2013.7.23:

The most recent development version of VESC no longer provides OSC access to the haptics device by default. To get direct access to the haptics device (both for OSC messaging and routing in your scene), include

<IMPORT inlineDEF='H3D\_EXPORTS' exportedDEF='HDEV' AS='HDEV'/>

in your scene. You can then route to and from 'HDEV' and send/receive OSC messages via the /VESC/HDEV/ container. If you want to rename the device, change the AS='HDEV' part to your liking.

The intent moving forward is that most access to the haptics device will be done indirectly via X3D prototypes.

* Karl

4/10/13:

It occurred to me that careless use of 'inputOutput' fields in your prototypes can also be a problem.

Suppose you have a prototype with a 'trackerPosition' inputOutput field in the ProtoInterface, for the purposes of getting the tracker position "into" your prototype, and that you ROUTE from the haptics device node into your prototype instance:

<IMPORT inlineDEF='H3D\_EXPORTS' exportedDEF='HDEV' AS='HDEV'/>

<ROUTE fromNode='HDEV' fromField='trackerPosition' toNode='my\_proto\_instance' toField='trackerPosition'/>

Then, suppose you sent:

/node/\*/registerFieldInterest add

to VESC.

You'd get OSC message updates from HDEV \*and\* from your prototype instance. If trackerPosition is updating once per frame, then the HDEV node is generating 60 OSC messages per second. Having 'trackerPosition' declared as inputOutput in the prototype unnecessarily doubles VESC's output to 120 OSC messages per second.

Now imagine that you have 10 instances of your prototype in the scene. VESC's message output increases by more than an order of magnitude, from 60 to over 600!

I guess as a general rule, if a field in your ProtoInterface exists only for the purpose of getting events into the prototype, make it inputOnly.

- Karl