

```

classDiagram
    class CustomObject {
        +ChangeEvent: EventHandler
        #OnChangeEvent()
    }
    class ObjectBounds {
        +MinX,MaxX: double
        +MinY,MaxY: double
        +MinZ,MaxZ: double
        +Create() override
        +Destroy() override
        +SetBounds(minX, maxX, minY, maxY, minZ, maxZ: double)
    }
    class CustomGraphObject {
        +Color: ColorRGB
        +ContourColor: ColorRGB
        +Visible: bool
        +Bounds: ObjectBounds
        +Position: Position3D
        +BoundsChangeEvent: EventHandler
        -BoundsChangeHandler(sender: object, e: EventArgs)
        -PositionChangeHandler(sender: object, e: EventArgs)
        #OnBoundsChangeEvent() virtual
        #OnPositionChangeEvent() abstract
        #Invalidate() virtual
        +Draw() abstract
        +SetColor(red, green, blue: double)
    }
    class GraphObject {
        #DefineBounds() abstract
        +ParamsDlg() abstract
    }
    class Orbit {
        +SemimajorAxis: double
        +SemiminorAxis: double
        +MeanAnomaly: double
        +Inclination: double
        +LongitudeOfAscendingNode: double
        +Eccentricity: double
        +ArgumentOfPeriapsis: double
        +ShowGrid: bool
        +ShowAxes: bool
        +ShowBaseOrbit: bool
        +ShowBaseOrbitAxes: bool
        +ShowBoundsContour: bool
        #OnPositionChangeEvent() override
        #DefineBounds() override
        -DefineAdditionalOrbitElements()
        -EnsureAngle(angle, minAngle, maxAngle: double): double
        -DrawBoundsContour()
        -DrawAxes()
        -DrawOrbit()
        -DrawBaseOrbit()
        -DrawGrid()
        +Draw() override
        +ParamsDlg() override
    }
    class CustomPosition3D {
        +Vector: Vector3D
    }
    class Position3D {
        +MoveTo(x, y, z: double)
        +MoveBy(deltaX, deltaY, deltaZ: double)
    }
    class AngularPosition3D {
        +RotateTo(x, y, z: double)
        +RotateBy(deltaX, deltaY, deltaZ: double)
    }
    class Camera {
        +Position: Vector3D
        +ViewPoint: Vector3D
        +FieldOfView: double
        +SphericalCoords: SphericalCoordinates
        +RenderEvent: EventHandler
        #OnRenderEvent()
        #DefineSphericalCoordsFromPosition()
        +SetPositionBySphericalCoords(azimuth, zenith, radius: double)
        +AdjustDistanceToTarget(distanceRatio: double)
        +MoveAroundTarget(pitchDelta, turnDelta: double)
    }
    class Scene {
        +HWnd: SimpleOpenGLControl
        +ScreenSize: ScreenSize
        +CubeBounds: bounds
        +FrustumFarPlane: double
        +ForeColor: ColorRGB
        +ShowCube: bool
        +ShowCubeAxes: bool
        +GraphObjects: Objects
        +Camera: Camera
        +Cursor: Cursor
        #glInit()
        #glRender()
        #glBitmapFont()
        -CameraRenderHandler(sender: object, e: EventArgs)
        -CameraChangeHandler(sender: object, e: EventArgs)
        -ObjectsChangeHandler(sender: object, e: EventArgs)
        #DrawObjects()
        #DrawCubeContour()
        +SetCubeBounds(minX, maxX, minY, maxY, minZ, maxZ: double)
        +SetScreenSize(screenWidth, screenHeight: double)
        +Draw()
        +ParamsDlg()
        +CameraCentered()
        +DefineCubeBounds()
        +SelectGraphObject(cursorPos: Cursor)
    }
    class MainForm {
        -CursorPosX: integer
        -CursorPosY: integer
        -MousePressed: bool
        -FullScreenMode: bool
        -Scene: Scene
        -Orbit1: Orbit
        -Orbit2: Orbit
        -Orbit3: Orbit
        -SceneChangeHandler(sender: object, e: EventArgs)
        -BoundsChangeHandler(sender: object, e: EventArgs)
        -SetFullScreenMode(fullScreenMode: bool)
    }
    class GraphObject2 {
        +Items: List<GraphObject>
        +SelectIndex: int
        +SelectedItem: GraphObject
        +Selected: boolean
        +Add(graphObj: GraphObject)
        +Clear()
        +Select(cursorPosX, cursorPosY, screenHeight: int; fieldOfView: double)
    }
    CustomObject <|-- ObjectBounds
    CustomObject <|-- CustomGraphObject
    CustomGraphObject <|-- GraphObject
    CustomGraphObject <|-- Orbit
    CustomPosition3D <|-- Position3D
    CustomPosition3D <|-- AngularPosition3D
    Camera *--> "1..1" Scene
    Scene *--> "1..1" CustomGraphObject
    Scene *--> "1..1" Orbit
    Scene *--> "1..1" GraphObject2
    MainForm *--> "1..1" Scene
    MainForm *--> "1..1" Orbit
    
```

The diagram illustrates the architecture of a Sun Sensor System. It features a hierarchy of classes: **CustomObject** is the base class, with **ObjectBounds** and **CustomGraphObject** as subclasses. **CustomGraphObject** further branches into **GraphObject** and **Orbit**. **CustomPosition3D** is a base class for **Position3D** and **AngularPosition3D**. The **Camera** class is associated with the **Scene** class (1..1 multiplicity). The **Scene** class is associated with **CustomGraphObject** (1..1), **Orbit** (1..1), and a **GraphObject** (1..1). The **MainForm** class is associated with the **Scene** class (1..1) and the **Orbit** class (1..1). The **Scene** class also has a static association with **VectorOperations**.

CustomObject (Base Class):

- +ChangeEvent: EventHandler
- #OnChangeEvent()

ObjectBounds (Subclass of CustomObject):

- +MinX,MaxX: double
- +MinY,MaxY: double
- +MinZ,MaxZ: double
- +Create() override
- +Destroy() override
- +SetBounds(minX, maxX, minY, maxY, minZ, maxZ: double)

CustomGraphObject (Subclass of CustomObject):

- +Color: ColorRGB
- +ContourColor: ColorRGB
- +Visible: bool
- +Bounds: ObjectBounds
- +Position: Position3D
- +BoundsChangeEvent: EventHandler
- BoundsChangeHandler(sender: object, e: EventArgs)
- PositionChangeHandler(sender: object, e: EventArgs)
- #OnBoundsChangeEvent() virtual
- #OnPositionChangeEvent() abstract
- #Invalidate() virtual
- +Draw() abstract
- +SetColor(red, green, blue: double)

GraphObject (Subclass of CustomGraphObject):

- #DefineBounds() abstract
- +ParamsDlg() abstract

Orbit (Subclass of CustomGraphObject):

- +SemimajorAxis: double
- +SemiminorAxis: double
- +MeanAnomaly: double
- +Inclination: double
- +LongitudeOfAscendingNode: double
- +Eccentricity: double
- +ArgumentOfPeriapsis: double
- +ShowGrid: bool
- +ShowAxes: bool
- +ShowBaseOrbit: bool
- +ShowBaseOrbitAxes: bool
- +ShowBoundsContour: bool
- #OnPositionChangeEvent() override
- #DefineBounds() override
- DefineAdditionalOrbitElements()
- EnsureAngle(angle, minAngle, maxAngle: double): double
- DrawBoundsContour()
- DrawAxes()
- DrawOrbit()
- DrawBaseOrbit()
- DrawGrid()
- +Draw() override
- +ParamsDlg() override

CustomPosition3D (Base Class):

- +Vector: Vector3D

Position3D (Subclass of CustomPosition3D):

- +MoveTo(x, y, z: double)
- +MoveBy(deltaX, deltaY, deltaZ: double)

AngularPosition3D (Subclass of CustomPosition3D):

- +RotateTo(x, y, z: double)
- +RotateBy(deltaX, deltaY, deltaZ: double)

Camera (Class):

- +Position: Vector3D
- +ViewPoint: Vector3D
- +FieldOfView: double
- +SphericalCoords: SphericalCoordinates
- +RenderEvent: EventHandler
- #OnRenderEvent()
- #DefineSphericalCoordsFromPosition()
- +SetPositionBySphericalCoords(azimuth, zenith, radius: double)
- +AdjustDistanceToTarget(distanceRatio: double)
- +MoveAroundTarget(pitchDelta, turnDelta: double)

Scene (Class):

- +HWnd: SimpleOpenGLControl
- +ScreenSize: ScreenSize
- +CubeBounds: bounds
- +FrustumFarPlane: double
- +ForeColor: ColorRGB
- +ShowCube: bool
- +ShowCubeAxes: bool
- +GraphObjects: Objects
- +Camera: Camera
- +Cursor: Cursor
- #glInit()
- #glRender()
- #glBitmapFont()
- CameraRenderHandler(sender: object, e: EventArgs)
- CameraChangeHandler(sender: object, e: EventArgs)
- ObjectsChangeHandler(sender: object, e: EventArgs)
- #DrawObjects()
- #DrawCubeContour()
- +SetCubeBounds(minX, maxX, minY, maxY, minZ, maxZ: double)
- +SetScreenSize(screenWidth, screenHeight: double)
- +Draw()
- +ParamsDlg()
- +CameraCentered()
- +DefineCubeBounds()
- +SelectGraphObject(cursorPos: Cursor)

MainForm (Class):

- CursorPosX: integer
- CursorPosY: integer
- MousePressed: bool
- FullScreenMode: bool
- Scene: Scene
- Orbit1: Orbit
- Orbit2: Orbit
- Orbit3: Orbit
- SceneChangeHandler(sender: object, e: EventArgs)
- BoundsChangeHandler(sender: object, e: EventArgs)
- SetFullScreenMode(fullScreenMode: bool)

GraphObject (Class):

- +Items: List<GraphObject>
- +SelectIndex: int
- +SelectedItem: GraphObject
- +Selected: boolean
- +Add(graphObj: GraphObject)
- +Clear()
- +Select(cursorPosX, cursorPosY, screenHeight: int; fieldOfView: double)

static VectorOperations (Static Class):

- +SetVector(x, y, z: double): Vector3D
- +SetVector(): Vector3D
- +SetVector(vector: Vector3D): Vector3D
- +VectorLength(vector: Vector3D): double
- +NormalizeVector(vector: Vector3D): Vector3D
- +VectorSubtract(vector1, vector2: Vector3D): Vector3D
- +VectorAdd(vector1, vector2: Vector3D): Vector3D
- +VectorCrossProduct(vector1, vector2: Vector3D): Vector3D
- +VectorDotProduct(vector1, vector2: Vector3D): double
- +ScaleVector(vector: Vector3D; factor: double): Vector3D
- +CreateRotationMatrix(const axis: Vector3D; angle: double): double[0..2, 0..2]
- +VectorTransform(vector: Vector3D; matrix: double[0..2, 0..2]): Vector3D
- +RotateVector(vector, axis: Vector3D; angle: double): Vector3D
- +MoveAroundTarget(position, viewPoint: Vector3D; const turnDelta, pitchDelta: double): double
- +AdjustDistanceToTarget(position, viewPoint: Vector3D; distanceRatio: double): double

