

XML Protocol

Common XML Markup Language

Data from various hardware systems is collected in the server and then repackaged in a common XML based language.

The Server accepts a subset of the standard Visual3D pipeline commands and then applies those commands to the real-time hardware data. The results are then packaged up in a common XML based language and broadcast.

All command are nested within the out command `<visual3d>` as follows:

```
<visual3d from="source">
  ... nested commands ...
</visual3d>
```

where the attribute from is either "client" or "server"

The initial `<visual3d>` command is what defines the protocol between clients and server.

Most commands will include a 'status' and possible a 'message' attribute. Status attribute can be one of OK, Warning or Error. The message attribute contains any additional information as related to the status attribute. Client applications can examine these attributes and respond accordingly.

Connect Command

The first XML command a client must send to Visual3DServer is the connect command. This command introduces the client to the sever. Syntax:

```
<visual3d from="client">
  <connect protocol="Biofeedback" name="SampleClientApp" version="1.0"
date="16-Aug-2013" environment="" />
</visual3d>
```

Connect command attributes:

Protocol: describes client purpose for the data. (string, no format)

Name: name of client application (string, no format)

Version: version number of client application (string, no format)

Date: date when client was created (string, no format)

Environment: various details about client environment (e.g., computer type, platform) (string, no format)

The server will echo the message back to the client, indicating if the command was accepted. If a TCP connection was made, the server will tell the client the UDP broadcast port number over which real-time data will arrive.

```
<visual3d from="server" status="ok">
  <connect status="ok" broadcastport="60283" message="127.0.0.1" />
</visual3d>
```

Server response command attributes:

Connect Status: will be either “ok” indicating a successful connection or “error” or “warning” indicating that there was a problem with the connection.

Broadcastport: if the connection was successful this will show the port number that the Server is broadcasting on.

Message: if the connection was successful the message will be the IP address that the server is operating from. If there was an error or a warning the message will provide further details.

Note 1: If there is a hardware system connected to the Server when the connect command from the client is received then the Server will respond with a combination of the connect command and the systemsget command, and example of which can be seen below in the systemsget section.

systemsget Command

From time to time the server will send all clients the systemsget command, which indicates when there has been a change in the available hardware data sources. Syntax:

```
<visual3d from="server" status="ok">
  <systemsget>
    <system value="C3D File" />
    <system value="Vicon DataStream" />
  </systemsget>
</visual3d>
```

Server Response Systemsget Command attributes:

System value: there will be one system value line for each of the hardware systems connected to the Server and it will provide the name of the hardware system connected.

Note 1: This is the only command that is “pushed” to clients. The Server will generate a new systemsget Command whenever a hardware system is connected or disconnected. All other server commands are in a one-to-one response to client commands.

Note 2: When the server responds to a connect command, the server includes which hardware data sources are available at the time of connection. This is done for convenience and results in the message below which is a combination of the connect and systemsget command.

```
<visual3d from="server" status="ok">
  <connect status="ok" broadcastport="60283" message="127.0.0.1" />
  <systemsget>
    <system value="C3D File" />
    <system value="Vicon DataStream" />
  </systemsget>
</connect>
```

```
</visual3d from="server" status="ok">
```

systemsset Command

A client tells the server which hardware systems in which to receive data from by sending the systemsset command. Syntax:

```
<visual3d from="client" status="ok">  
  <systemsset>  
    <system name="C3D File" />  
  </systemsset>  
</visual3d>
```

Client Systemsset Command attributes:

System name: this is the hardware system that the client is requesting data from.

Server response:

```
<visual3d from="server" status="ok">  
  <systemsset status="ok" value="1" />  
</visual3d>
```

Server Systemsset Response attributes:

Systemsset status: will return "ok" if the data from the requested hardware system is being sent.

Value: this will indicate the number of hardware systems which are sending data to the client.

Note: The attribute **value** indicates how many hardware systems are engaged by the particular client.

Pipelineset Command

A sample of a possible fifth XML command is below. It is generated by the client and tells Visual3DServer which pipeline should be applied to the data that it is asking for. Syntax:

```
<visual3d from="client" status="ok">  
  <pipelineset  
value="Compute_Model_Based_Data/RESULT_NAME=LABDATA/FUNCTION=CLIENT_SYSTEM;"  
  />  
</visual3d>
```

Client Pipelineset Command attributes:

Pipelineset value: this is the pipeline command that will be used to process the data being exchanged.

A sample of a possible sixth XML command is below. It is generated by Visual3DServer and confirms that a valid pipeline request was sent and then sends the requested data to the client. Syntax:

```
<visual3d from="server" status="ok">
  <pipeline set status="ok" message="" />
</visual3d>
```

Server Pipelineset Response attributes:

Pipelineset status: status will be “ok” if the pipeline that has been sent is acceptable. If there are problems with the pipeline the server will return either “warning” or “error”.

Message: if the pipeline status is “ok” there will be no message, however if either “warning” or “error” are returned message will provide further details.

From:
<https://has-motion.com/wiki/> - Wiki Documentation Site

Permanent link:
https://has-motion.com/wiki/doku.php?id=other:visual3dserver:documentation:xml_protocol&rev=1721230991

Last update: 2024/07/17 15:43

