# The Virtual Universe Motion Assistant

(C)2011 IRAI

## Main concept

The "motion assistant" introduced in the 1.100 Virtual Universe version allows to create quickly and easily 3D objects motion and detection. "motion assistant" automatically generates behaviors from created configuration. Once behaviors have been generated, it is possible to return in the "motion assistant" to modify the configuration.

## Opening "motion assistant"

To open the "motion assistant", select a child 3D sprite in the setup window and click on the "motion assistant" green button on the left side.

 

## "setting" mode, "try" mode

In the "motion assistant" window, a "two states" Try button allows to switch between a "setting" mode (not pushed button): modifying motions and sensor properties, to a "try" (pushed button): test motions and sensors.

## "motion assistant" window

Select the motion type

Name of the currently edited sprit3d

 

Show the sensors positions and state (in "try" mode

Allow to define sensors and display their state in "try" mode

Allow to make manually the motion in "setting mode", show the motion position in "try" mode

Define motion limits

Switch between "setting" and "try" mode

Change the motion commande type

Close the "motion assistant"

The other items depend on the type of motion and will be explained below

## Definition of minimum and maximum positions

It is possible to enter directly positions values in "Min" and "max" edit boxes.

 

It is also possible to move or rotate the 3D sprite, by using the trihedral in the 3D view, to the desired position and then to click on one of the "Set" buttons:



Above example: move by clicking and dragging the red arrow, then click "Set":

 

"Min" position must be less than "max" position.

3D sprite position will be automatically reset when "motion assistant" will be closed or when "try" mode will be activated.

## Differences between "Force" and "Move" modes

"motion assistant" permits two main modes: "Force" and "Move".

"Force" mode defines a motion which uses completely the physical engine: the motion is created by forces applied to the 3d sprite.

"Move" mode does not use the physical engine to create the motion. Motion is imposed to the 3d sprite from different criteria defined in the "motion assistant": time, speed or position. Object position is still sent to the physical engine, so 3d sprite moves can interact with other 3D worlds objects.

To summarize, we can say that the "Force" mode is more realistic, and the "Move" mode easier to implement. It will be necessary to choose one or the other of these modes depending on the purpose (= speed of development "Move" or physical realism = "Force").

## "Force" Mode"

With this mode, sprite3d motion is created with two forces (for translations) or two net torques (for rotation). The physical engine manages the motion.

 

Force or net torque resulting in a motion toward the maximum position

Force or net torque resulting in a motion toward the minimum position

Axis on which
movement will take place, axis of rotation for rotations

Joint type

Notes:

Buttons to initiate or stop the application of each force or net torque in "try" mode. The left button is related to the "minimum" force or net torque, the right to "maximum"

Logarithmic slider for adjusting the force or net torque in "setting" or "try" mode. The slider on the left is related to the "minimum" force or net torque, the right to "maximum"

- in "try" mode, enabling a force or a net torque by clicking on "Apply" and then adjusting the force or the net torque with the associated slider is a good way for setting the motion speed,

- physical characteristics (eg. mass) of the object affect behavior. To modify those characteristics, open the "physic" headings in the properties of the 3d sprite in the setup window.



### Behavior and joints generated by "Force" mode



The "motion assistant" generates automatically the joint between the 3d sprite and his parent and two opposite "Local force" or "Local net torque" behaviors. "Use physic" parent attribute is also modified.

## "Move" mode

In this mode, motion is imposed by one or more behaviors generated by the "motion assistant".

### "Time" sub mode

With this sub mode ("Move / time"), motion is defined by time taken to perform the total move in one direction and the other (two values must be set).

Times have to be written in second.

The motion is stopped when 3d sprite reaches minimum or maximum position.

 

Time to make the movement from "min" position to "max" position

Time to make the movement from "max" position to "min" position

The "Move / Time" sub mode has two possible command types (selectable by clicking on the "..." button"):



#### "One command"

Will drive the motion depending on a boolean state: true=motion toward max, false=motion toward min.

For this command type, in "try" mode, the "Activate" button causes, if pushed, motion toward "max" position or toward "min" position if released:

 

In this case ("One command"), the "motion assistant" generates one "move" or "rotate" behavior. The boolean state of this behavior drive the motion:



#### "Two commands"

Will drive the motion from two boolean states. The following show link between the two boolean states and the motion (in this table: 0=false, 1=true):

|  |  |  |
| --- | --- | --- |
| Boolean state 1 (max toward min) | Boolean state 2 (min toward max) | Motion |
| 0 | 0 | None |
| 0 | 1 | Toward max |
| 1 | 0 | Toward min |
| 1 | 1 | None |

For this command type ("two commands"), in "try" mode, "Activate min" and "Activate max" buttons activate the two boolean states.

 

In this case ("two commands"), the "motion assistant" generates a "move" or "rotate" behavior and two "generic read" behaviors associated to the two boolean states used to drive the motion. The state 0/1 of these two behaviors drives the motion.



### "speed" sub mode

With this sub mode ("Move / Speed"), the motion is defined by a speed.

There is two possible commands for this sub mode:

 

Cancel

#### "Behavior value defines speed"

Will drive the motion from a value which is defining the speed.

For this type of command, in "try" mode, the logarithmic slider and the speed edit box allow to define dynamically the speed (signed value: motion toward max for positive values, motion toward min for negative values):

 

#### "Preset speeds"

Will drive the motion from a predefined speeds list.

 

Cancel

The "Add" button allow to create a new preset speed:



The name and associated speed have to be entered, so a list is created as follow:

 

Cancel

To modify or delete a preset speed, double click on the corresponding line.

For this type of command, in "try" mode, a predefined speeds list appears:



Selecting a preset speed in the list will change immediately the speed. Defining a speed equal to 0 allow to stop the motion by selecting this speed.

The "motion assistant" generates one "Move" or "Rotate" behavior and one "generic read" behavior associated to each preset speed. A true state for each of these behaviors will apply the corresponding preset speed. If several behaviors have a true state, the first one (higher behavior in the list) will have priority. Reminder: for behaviors, equal to 0 is a false boolean state and not equal to zero a true state.



### "position" sub mode

With this sub mode ("mode / position"), the motion is defined by a position apply immediately or calculated with a speed profile: acceleration / deceleration.

Immediate position defining allows to set immediately the position of a 3d sprite with a numeric value calculated, by example, by an axis control of a PLC.

The "Move / Position" mode has two commands type and one option selectable by clicking on the "..." button:



#### "Behavior value defines position"

A numerical value will define the 3d sprite position.

For this command type, "min" and "max" positions defined in the "motion assistant" are not used to limit the motion.

#### "The value of the behavior between the two values (Begin and End) selected below defines the position between min and max"

A numerical value will define the 3d sprite position. If "Val" is the driving numerical value, the following formula give the 3d sprite position:

3d sprite position = (((Val - Begin)/(End-Begin)) x (Max-Min))+Min

"Val" is bounded between "Begin" and "End". If "Val" value is outside the bounds "Begin End", then the value is modified (for calculation only) to fit into the bounds.

The numerical value may vary between Begin and End and will define a position between "Min" and "Max". This allows to define smart bounds (0 to 1024, or -128 to 128 by example) to obtain a position between "Min" and "Max" as defined in the "motion assistant".

For this two command types (the two previously described), in "try" mode, a slider and an edit box allow to change the position:

 

A "move" or "rotate" behavior is generated :



#### "Use an acceleration / deceleration profile..."

This option, relevant to the two command types (the two previously described), sets up an "axis" management. The numerical value gives a position to reach, either directly or scaled (see before). Associated parameters for this option, define the maximum speed and times for acceleration and deceleration. These times as defined as time taken too pass from null speed to maximum speed (acceleration) or from maximum speed to null speed (deceleration).

Note: in case the amplitude of a motion does not achieve the maximum speed over acceleration and / or deceleration, the values ​​of acceleration and deceleration (instead of the time) are still maintained.

The motion starts as soon as the numerical value is changed. In "try" mode, a "Go" button allows to validate this change.

 

If this option ("Use an acceleration / deceleration profile) is used, then 4 "generic read" behaviors are automatically generated:



In order (from top to bottom), these behaviors are:

- "reqpos" : the value of this behavior give the requested position,

- "maxspeed" : the maximum speed (preset with the value defined in the "motion assistant"),

- "acctime" : acceleration time (preset with the value defined in the "motion assistant"),

 - "dectime" : deceleration time (preset with the value defined in the "motion assistant").

Note: "maxspeed", "acctime" and "dectime" values may be changed dynamically before modifying the "reqpos" value to start the motion.

## Sensors

Whatever the method chosen for the motion, sensors may be associated with the 3d sprite. The sensors can be of two types:

- digital: the sensor gives a boolean information depending on the 3d sprite position,

- numerical : the sensor gives a numerical information giving the object position.

For adding a sensor, click on the "Add a sensor" button:



The name of the sensor and the detection zone (begin and end) may be defined:



Once the sensor is created, its detection area can be viewed and modified directly from the "motion assistant". The modification is performed by grasping the ends of the detection area:

 

To modify or delete a sensor, double click on the corresponding line in the sensors list:

 

B

A

Notes:

- any number of sensors can be created, this allows for example to obtain information on the intermediate positions of a 3d sprite (other than the min and max position).

- If a sensor disappears from the visible area (A) modify its properties "Start" and / or "End" by double clicking on the corresponding line (B).

In "try" mode digital sensors states (red for true, green for false) is displayed:



Digital sensors have a "true (1)" state if the 3d sprite position is between "min" and "max" values and a "false (0)" state otherwise.

A "test joint" behavior is generated for each digital sensor:



### Numerical sensors

To define a numerical sensor select "Position":



Detection zone and scaling values, are used to calculate the value returned by the sensor.

The formula is as follow ("Val" is the value returned by the sensor):

Val=(((sprite 3d position-Begin)/(End-Begin))x(Value for begining-Value for end))+Value for begining

If the sprite 3d position is less than "Begin", it is regarded as being equal to "Begin".

If the sprite 3d position is greater than "End", it is regarded as being equal to "End".

A "get 3d sprite informations" behavior is generated for each numerical sensor:

