Prog Notes – openHTM

Contents

[1. Picking 2](#_Toc426305543)

[a. Mouse Locations 2](#_Toc426305544)

[b. 10/07/2015 – Debugging Picking of Distal 14](#_Toc426305545)

[c. Picking Cubes 17](#_Toc426305546)

[d. Cubes in Coordinate System 17](#_Toc426305547)

[3. Initial Camera placement 20](#_Toc426305548)

[a. Change Z direction 21](#_Toc426305549)

[4. Class to Dataset 22](#_Toc426305550)

[a. Class to DataTable 22](#_Toc426305551)

[1. Data to Display: 23](#_Toc426305552)

[a. Cell 23](#_Toc426305553)

[b. Distal Segment : Segment 24](#_Toc426305554)

[c. Proximal Segment : Segment 25](#_Toc426305555)

[2. DataTable 25](#_Toc426305556)

# Picking

## Mouse Locations

Window events (Click etc.) use MouseEventArgs e (getPickingRay() also)

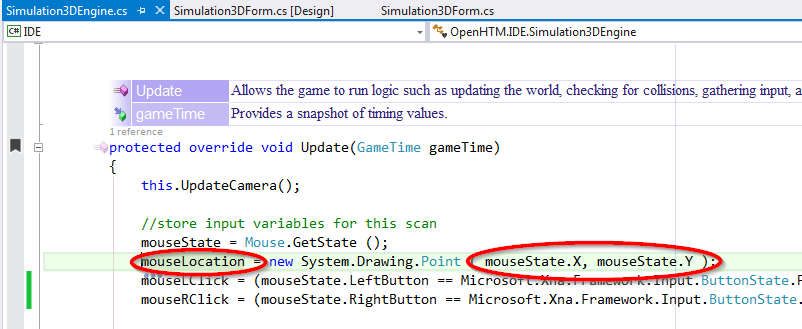
XNA Update() method uses Mouse.GetState() function

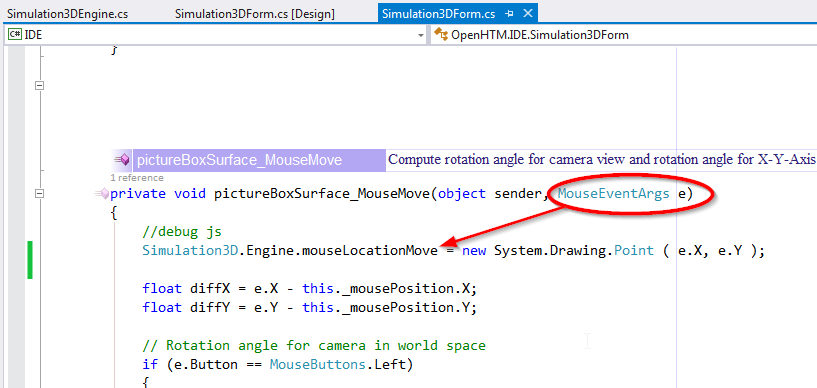
Problem: locations returned are different..

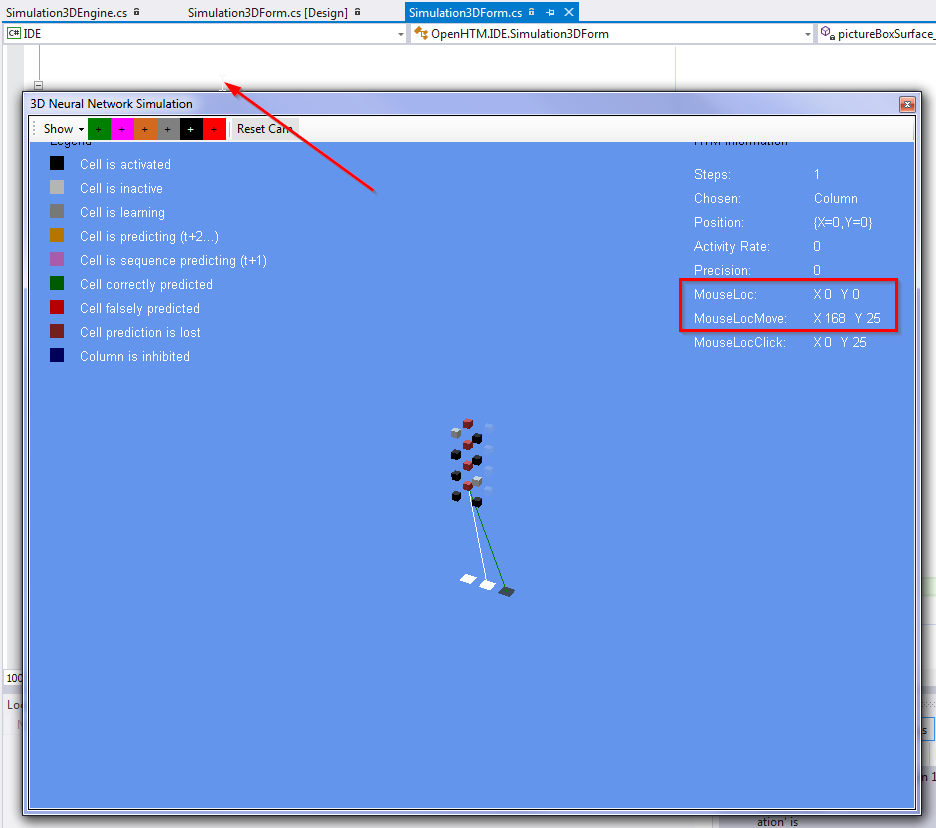
mouseLoc = MouseState.X, MouseState.Y

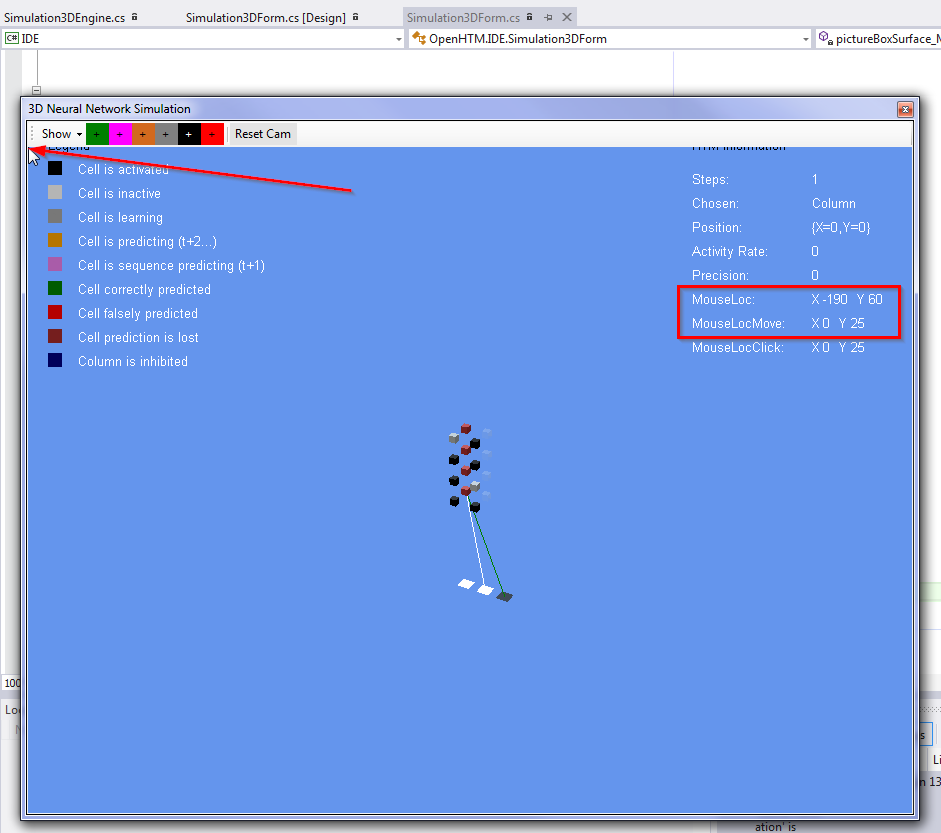
mouseLocMove = MouseEventArgs e

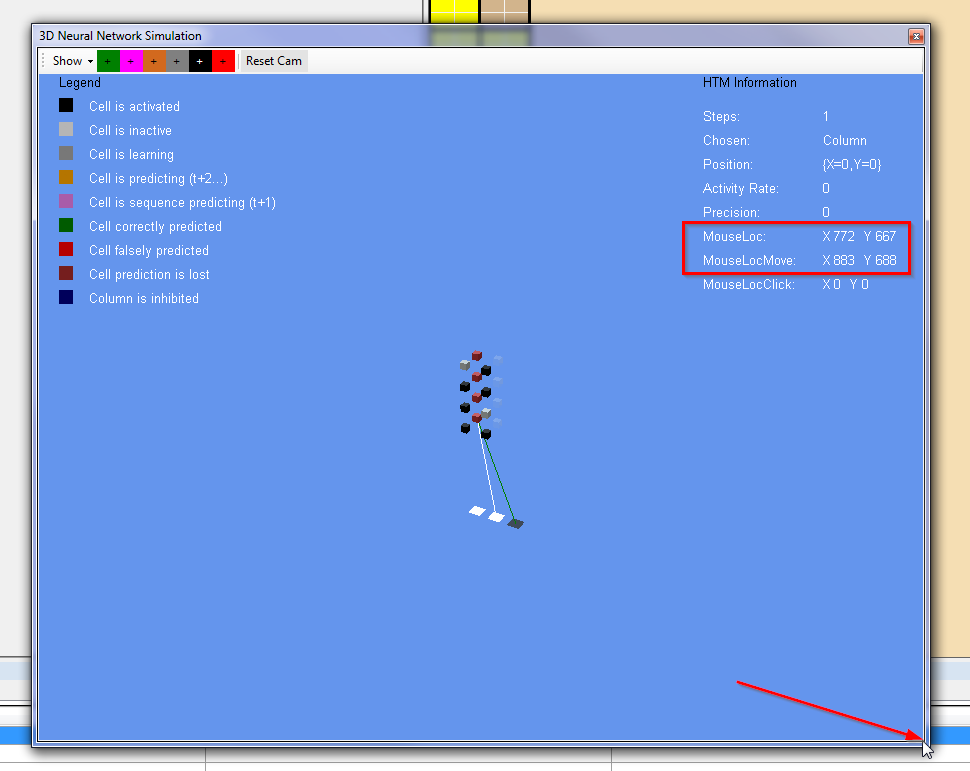
mouseLocClick = MouseEventArgs e











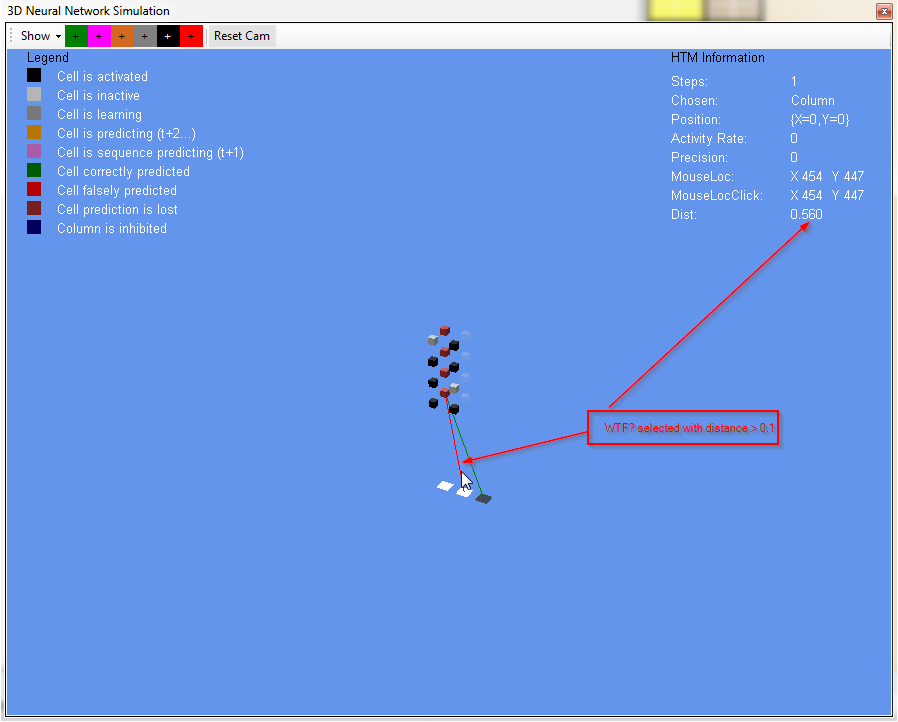
\*\*\* Do not use Mouse.State for ray picking.

2015 06 28

Still something wrong with picking.

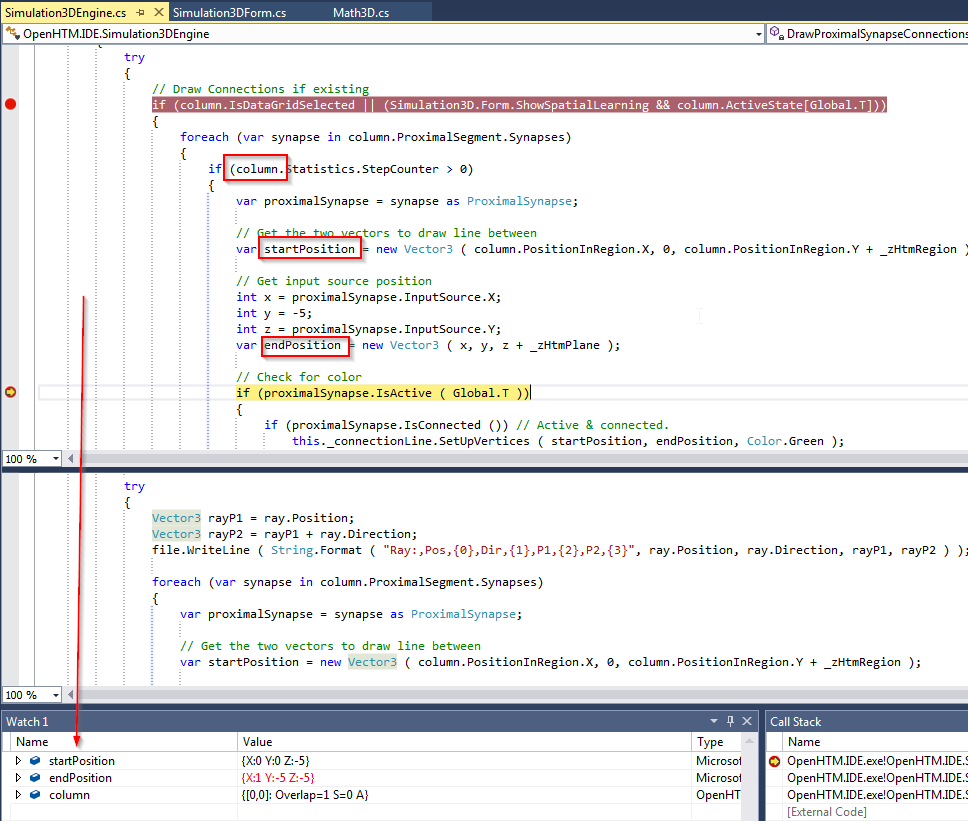
Things to check/verify:

1. R-Click mouse and drag HTMObjects – picking lost.
2. Line selected with distance > 0.1???



Test:

IN DrawProximalSymapseConnections(…) check line endpoints of synapses:



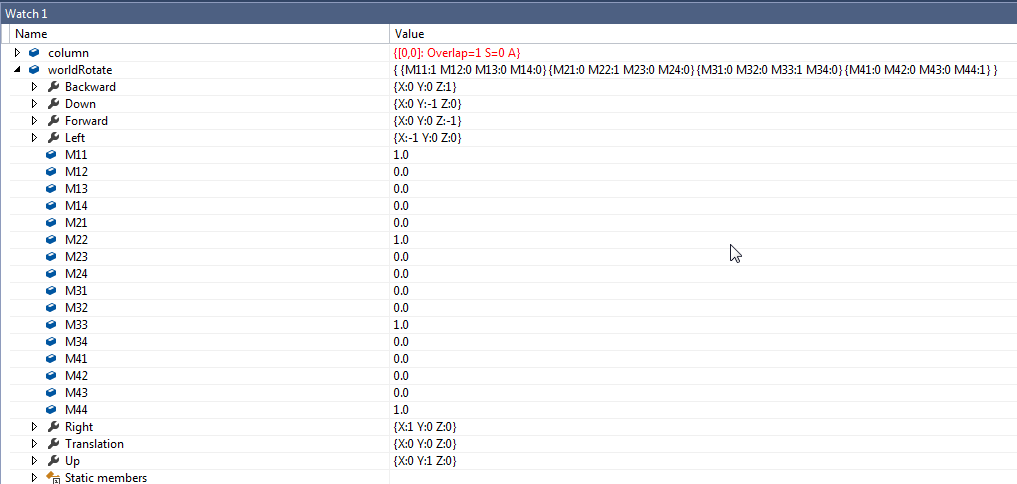
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DrawProximalSymapseConnections | | | PickProximalSymapseConnections | | |
| Column | StartPoint | EndPoint |  | StartPoint | EndPoint |
| 0,0 | 0,0,-5 | 2,-5,-5 | 0,0 | 0,-5,-5 | 2,-5,-5 |
|  | 0,0,-5 | 1,-5,-5 |  | 0,0,-5 | 1,-5,-5 |
| 0,1 | 0,0,-5 | 1,-5,-5 | 0,1 | 0,0,-4 | 2,-5,-5 |
| 1,0 | 0,0,-5 | 1,-5,-5 |  | 0,0,-4 | 0,-5,-5 |
| 1,1 | 0,0,-5 | 1,-5,-5 | 1,0 | 1,0,-5 | 0,-5,-5 |
|  |  |  |  | 1,0,-5 | 1,-5,-5 |
|  |  |  | 1,1 | 1,0,-4 | 0,-5,-5 |
|  |  |  |  | 1,0,-4 | 2,-5,-5 |

Problem – should be the same line endpoints when Draw and Pick methods called.

worldRotate matrix:

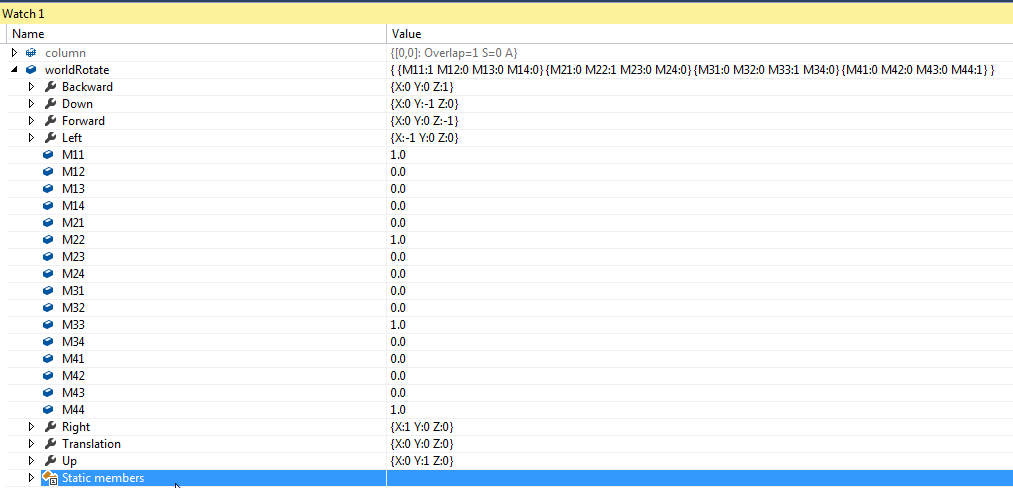
Draw:

{ {M11:1 M12:0 M13:0 M14:0} {M21:0 M22:1 M23:0 M24:0} {M31:0 M32:0 M33:1 M34:0} {M41:0 M42:0 M43:0 M44:1} }



Pick:

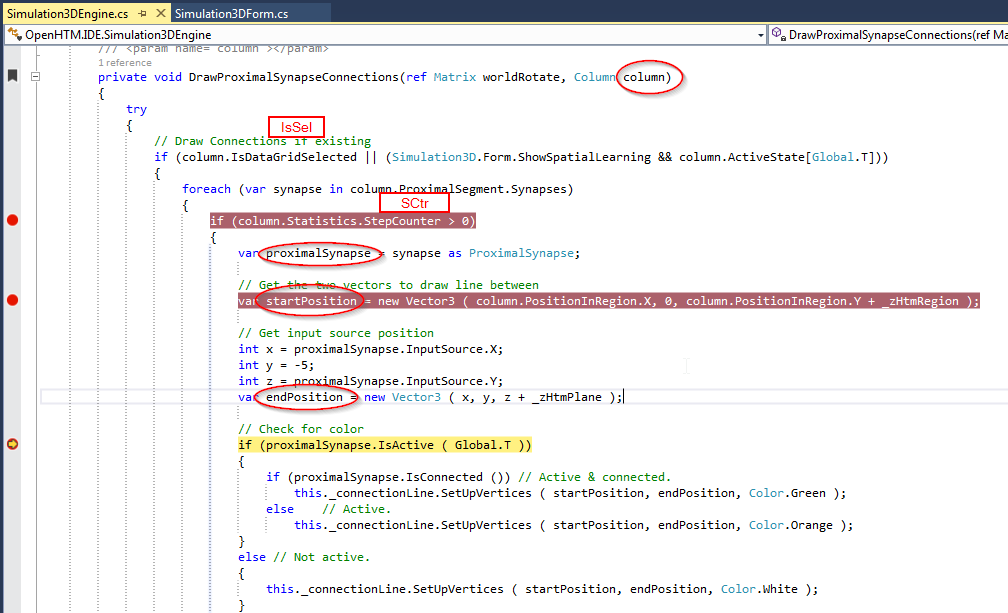
{ {M11:1 M12:0 M13:0 M14:0} {M21:0 M22:1 M23:0 M24:0} {M31:0 M32:0 M33:1 M34:0} {M41:0 M42:0 M43:0 M44:1} }

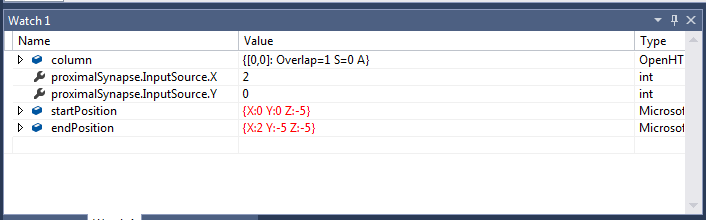


????? Check how displaying Active/Inactive Synapse Connections affects synapse lists… - Doesn’t look like it

????? Compare Column and synapse.XY values in Draw and Pick:

Watch these values:





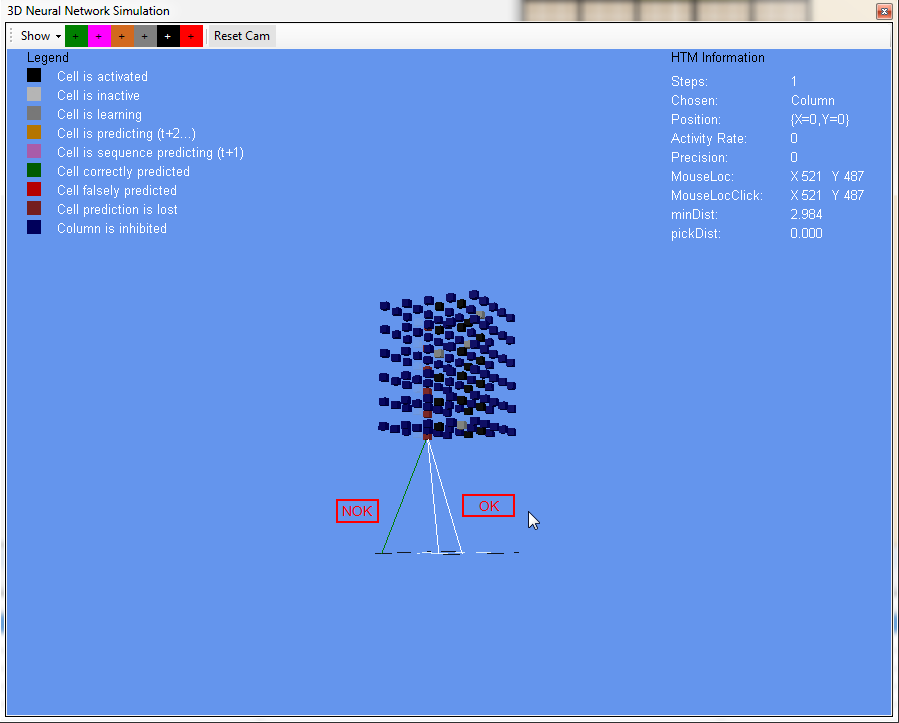
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DrawProximalSymapseConnections | | | | | | | | PickProximalSymapseConnections | | | | | | |
| Column | | | Synapse | | | Position | | Column | | | Synapse | | Position | |
| Loc | IsSel | SCtr | | X | Y | Start | End | Loc | IsSel | SCtr | X | Y | Start | End |
| 0,0 | T | 1.0 | | 2 | 0 | 0 0 -5 | 2 -5 -5 | 0,0 | T | 1.0 | 2 | 0 | 0 0 -5 | 2 -5 -5 |
|  |  |  | | 1 | 0 | 0 0 -5 | 1 -5 -5 |  |  |  | 1 | 0 | 0 0 -5 | 1 -5 -5 |
| 0,1 | F |  | |  |  |  |  | 0,1 | F |  |  |  |  |  |
| 1,0 | F |  | |  |  |  |  | 1,0 | F |  |  |  |  |  |
| 1,1 | F |  | |  |  |  |  | 1,1 | F |  |  |  |  |  |
|  |  |  | |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | |  |  |  |  |  |  |  |  |  |  |  |

Looks like **column.IsDataGridSelected** controls connection display and pick.

column.Statistics.StepCounter – is incremented when Cell.NextStep is executed.

**Test picking on project OIOO\_5x5:**

Problem: Green line doesn’t show mouse over (blink)…



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DrawProximalSymapseConnections | | | | | | | | PickProximalSymapseConnections | | | | | | |
| Column | | | Synapse | | | Position | | Column | | | Synapse | | Position | |
| Loc | IsSel | SCtr | | X | Y | Start | End | Loc | IsSel | SCtr | X | Y | Start | End |
| 0 0 | T | 1.0 | | 3 | 3 | 0 0 -5 | 3 -5 -2 | 0 0 | T | 1.0 | 3 | 3 | 0 0 -5 | 3 -5 -2 |
|  |  |  | | 0 | 4 | 0 0 -5 | 0 -5 -1 |  |  |  | 0 | 4 | 0 0 -5 | 0 -5 -1 |
|  |  |  | | 2 | 3 | 0 0 -5 | 2 -5 -2 |  |  |  | 2 | 3 | 0 0 -5 | 2 -5 -2 |
| 0 1 | F |  | |  |  |  |  | 0 1 | F |  |  |  |  |  |
| 0 2 | F |  | |  |  |  |  | 0 2 | F |  |  |  |  |  |
| 0 3 | F |  | |  |  |  |  |  |  |  |  |  |  |  |
| 0 4 | F |  | |  |  |  |  |  |  |  |  |  |  |  |
| 1 0 | F |  | |  |  |  |  |  |  |  |  |  |  |  |

Mouse on 1st connection from right: mouse XY = 440 448

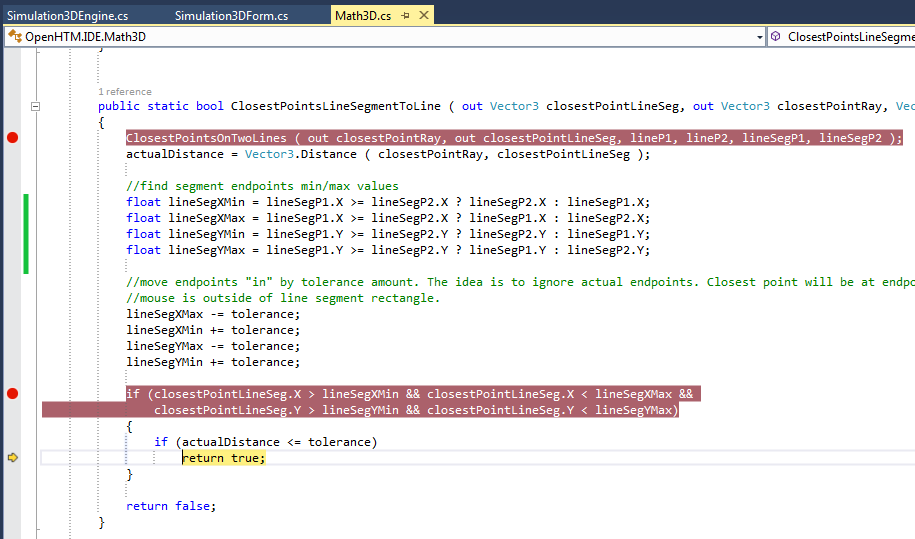
2nd connection from right: mouse XY = 428 444

3rd connection from right: mouse XY = 398 450 (this connection does not show mouseover)

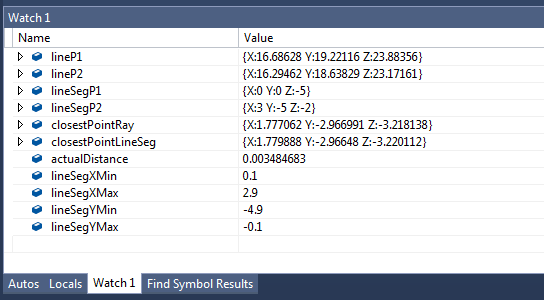
Test Pick with these values:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PickProximalSymapseConnections | | | | | | | | | | |
| Ray | | Column | | | Synapse | | Position | |  |  | |
| Mouse X | Mouse Y | Loc | IsSel | SCtr | X | Y | Start | End | intersect | actDist | |
| 440 | 448 | 0 0 | T | 1.0 | 3 | 3 | 0 0 -5 | 3 -5 -2 | T | 0.00348 | |
|  |  |  |  |  | 0 | 4 | 0 0 -5 | 0 -5 -1 | F | 1.816 | |
|  |  |  |  |  | 2 | 3 | 0 0 -5 | 2 -5 -2 | F | 0.507 | |
|  |  |  |  |  |  |  |  |  |  |  | |
| 398 | 450 |  |  |  | 3 | 3 |  |  | F | 1.893 | |
|  |  |  |  |  | 0 | 4 | 0 0 -5 | 0 -5 -1 | F | 0.0108 | |
|  |  |  |  |  |  |  |  |  |  |  | |
|  |  |  |  |  |  |  |  |  |  |  | |

Inside ClosestPointsLineSegmentToLine(…)



mouse XY = 440 448



if (closestPointLineSeg.X > lineSegXMin && 1.779 > 0.1

closestPointLineSeg.X < lineSegXMax && < 2.9

closestPointLineSeg.Y > lineSegYMin && -2.966 > -4.9

closestPointLineSeg.Y < lineSegYMax) < -0.1

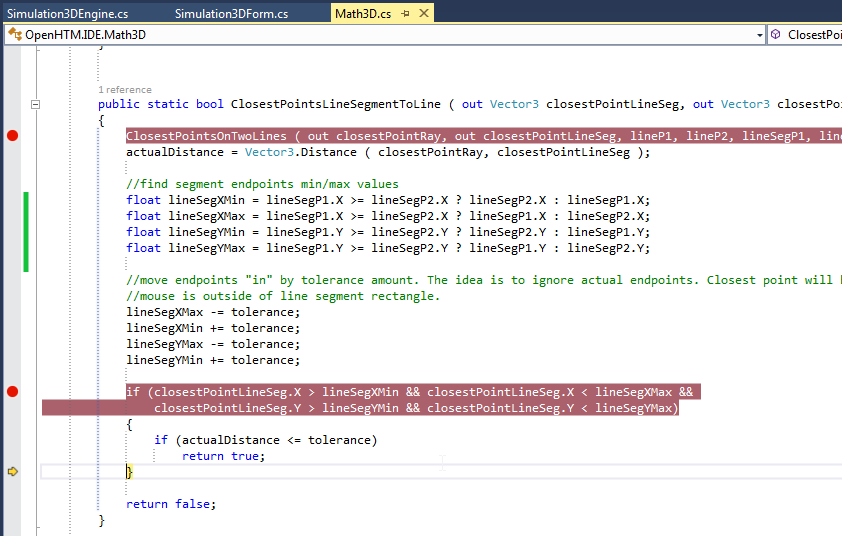
{

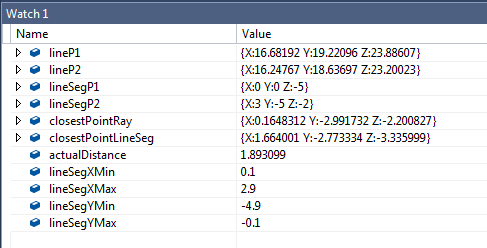
if (actualDistance <= tolerance)

return true;

}

mouse XY = 398 450





if (closestPointLineSeg.X > lineSegXMin && 1.246 > 0.1

closestPointLineSeg.X < lineSegXMax && < 1.9

closestPointLineSeg.Y > lineSegYMin && -3.116 > -4.9

closestPointLineSeg.Y < lineSegYMax) < -0.1

{

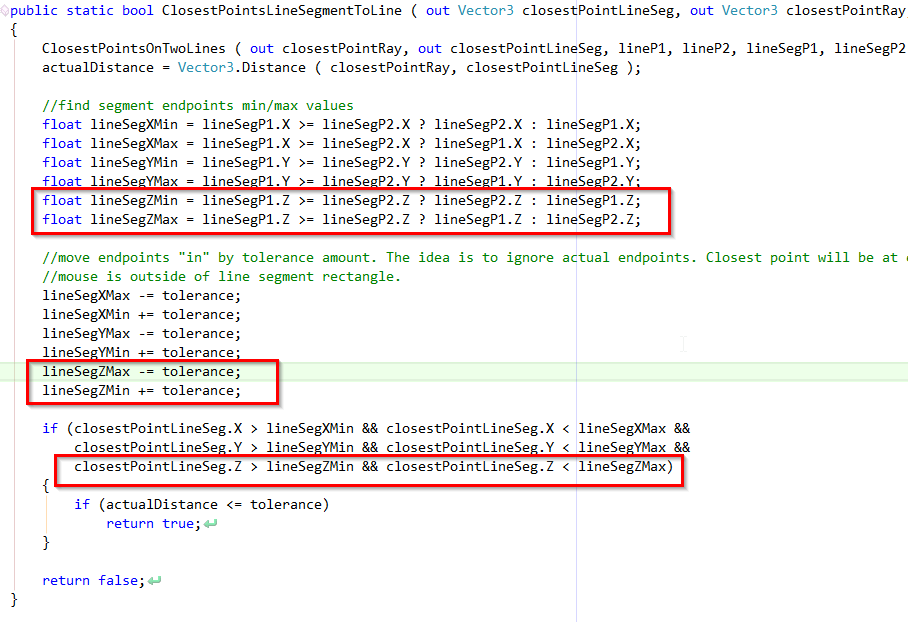
if (actualDistance <= tolerance)

return true;

}

2 Jun 2015

Added Z to ClosestPoints function:



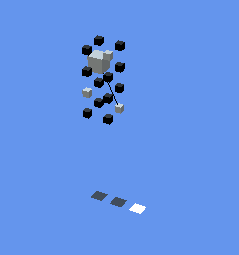
Test Mouse Locations:

440, 448

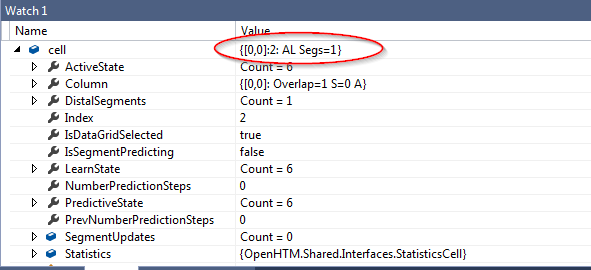
428, 444

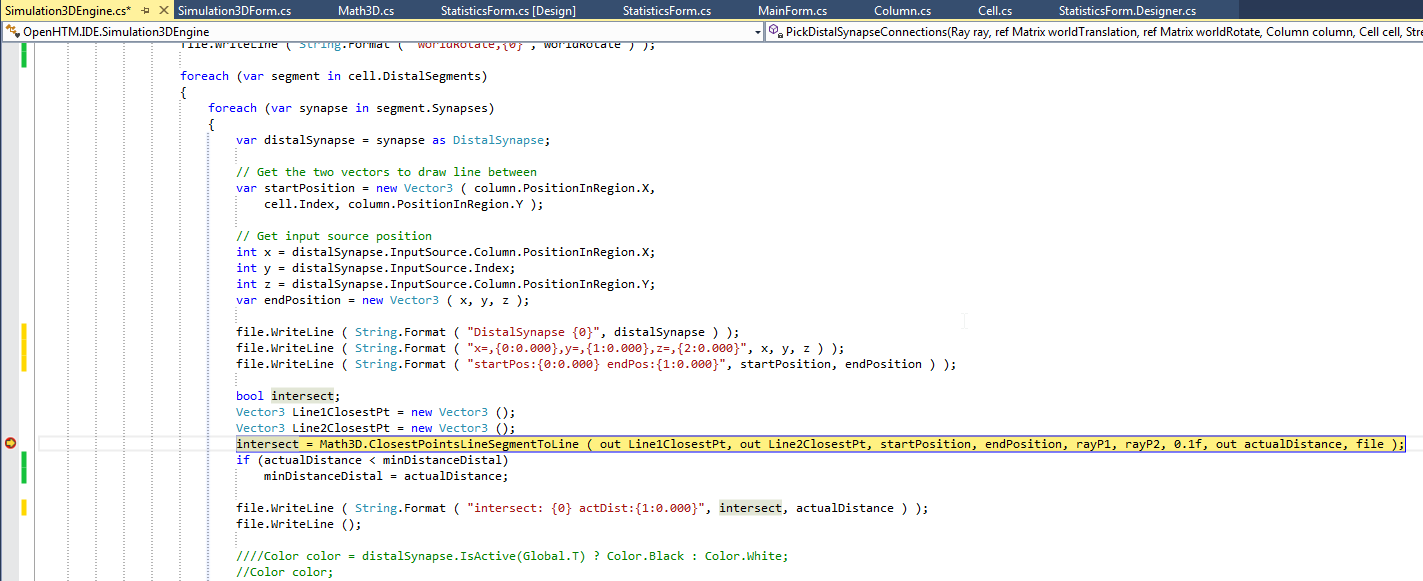
398, 450

## 10/07/2015 – Debugging Picking of Distal



This cell is Col 0,0 cell 2:





x = 1

y = 0

z = 0

startPosition = {X:0 Y:2 Z:0}

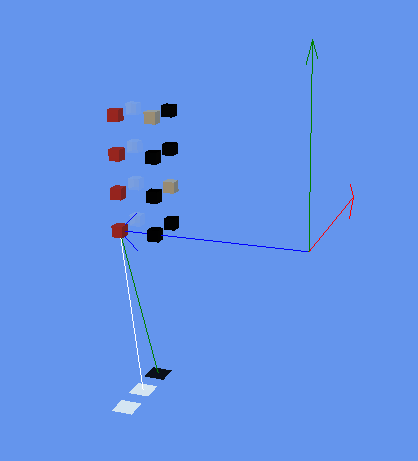
endPosition = {X:1 Y:0 Z:0}

|  |  |
| --- | --- |
|  |  |

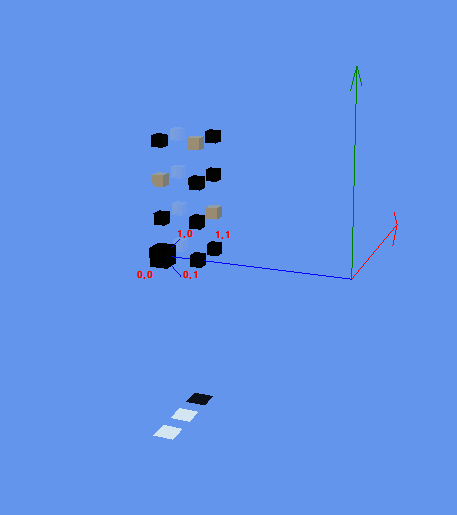
## Picking Cubes

## Cubes in Coordinate System

Why are cells drawn not from 0,0,0 but like this:



Column numbering:



In DrawHtmlRegion(…):

…

var translationVector = new Vector3(column.PositionInRegion.X, cell.Index, column.PositionInRegion.Y);

…

worldTranslation = Matrix.CreateTranslation(translationVector) \* worldTranslationZ;

Matrix world = worldScale \* worldTranslation \* worldRotate;

// Draw cube

this.\_cube.Draw(world, this.\_viewMatrix, this.\_projectionMatrix, color, alphaValue);

>>> worldTranslationZ causes cells to be shifted by 0,0,-5 (by \_zHtmRegion)

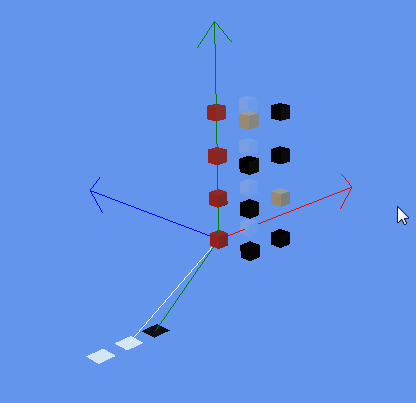
Changed:

<<<

// Coordinates constants

private const float \_zHtmRegion = -0.0f; //was-5.0f

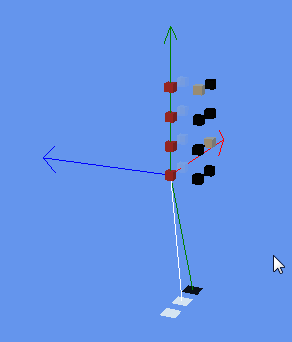
Result: Input plane still shifted 0,0,-5



<<<

private const float \_zHtmPlane = -0f; // Shift to the side //was -5.0f

:::



# Initial Camera placement

In ResetCamera()

Added:

GetSize() gets the size of the region. Camera is placed Z\*2 + 25 from origin. This places most of the region in camera view.

protected internal void ResetCamera()

{

//position camera relative to region size

//Y=4 slightly raised,

//pitch =-10 look slightly down

//Z=Size/3 + 3 - shift to right to give angled view (+ 3 provides shift for small regions)

this.\_posCamera = new Vector3 ( -25, 4, GetSize().X/3 + 3 );

// Reset rotation angle for camera

this.\_yawCamera = (float)MathHelper.ToRadians(-90);

this.\_pitchCamera = (float)MathHelper.ToRadians ( -10 );

// Reset rotation angle for htm-objects

this.\_yawHtm = 0f;

this.\_pitchHtm = 0f;

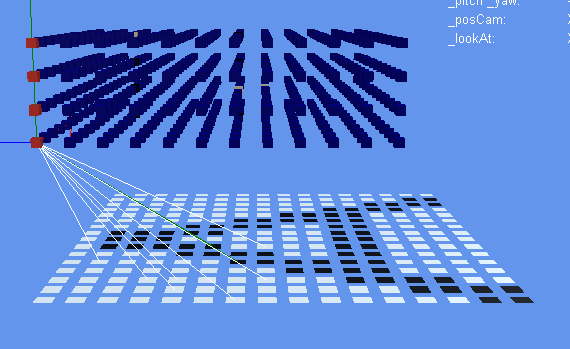
// Reset zoom

this.\_zoomCamera = 35f;

this.UpdateCamera();

}

>>>



## Change Z direction

Introduced \_zDir = -1f;

Change Cells direction:

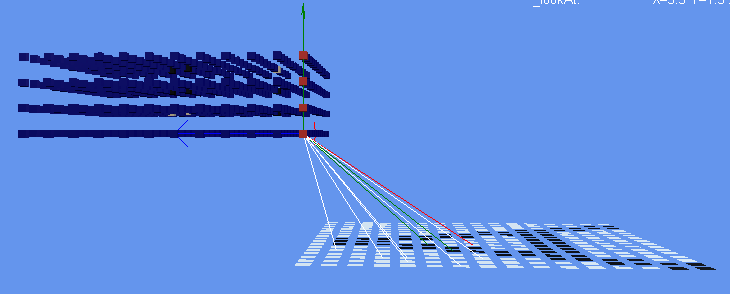
DrawHtmlRegion():

//calculate cell world coordinates

var translationVector = new Vector3(column.PositionInRegion.X, cell.Index, column.PositionInRegion.Y \* \_zDir);

PickHtmlRegion():

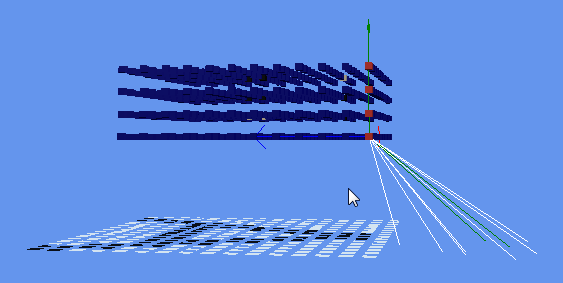
var translationVector = new Vector3 ( column.PositionInRegion.X, cell.Index, column.PositionInRegion.Y \* \_zDir );



Change HtmPlane direction:

In DrawHtmPlane():

Matrix worldTranslation = Matrix.CreateTranslation(new Vector3(x \* cf, 0, z \* cf \* \_zDir)) \* worldTranslationBehindDown;



Input pattern gets reversed also…decided against changing z-direction. Back to normal.

# Class to Dataset

Dataset is a collection of Datatables. You can define relationships between those tables.

## Class to DataTable

Forum:

<http://stackoverflow.com/questions/18746064/using-reflection-to-create-a-datatable-from-a-class>

my favorite homemade function. it create and populate all at same time. throw any object.

public static DataTable ObjectToData(object o)

{

DataTable dt = new DataTable("OutputData");

DataRow dr = dt.NewRow();

dt.Rows.Add(dr);

o.GetType().GetProperties().ToList().ForEach(f =>

{

try

{

f.GetValue(o, null);

dt.Columns.Add(f.Name, f.PropertyType);

dt.Rows[0][f.Name] = f.GetValue(o, null);

}

catch { }

});

}

Amother good write-up on data binding:

<http://www.developer.com/net/vb/article.php/1146371/Serializing-Objects-to-a-DataSet-in-Visual-Basic-NET.htm>

Looks like it may be possible to do automatic Data – Dataset translation this way:

1. For Cell class for example, create CellData class with items you want to display.
2. – or can you just use [Serializable] attribute???
3. Use XML Serializer as described here: <http://www.developer.com/net/vb/article.php/1146371/Serializing-Objects-to-a-DataSet-in-Visual-Basic-NET.htm>

## Elements data to display:

## Cell

Column Column

int Index //Position in Column

List<SegmentUpdate> SegmentUpdates

List<bool> ActiveState

List<bool> PredictiveState

List<bool> LearnState

bool IsSegmentPredicting

int NumberPredictionSteps

bool IsDataGridSelected

StatisticsCell Statistics

float LearningCounter

float NumberSegments

float MaxNumberSynapses

Statistics:

float ActivityCounter /// Records active cells.

float ActivityPrecision

float ActivityRate

float CorrectSegmentPredictionCounter

float CorrectPredictionCounter

float PredictPrecision

float SegmentPredictionCounter

float StepCounter

Selectable3DObject:

mouseSelected

mouseOver

isVisible

## Distal Segment : Segment

int MaxTimeSteps

bool IsSequence

int NumberPredictionSteps

Segment:

float ActivationThreshold

List<Synapse> Synapses

List<bool> ActiveState

## Proximal Segment : Segment

Segment:

float ActivationThreshold

List<Synapse> Synapses

List<bool> ActiveState

To implekment XMLSerializer following files had memebers with private/no setter altered to public setter:

Cell.cs

DistalSegment.cs

Column.cs

SegmentUpdate.cs

Region.cs - also jagged array disabled from serialization

# DataTable