Tooyaa (thuja.hynes):
Good morning :) Thank you for attending today's presentation
and thank you Chantal and Tulpa for hosting this stimulating series at Science Circle.

Bruce (bruce.mowbray): yeppers
Talliver Hartnell: yep
Dawn Rhiannyr: wonderful

Tooyaa (thuja.hynes): Quaternions and Hamiltonians
It's not my intent today to do a building class, or a deep math theory exploration. What I hope to accomplish, is a review of a brilliant technique for simplifying and understanding the difficult concept of rotations in 3D.

Calculating 3-dimensional rotations is not limited to computer graphics, but is necessary also for computer vision, robotics, control theory, signal processing, attitude control, physics, bioinformatics, molecular dynamics, computer simulations, and orbital mechanics.

Tooyaa (thuja.hynes):
For example, it is common for the attitude-control systems of spacecraft to be commanded in terms of quaternions.

Our brains are masterful at very quick and precise calculations to orient our bodies properly in the vertical gravitational field, and also the other two dimensions, even without the use of quaternions.

Tooyaa (thuja.hynes):
Here, a shark shape is imposed on a 3-dimensional coordinate system to illustrate the kind of rotations a shark is able to to, and must do in order to survive. As far as I know, sharks cannot perform complex math.
Tooyaa (thuja.hynes):

Some brains are even more brilliant than average.

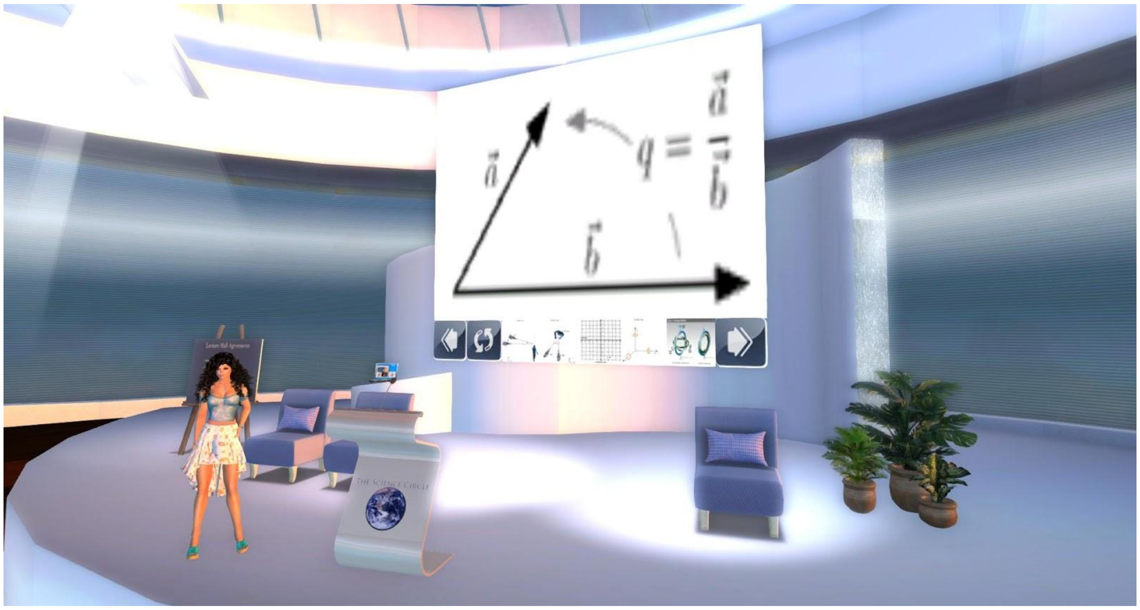
Stephen Xootfly: Biology is wonderful stuff!

Tooyaa (thuja.hynes):
Displayed is a plaque on Broome Bridge on the Royal Canal in Dublin, Ireland, commemorating William Rowan Hamilton's discovery of the quaternion in 1843.

Patio Plasma: A legendary plaque

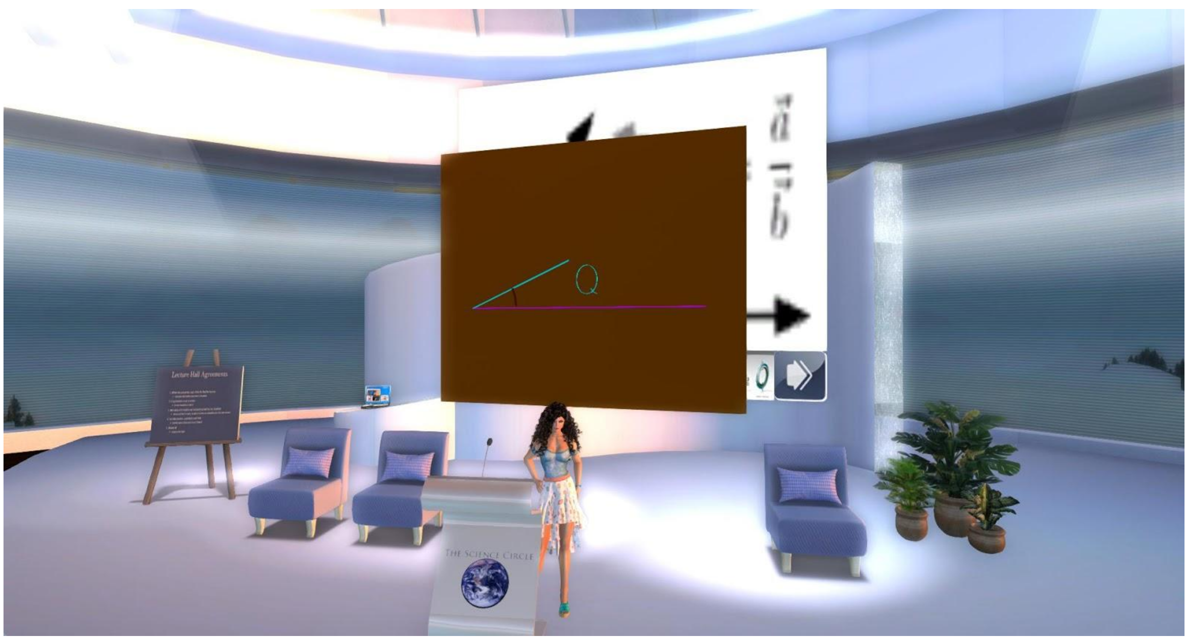
Tooyaa (thuja.hynes):
It reads:
"Here as he walked by
on the 16th of October 1843.
Sir William Rowan Hamilton
in a flash of genius discovered
the fundamental formula for
quaternion multiplication
i^2 = j^2 = k^2 = ijk = -1 "

Bruce (bruce.mowbray): ha ha.
Talliver Hartnell: :)
Stephen Xootfly: is on the edge of his seat
Chantal (nymf.hathaway): ㋡



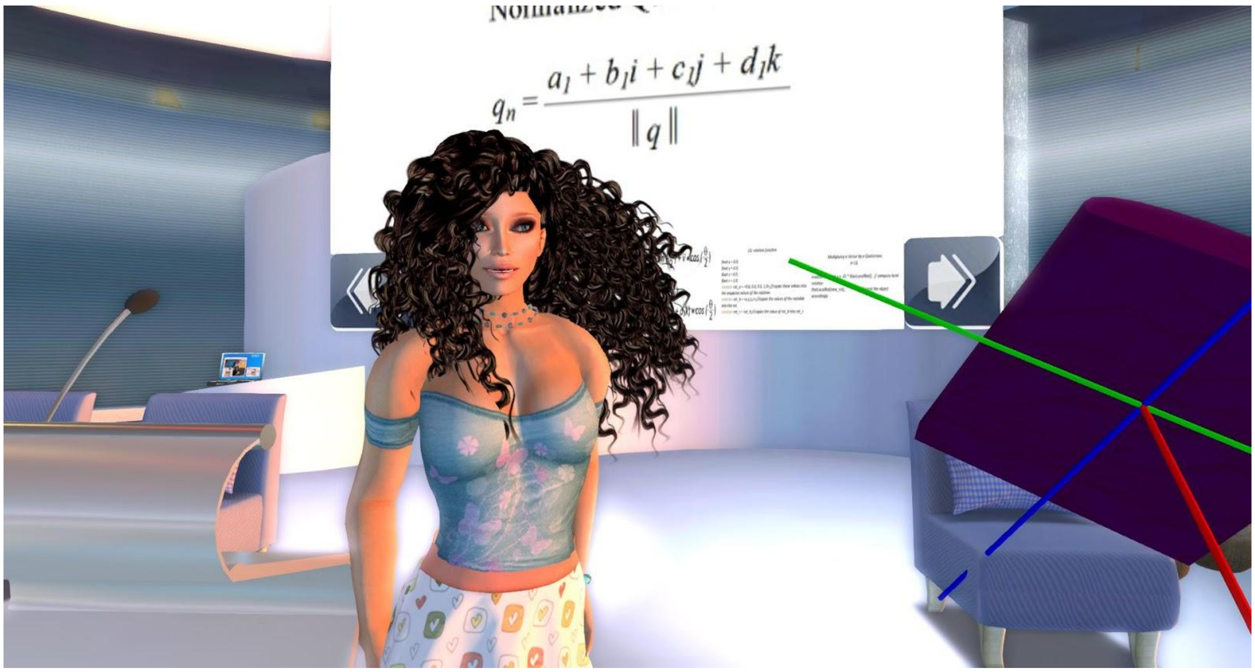
Tooyaa (thuja.hynes):
Hamilton himself defined a quaternion as the quotient of two directed lines in tridimensional space or as the quotient of two vectors.

Bruce (bruce.mowbray): I see it fine.
Patio Plasma: I can see it.
Areyn Laurasia: It's a bit dark but visible.



Tooyaa (thuja.hynes):
Another way to state this is, the quaternion acts on one vector (b) to change its magnitude and direction, to produce another vector (a).
Marina (marinamarhs): ty

Chantal (nymf.hathaway): ㋡
Justa Destiny: lol
Bruce (bruce.mowbray): ;-))
Stephen Xootfly: laughs
Patio Plasma: got it
Justa Destiny: yes
Talliver Hartnell: The idea does
Chantal (nymf.hathaway): hehehhe
Bruce (bruce.mowbray): Yes, it makes sense.
Dawn Rhiannyr: :)



Tooyaa (thuja.hynes):
The Quaternion is a polynomial containing 4 terms: a constant and three imaginary number components. It describes an orientation in 3D space.
Quaternions can be considered to be two parts: a scalar part and a vector part. The scalar part is the first term, the real term. The vector part is composed of the three imaginary components.

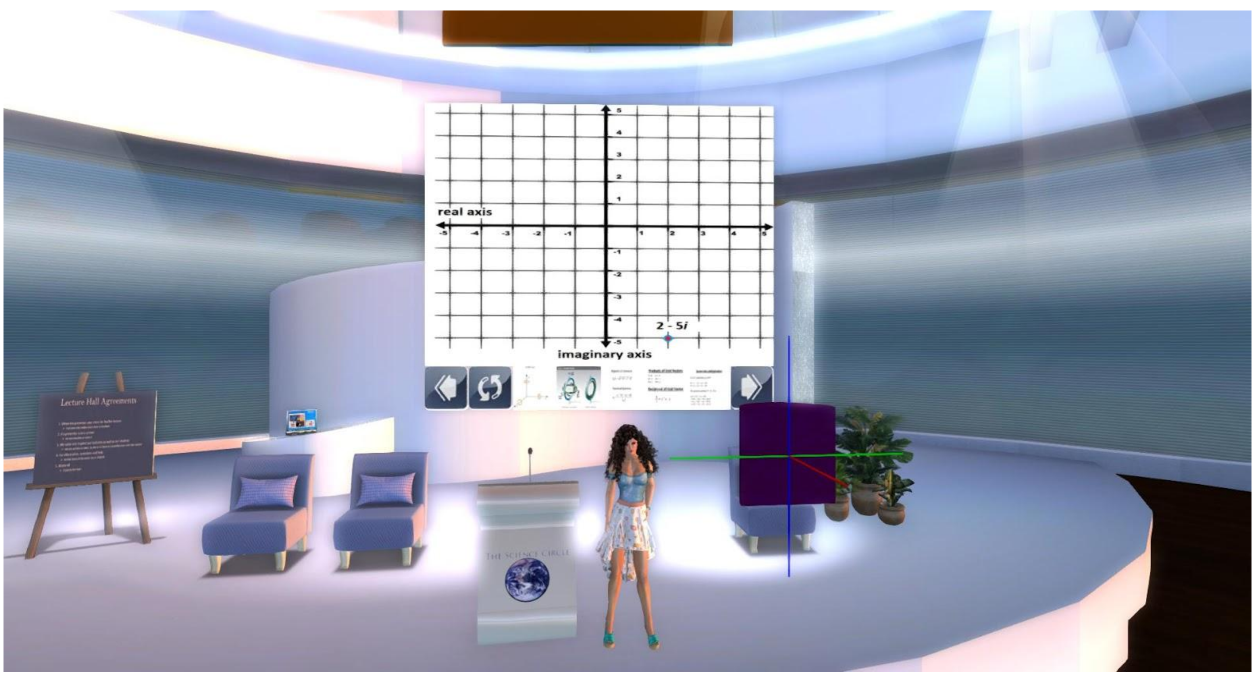
A quaternion stores an axis and the amount of rotation about the axis.

An object's orientation is always relative, either to a world-frame, or to the object's previous rotation, or to the axial orientation of any other object or system.

Tooyaa (thuja.hynes):

In Second Life, when we manipulate objects, we have a choice of reference frames: local, or global. The local reference frame of an object is the axial orientation it has when it is initially rezzed, like a 3D axes locked to the object.

Rotating an object within its own local reference frame is tricky and error-prone, but sometimes necessary.



Tooyaa (thuja.hynes):

The global reference frame for rotation is the SL axial system. This is absolute, for example the blue axis arrow always points up and down, but the center of rotation is the same as for the local frame, which is the center of the object itself.

Tooyaa (thuja.hynes):

Another kind of rotation is used also, and that is camera rotation. The rules and calculations for camera rotations are the same as for any object, even in that the center of rotation is the camera target object.

More about rotations in Second Life later in the presentation.

Now a little about imaginary and complex numbers. i, j, and k are commonly used to represent what are called imaginary numbers. The are imaginary because they square to be negative, which no real number does.

A binomial combination a + bi is called a complex number because it has a real part "a" and an imaginary part "bi".

Tooyaa (thuja.hynes):

The most curious and wonderful aspect of complex numbers is, though they may not be entirely real, they are fantastically useful for describing all manner of actual events.

A binary complex number itself can be considered a location in 2-space, if one axis is real, and the other is imaginary.

For example, the complex number 2 - 5i could be graphed as in this image.

Talliver Hartnell: yes

Tooyaa (thuja.hynes):

The most curious and wonderful aspect of complex numbers is, though they may not be entirely real, they are fantastically useful for describing all manner of actual events.

When an object rotates in space, that rotation is typically considered with respect to 3 axes: x (right/left), y (forward/back), and z (up/down).

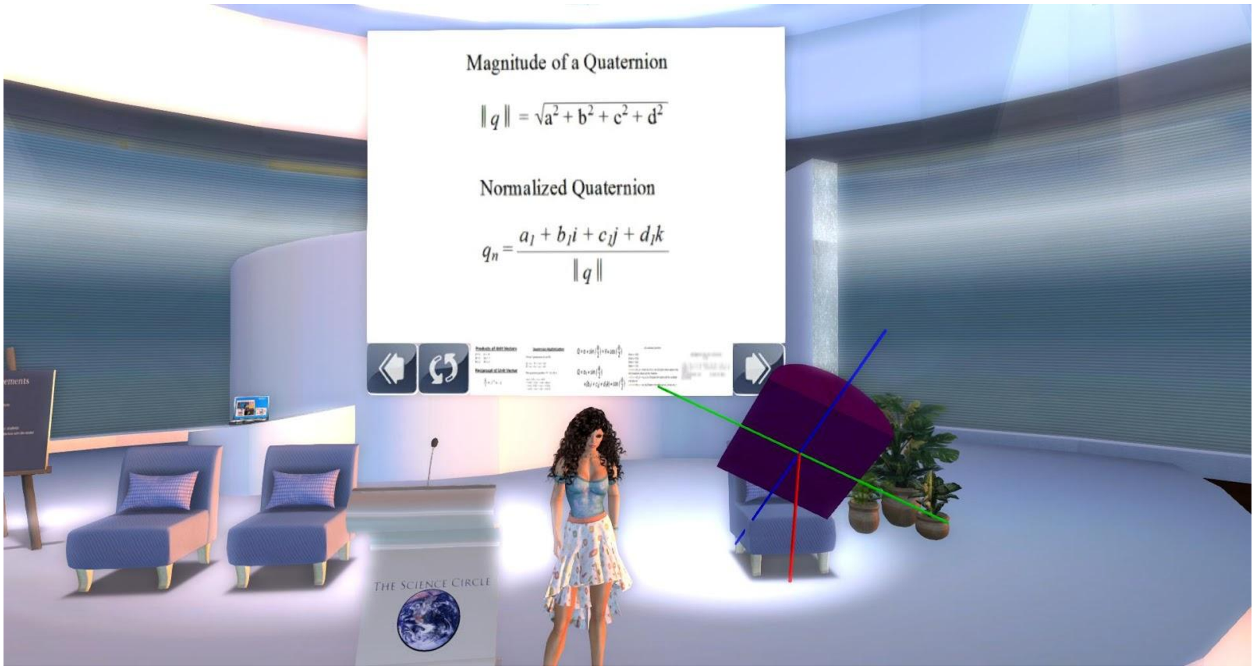
A simpler method of rotating objects is using Euler angles; basically rotating an object sequentially on each axis, usually first the z-, then the y-, then the x-axis.

Patio Plasma: And the order matters

Tooyaa (thuja.hynes):
When object rotations are done by other methods, such as Euler angles or matrices, there is a risk of something called gimbal lock.

Gimbal lock is the phenomenon of two rotational axis of an object pointing in the same direction. Simply put, it means your object won't rotate how you think it ought to rotate.

Quaternions avoid gimbal lock because they do not work on individual axes of rotation, but by the system as a whole.



Tooyaa (thuja.hynes):

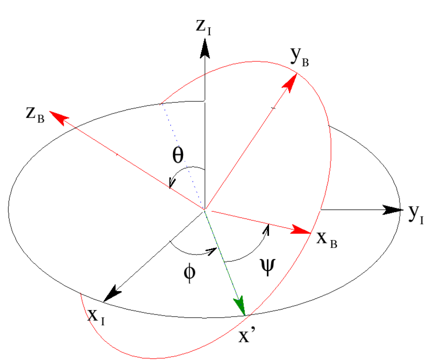
An important step before using quaternions to calculate rotations, is to normalize them. This means essentially to make the axial reference vector(s) 1 unit long. If this is not done, repeated calculations may be lead to significant errors.

As shown, the norm of a quaternion, or any vector, is the square root of the sum of the squares of the coefficients.

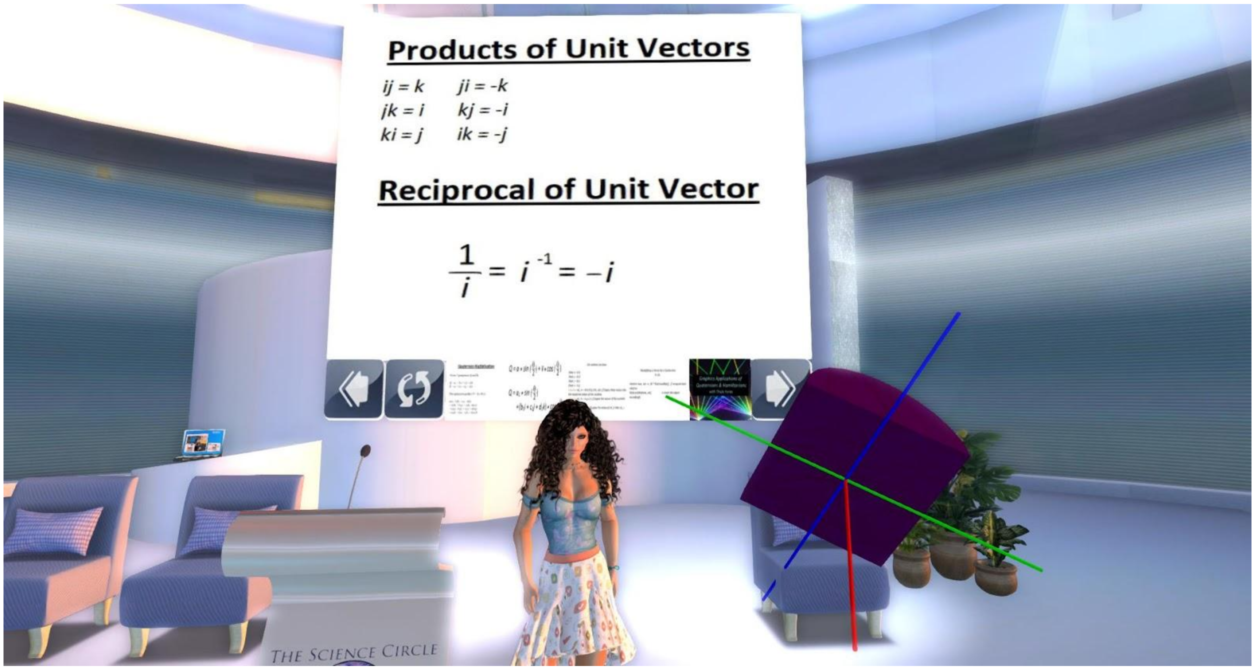
Conjugation by a unit quaternion (a quaternion of absolute value 1) with real part cos(?) is a rotation by an angle 2?, the axis of the rotation being the direction of the imaginary part. The advantages of quaternions are:

Nonsingular representation (compared with Euler angles for example).

Complex number multiplication is associative and distributive but not commutative. That is, the order of multiplication matters.



(Euler Angles**)**



Tooyaa (thuja.hynes):
Here we see a few of the rules which define right-to-left multiplication of unit vectors when conceived as imaginary numbers.

Here you see the result of multiplying two quaternions, which is a new quaternion.

Stephen Xootfly: Ouch. brain pain.
Max Chatnoir: :-)
Stephen Xootfly: smiles
Bruce (bruce.mowbray): :)
Dawn Rhiannyr: Indeed :))
herman Bergson: Had the same problem lately :-)
Talliver Hartnell: A new orientation
vȋʀţȕą̊ɫʝőɫɫẏ (virtualjolly): ◕‿◕
Justa Destiny: A simple rotation rotates half way between Q and R ?
Tooyaa (thuja.hynes): slerp

**(**Stephen Xootfly’s brain**)**

Tooyaa (thuja.hynes):

Another, perhaps more straightforward, method of acting on a vector is multiplication by a quaternion in trigonometric form.

To represent a rotation of a given angle around a given axis, first, convert the axis to a unit vector (a vector of magnitude 1).

Take the x, y, and z components of the unit vector and multiply them by the sine of half the angle (these are the x, y, and z elements of the quaternion). The s element of the quaternion is equal to the cosine of half the angle. This formula is shown on t.

Finally, a note about our own beloved Second Life. Among the many attractive features of this complex virtual world, is the ability to script object rotations. LSL uses quaternions for this, in its function "rotation".

Patio Plasma: You were just seeing if we were awake.
Chantal (nymf.hathaway): :)

Tooyaa (thuja.hynes):

A rotation is a variable type comprising 4 floats used together as a single item. This data is interpreted as a quaternion. As with vectors, each component can be accessed via '.x', '.y', '.z', and '.s'.

Max Chatnoir: Why isn't it z, y, x if the order matters, or does the program fix the order correctly?

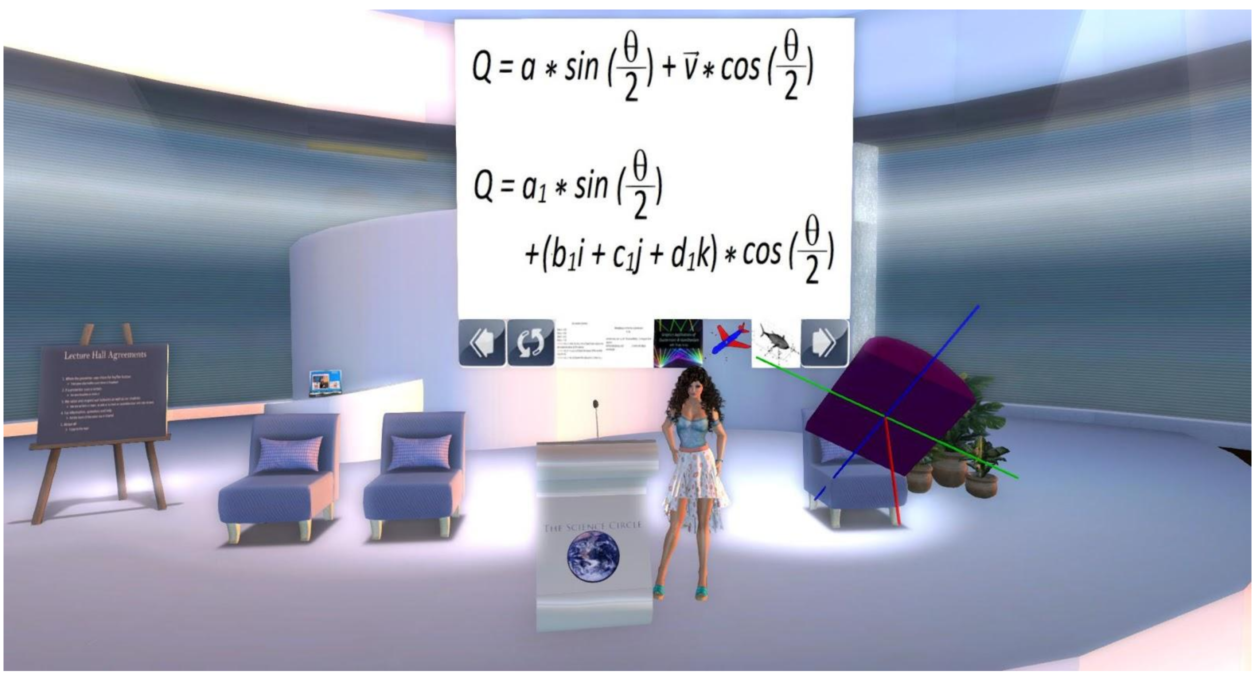
Tooyaa (thuja.hynes):

Euler angle rotations are made in reference to the z-axis and the line of nodes N which is the intersection of the beginning and the ending xy-planes. Really it's not rotation around z, y, and x, but rather first rotation around the original z-axis, next rotation around the line of Nodes, which is orthogonal, and final rotation around the new Z-axis, which is orthogonal. There are six possible ways of doing this, and any of the six can be programmed. The other question pertained to the mathematics of 'gimbal lock.' Briefly, if the second rotation around the N axis is 0- or 180- degrees, then a degree of freedom is lost, and the result is the final rotation will be around the original z-axis, Mathematically, this is equivalent to a singularity, a single rotation, rather than 3 rotations.

Max Chatnoir: Ah, OK. Thanks.

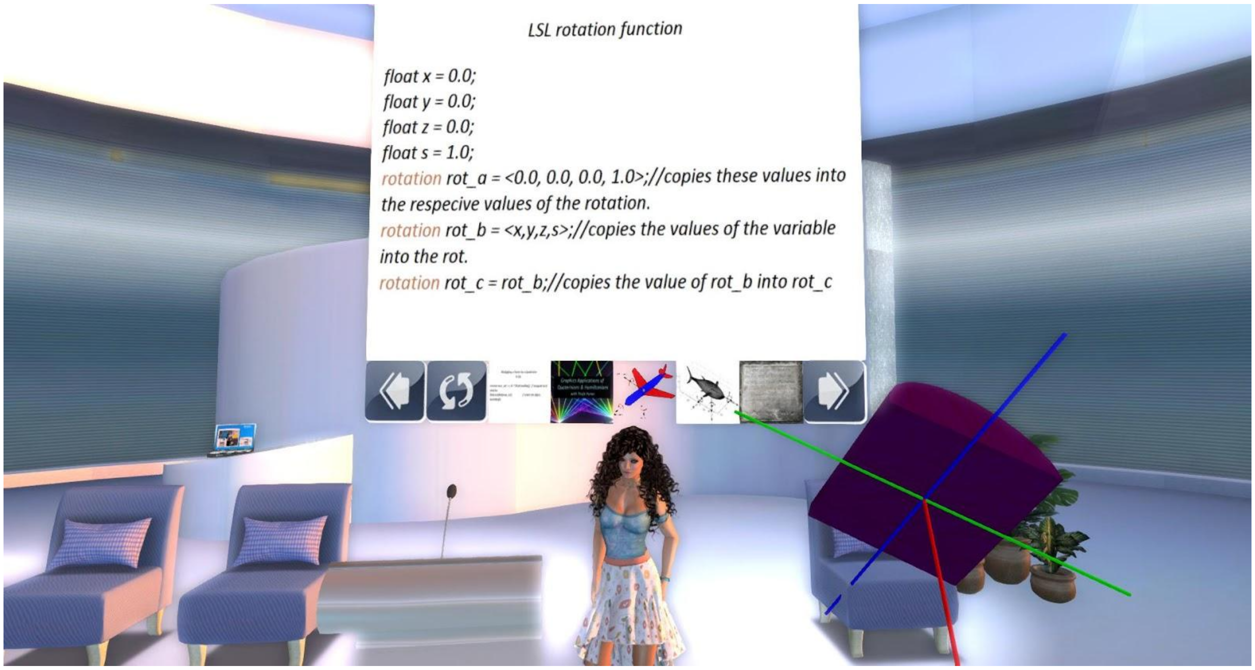
Tooyaa (thuja.hynes):

In Second Life, you can use rotations without dealing with the individual elements of quaternions. LSL offers library calls that convert between a quaternion and a vector containing the equivalent Euler representation: llEuler2Rot and llRot2Euler.



Tooyaa (thuja.hynes):

An object can be rotated around an arbitrary point by multiplying a vector by a rotation in the manner described above. The vector should be the difference between the object's current position and the desired "center-point" of rotation. Take the result o.



Tooyaa (thuja.hynes):

This rotates the object around its local x-axis, which depends on its current orientation. Think of this as specifying the rotation from the object's point of view, that is, relative to the direction it is currently facing.

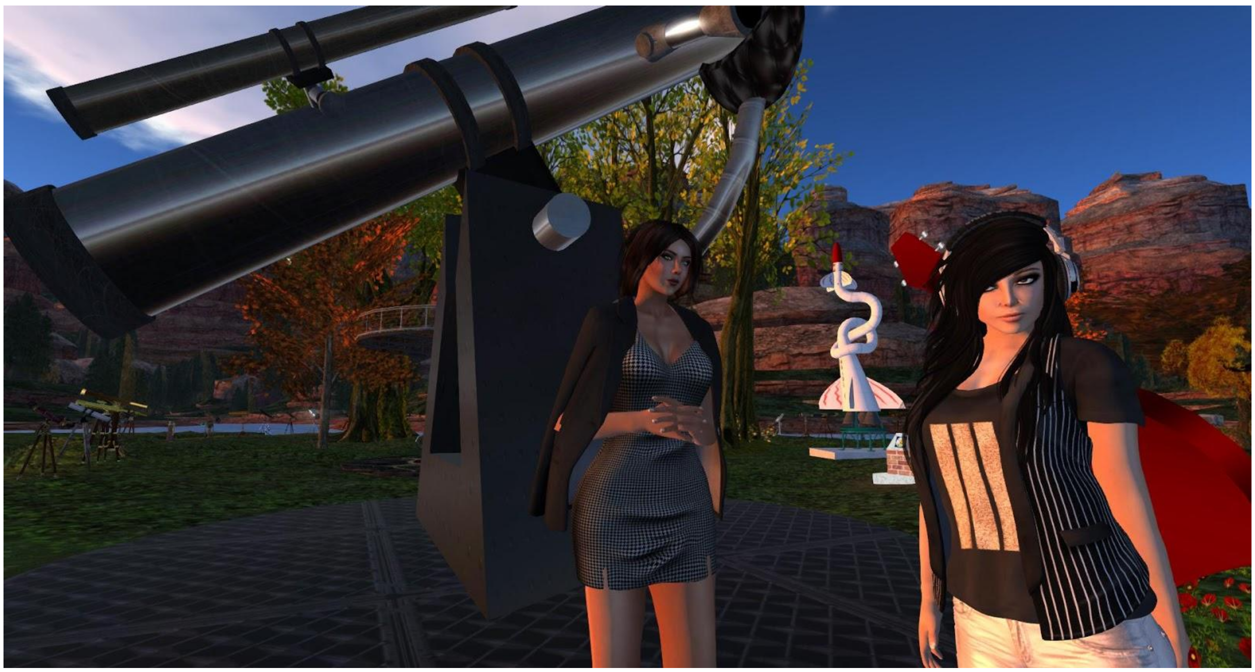
At this time I will try to answer questions you may have.
Thank you all for coming today.
I hope this math was interesting and will be useful to you sometime.

***Patio Plasma:** Excellent presentation*

Max Chatnoir: Do!
Stephen Xootfly: Wonderful. :)
Max Chatnoir: A revelation. :-)
Chantal (nymf.hathaway): Thank you Tooyaa, wonderful presentation!
Ţulpa (jes.cobalt): Thank you Tooyaa ㋡
Talliver Hartnell: ty
vĩʀʈǖą̥̈́łʝőłłẏ (virtualjolly): brava ◕‿◕
Paolo Rousselot: Thanks!
Stephen Xootfly: Thank you.
Chantal (nymf.hathaway): ㋡
herman Bergson: My brain is rotated....
Dawn Rhiannyr: Thank you Tooyaa
Areyn Laurasia: Thank you :)
Patio Plasma: I heard that [James Maxwell]() wrote his electromagnetic equations in quaternion form
Justa Destiny: Yes you mentioned a formula to smoothly move from one position to another at 30 fps, can we get the formula again please ^^
Paolo Rousselot: Suffered brain snap only a few times...
Ţulpa (jes.cobalt): ㋡ Paolo
Dawn Rhiannyr: :)
vĩʀʈǖą̥̈́łʝőłłẏ (virtualjolly): Complimenti bella sim e belli avi ◕‿◕ molto serio mi piace
virtualjolly Resident: compliments the beautiful sim and beautiful avi ◕‿◕ very serious like

Max Chatnoir: Can you come and teach math at my school? That was really nice.
Chantal (nymf.hathaway): Always nice on a Sunday morning and deserved Tooyaa, great presentation. Makes a great PDF.
Tooyaa (thuja.hynes): A process of iterating little rotations to give a convincing sense of smooth rotation from start to finish.
Justa Destiny: How would the math be like? To move from A to B for example
Tooyaa (thuja.hynes): slerp
Justa Destiny: slerp
herman Bergson: http://en.wikipedia.org/wiki/Slerp
Paolo Rousselot: Sorry to have to leave, this has been wonderful!
Ʈulpa (jes.cobalt): waves Paolo ㋡
Chantal (nymf.hathaway): Waves at Paolo ㋡
Justa Destiny: Yes
Justa Destiny: I would do a do while ?
Areyn Laurasia: Would be handy for a machinima panning, control the camera?
Justa Destiny: Any LSL library call could help me?
Justa Destiny: ok
Tooyaa (thuja.hynes): llEuler2Rot and llRot2Euler
Justa Destiny: tyvm
Chantal (nymf.hathaway): Tooyaa, Jes and I will create a PDF and spread the word. I need to log off now to go to OS for the testrun on the field trip next week. Thank you for giving us this great presentation!
Chantal (nymf.hathaway): JP ㋡
Ʈulpa (jes.cobalt): Hi JP ㋡

Chantal (nymf.hathaway): You are too late...you missed Tooyaa's wonderful presentation.
Dawn Rhiannyr: Hi JP :)
Talliver Hartnell: Is gimbal lock a physical process or math or both?
Chantal (nymf.hathaway): Euler all over it ㋡
JeanPierre Euler: Hello!
Chantal (nymf.hathaway): :)))
Justa Destiny: Yes
Tooyaa (thuja.hynes): tensor math
Chantal (nymf.hathaway): This lady needs to go OS ㋡ Thanks everyone ㋡
Ţulpa (jes.cobalt): Does as well :(
Dawn Rhiannyr: Says thank you and waves.



Chantal & Jes Sept 2014