Select a Row in the DataGrid to display that value in the graph.

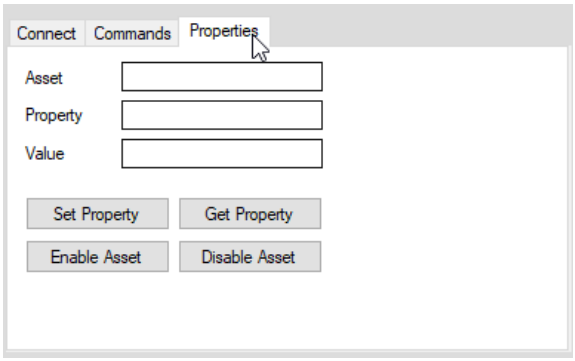
Receiving NatNet data in a .NET Environment



Issuing remote control commands to Motive



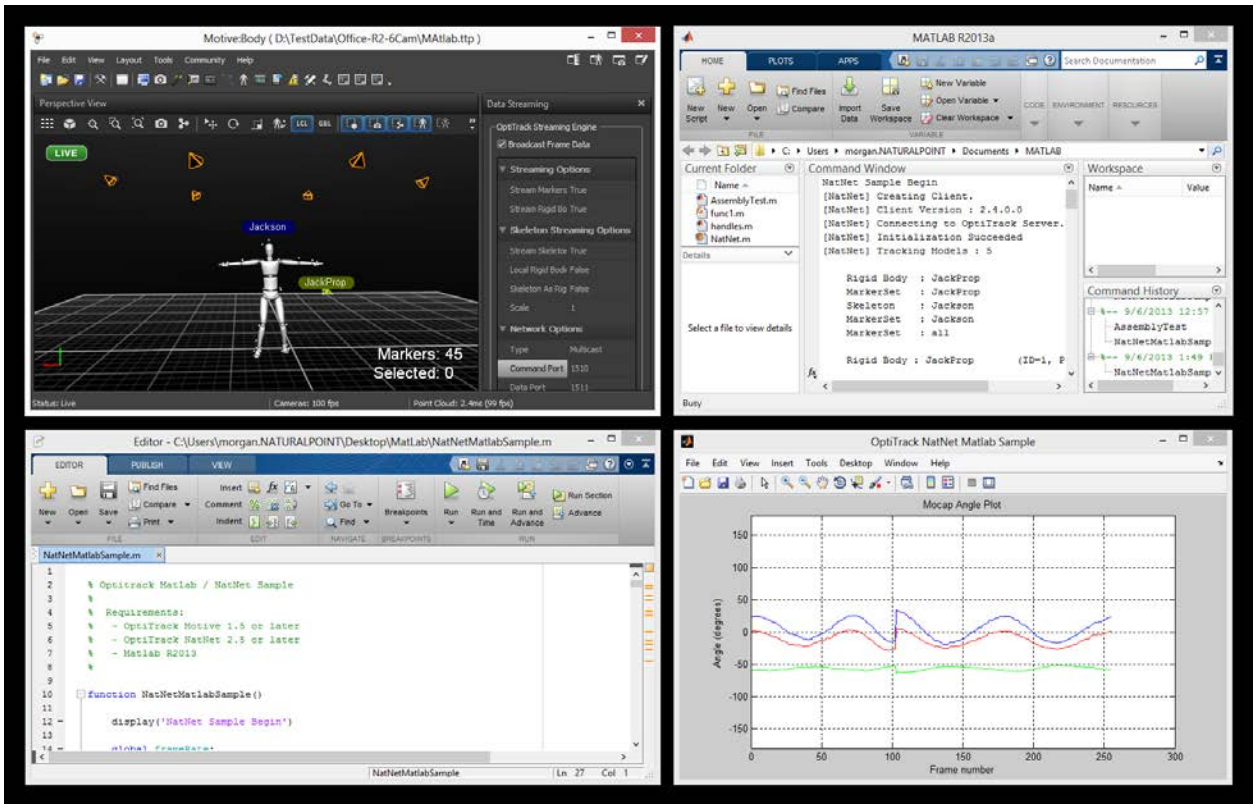
Remote configuration of properties in Motive



RUNNING THE MATLAB SAMPLE

1. **[Motive]** Start a NatNet server application (e.g. Motive).
2. **[Motive]** Enable NatNet streaming from the Server application.
3. **[Matlab]** Start Matlab
4. **[Matlab]** Open the NatNetMatlabSample.m file.
5. **[Matlab]** From the editor window, press **Run**

Real-Time Streaming Mocap data from Motive into MATLAB



TECHNICAL SUPPORT

NaturalPoint is committed to providing best-in-class technical support.

In order to provide you with the most up to date information as quickly as possible, we recommend the following procedure:

1. Update to the latest software. For the latest versions of OptiTrack software, drivers, and SDK samples, please visit our downloads section:

   http://www.optitrack.com/downloads/

2. Check out the OptiTrack FAQs:

   http://www.optitrack.com/support/faq/general.html

3. Check the forums. Very often a similar issue has been reported and solved in the forums:

   http://forum.naturalpoint.com/

4. Contact technical support:

   Phone: 541-753-6645

   Fax: 541-753-6689

   Email Form: http://www.optitrack.com/contact/

   Mail: NaturalPoint, Inc.
   P.O. Box 2317
   Corvallis, OR 97339